

Complete Technical
Information for the
Support Professional



Microsoft
WINDOWS
FOR WORKGROUPS Resource
Kit

Addendum for Operating System Version 3.11

**Microsoft®
Windows™ for Workgroups
Resource Kit**

***Complete Technical Information
for the Support Professional
for Microsoft Windows for Workgroups
Version 3.1***

Microsoft Corporation

Information in this document is subject to change without notice and does not represent a commitment on the part of Microsoft Corporation. The software described in this document is furnished under a license agreement or nondisclosure agreement. The software may be used or copied only in accordance with the terms of the agreement. It is against the law to copy the software on any medium except as specifically allowed in the license or nondisclosure agreement. The purchaser may make one copy of the software for backup purposes. No part of this manual may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or information recording and retrieval systems, for any purpose other than the purchaser's personal use, without the express written permission of Microsoft.

© 1992 Microsoft Corporation. All rights reserved.

Printed in the United States of America. 1 2 3 4 5 6 7 8 9

Microsoft, MS, and MS-DOS are registered trademarks and Windows and the Windows logo are trademarks of Microsoft Corporation.

This document was prepared with Microsoft® Word for Windows™ 2.0 and TrueType®.

Document No. 0030-37275

1-2-3 is a registered trademark of Lotus Development Corporation. 3+ is a registered trademark of 3Com Corporation. 3+Mail is a registered trademark of 3Com Corporation. 3+Open is a trademark of 3Com Corporation. 3+Share is a registered trademark of 3Com Corporation. 386MAX is a registered trademark of Qualitas, Inc. 3Com is a registered trademark of 3Com Corporation. Adobe is a registered trademark of Adobe Systems, Inc. Adobe Type Manager is a registered trademark of Adobe Systems, Inc. ArcNet is a registered trademark of Datapoint Corporation. AS/400 is a registered trademark of International Business Machines Corporation. AST is a registered trademark of AST Research, Inc. AT&T is a registered trademark of American Telephone and Telegraph Company. AT&T Easylink is a servicemark of American Telephone and Telegraph Company. Banyan is a registered trademark of Banyan Systems, Inc. Bitstream is a registered trademark of Bitstream, Inc. BlueMAX is a trademark of Qualitas, Inc. Borland is a registered trademark of Borland International, Inc. Canon is a registered trademark of Canon USA Inc. cc:Mail is a trademark of cc:Mail, Inc., a wholly owned subsidiary of Lotus Development Corporation. COMPAQ is a registered trademark of Compaq Computer Corporation. CompuServe is a registered trademark of CompuServe, Inc. Da Vinci Systems is a trademark of Da Vinci Systems Corporation. Data General is a registered trademark of Data General Corporation. DCA is a registered trademark of Digital Communications Associates, Inc. DEC is a registered trademark of Digital Equipment Corporation. DeskJet is a registered trademark of Hewlett-Packard Company. Easylink is a registered trademark of American Telephone and Telegraph Company. Epson is a registered trademark of Seiko Epson Corporation, Inc. EtherExpress is a trademark of Intel Corporation. EtherLink is a registered trademark of 3Com Corporation. Everex is a trademark of Everex Systems, Inc. Excelan is a registered trademark of Novell, Inc. ExecJet is a registered trademark of International Business Machines Corporation. Facelift is a trademark of Bitstream, Incorporated. GENie is a trademark of General Electric Corporation. Hercules is a registered trademark of Hercules Computer Technology, Inc. Hewlett-Packard is a registered trademark of Hewlett-Packard Company. IBM is a registered trademark of International Business Machines Corporation. Intel is a registered trademark of Intel Corporation. Intel386 is a trademark of Intel Corporation. Kodak is a registered trademark of Eastman Kodak Company. LANSight is a trademark of Intel Corporation. LaserJet is a registered trademark of Hewlett-Packard Company. Linotronic is a trademark of Linotype AG and its subsidiaries. Logitech is a trademark of Logitech, Inc. Lotus is a registered trademark of Lotus Development Corporation. Lotus Notes is a registered trademark of Lotus Development Corporation. Mac is a registered trademark of Apple Computer, Inc. Macintosh is a registered trademark of Apple Computer, Inc. MCI MAIL is a registered servicemark (use ®) of MCI Communications Corp. Micro Channel is a registered trademark of International Business Machines Corporation. NEC is a registered trademark of NEC Corporation. NetWare is a registered trademark of Novell, Inc. Norton DiskDoctor is a registered trademark of Peter Norton Computing. Norton Utilities is a registered trademark of Peter Norton Computing. NotesMail is a trademark of Lotus Development Corporation. Novell is a registered trademark of Novell, Inc. ObjectView is a trademark of MATESYS CORPORATION N.A. OfficeVision is a trademark of International Business Machines Corporation. Okidata is a registered trademark of Oki America, Inc. Olivetti is a registered trademark of Ing. C. Olivetti. Operating System/2 is a registered trademark of International Business Machines Corporation. OS/2 is a registered trademark licensed to Microsoft Corporation. OS/400 is a registered trademark of International Business Machines Corporation. PaintJet is a registered trademark of Hewlett-Packard Company. Panasonic is a registered trademark of Matsushita Electric Co., Ltd. Pathworks is a trademark of Digital Equipment Corporation. PC Tools Deluxe is a trademark of Central Point Software, Inc. PC Tools is a trademark of Central Point Software, Inc. PC/XT is a trademark of International Business Machines Corporation. Personal Computer AT is a registered trademark of International Business Machines Corporation. Personal Computer XT is a trademark of International Business Machines Corporation. Personal System/1 is a trademark of International Business Machines Corporation. Personal System/2 is a registered trademark of International Business Machines Corporation. Philips is a registered trademark of Philips International B.V. PostScript is a registered trademark of Adobe Systems, Inc. PROFS is a registered trademark of International Business Machines Corporation. Qemm is a trademark of Quarterdeck Office Systems. Qualitas is a registered trademark of Qualitas, Inc. Quickwriter is a registered trademark of International Business Machines Corporation. Quietwriter is a registered trademark of International Business Machines Corporation. Retix is a registered trademark of Retix. Rumba is a registered trademark of Wall Data Incorporated. SatisFAXtion is a registered trademark of Intel Corporation. SideKick is a registered trademark of Borland International, Inc. Soft-Switch is a trademark of Soft-Switch Inc. Sound Blaster is a trademark of Creative Technology Ltd. Sound Blaster Pro is a trademark of Creative Technology Ltd. Stacker is a trademark of STAC Electronics. System/360 is a trademark of International Business Machines Corporation. TCS is a registered trademark of Eurotherm International P.L.C. Tektronix is a registered trademark of Tektronix, Inc. Texas Instruments is a registered trademark of Texas Instruments, Inc. ThinkJet is a registered trademark of Hewlett-Packard Company. TokenExpress is a trademark of Intel Corporation. Toshiba is a registered trademark of Kabushiki Kaisha Toshiba. Triumph is a registered trademark of Triumph Adler AG. TrueType is a registered trademark of Apple Computer, Inc. Tulip is a registered trademark of Tulip Computers International, BV. Unisys is a registered trademark of Unisys Corporation. Vectra is a registered trademark of H-P Company. Video Seven is a trademark of Headland Technology, Inc. VINES is a registered trademark of Banyan Systems, Inc. VMS is a registered trademark of Digital Equipment Corporation. WANG is a registered trademark of Wang Laboratories. Western Digital is a trademark of Western Digital Corporation. WordPerfect is a registered trademark of WordPerfect Corporation. WYSE is a registered trademark of Wyse Technology. Xerox is a registered trademark of Xerox Corporation. XGA is a registered trademark of International Business Machines Corporation. XT is a trademark of International Business Machines Corporation. Zenith is a registered trademark of Zenith Data Systems Corporation.

UNLESS NOTED OTHERWISE IN ANY PORTION OF THE SOFTWARE, THE FOLLOWING TERMS AND CONDITIONS SHALL APPLY TO THE USE OF THE SOFTWARE.

MICROSOFT SOFTWARE LICENSE: Windows for Workgroups Resource Kit

1. GRANT OF LICENSE. Microsoft grants to you the right to use one copy of the enclosed Microsoft software program (Software) on a single terminal connected to a single computer (i.e., with a single CPU). You may not network the Software or otherwise use it on more than one computer or computer terminal at the same time.

2. COPYRIGHT. The Software is owned by Microsoft or its suppliers and is protected by United States copyright laws and international treaty provisions. Therefore, you must treat the Software like any other copyrighted material (e.g., a book or musical recording) except that you may either (a) make one copy of the Software solely for backup or archival purposes, or (b) transfer the Software to a single hard disk provided you keep the original solely for backup or archival purposes. You may not copy the written materials accompanying the Software.

3. OTHER RESTRICTIONS. This Microsoft License Agreement is your proof of license to exercise the rights granted herein and must be retained by you. You may not rent or lease the Software. You may not reverse engineer, decompile or disassemble the Software.

NO WARRANTY. ANY USE BY YOU OF THE SOFTWARE IS AT YOUR OWN RISK. THE SOFTWARE IS PROVIDED FOR USE AS IS WITHOUT WARRANTY OF ANY KIND. MICROSOFT AND ITS SUPPLIERS DISCLAIM ALL WARRANTIES OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, WITHOUT LIMITATION, IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

NO LIABILITY FOR CONSEQUENTIAL DAMAGES. In no event shall Microsoft or its suppliers be liable for any damages whatsoever (including, without limitation, damages for loss of business profits, business interruption, loss of business information, or other pecuniary loss) arising out of the use or inability to use this Microsoft product, even if Microsoft has been advised of the possibility of such damages. Because some jurisdictions do not allow the exclusion or limitation of liability for consequential or incidental damages, the above limitation may not apply to you.

You may have additional rights, which vary from jurisdiction to jurisdiction.

U.S. GOVERNMENT RESTRICTED RIGHTS

The Software and documentation are provided with RESTRICTED RIGHTS. Use, duplication, or disclosure by the Government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of The Rights in Technical Data and Computer Software clause at DFARS 252.227-7013 or subparagraphs (c)(1) and (2) of the Commercial Computer Software Restricted Rights at 48 CFR 52.227-19, as applicable. Manufacturer is Microsoft Corporation, One Microsoft Way, Redmond, WA 98052-6399.

If you acquired the Software in the United States, this Agreement shall be governed by the laws of the State of Washington.

Should you have any questions concerning the Microsoft software, please write your local Microsoft subsidiary or: Microsoft Customer Sales and Service, One Microsoft Way, Redmond, WA 98052-6399.

Contents

Microsoft Software License Agreement.....iii

Welcome **xi**

About the Windows for Workgroups Resource Kit xii

Conventions in This Manual..... xiv

Part 1 Technical Overview

**Chapter
1**

Networking—A Technical Discussion **1-1**

Overview of Networking..... 1-3

Physical Characteristics of a LAN..... 1-10

Popular LAN Architectures..... 1-19

Network Adapter Cards 1-25

Network Device Interface Specification (NDIS)..... 1-29

Network Protocols and Transports 1-31

Communication Between Network Transports and Applications 1-34

Servers and Workstations 1-37

Additional Reading..... 1-40

**Chapter
2**

Windows for Workgroups Architecture **2-1**

Review of Windows 3.1 Architecture 2-2

Windows for Workgroups 3.1 Architecture..... 2-11

Part 2 Installation and Setup

**Chapter
3**

Windows for Workgroups Installation **3-1**

About Windows for Workgroups Setup 3-2

Windows for Workgroups Setup: A Technical Discussion..... 3-7

Setting Up Windows for Workgroups for a Network Installation..... 3-17

Using Automated Setup: Setup /h 3-22

Maintaining Windows for Workgroups with Setup 3-31

Chapter			
4	Windows for Workgroups Files		4-1
	About the Windows for Workgroups Files		4-2
	WIN.COM		4-2
	The Core Files		4-3
	Drivers, Fonts, and International Support Files		4-3
	MS-DOS Support Components of Windows for Workgroups		4-12
	Windows for Workgroups Applications, Setup, and Other Files		4-17
	Files You Can Delete		4-22
Chapter			
5	Windows for Workgroups Setup Information Files		5-1
	About the Setup Information Files		5-3
	SETUP.INF: Setup Information File		5-6
	APPS.INF: Application Information File		5-42
	CONTROL.INF: The Windows Control Panel Information File		5-48
	NETWORK.INF: Network Information File		5-50
	OEMSETUP.INF: OEM Network Setup Information File		5-64
	Modifying .INF Files for Custom Installations		5-69
Chapter			
6	The Windows for Workgroups Initialization Files		6-1
	About the Initialization Files		6-2
	WIN.INI: Windows for Workgroups Initialization File		6-6
	SYSTEM.INI: System Initialization File		6-29
	PROTOCOL.INI: Network Protocol Initialization File		6-73
	PROGMAN.INI: Program Manager Initialization File		6-76
	CONTROL.INI: Control Panel Initialization File		6-79
	WINFILE.INI: File Manager Initialization File		6-80
	MSMAIL.INI: Microsoft Mail Initialization File		6-82
	SHARED.INI: Shared Mail Settings Initialization File		6-94
	SCHDPLUS.INI: Schedule+ Initialization File		6-95

Part 3 Special Topics

Chapter			
7	Additional Windows for Workgroups Information		7-1
	About Windows for Workgroups Information		7-2
	Setup and Installation Issues		7-2
	Network Information		7-4
	Information about Using Windows for Workgroups		7-12

Chapter			
8	Network Integration with Microsoft LAN Manager and Novell NetWare		8-1
	About Network Integration with Microsoft LAN Manager and Novell NetWare		8-3
	Microsoft LAN Manager and Compatible Networks.....		8-4
	Novell NetWare Networks		8-13

Part 4 Configuring Windows for Workgroups

Chapter			
9	Tips for Configuring Windows for Workgroups		9-1
	Tips for Optimizing Performance.....		9-2
	Optimizing Server/Application Performance.....		9-16

Part 5 Using Windows for Workgroups

Chapter			
10	New and Updated Accessories		10-1
	Enhancement to Common Dialogs.....		10-2
	ClipBook Viewer.....		10-6
	Net Watcher		10-9
	WinMeter		10-12
	Hearts.....		10-14
	File Manager		10-14
	Print Manager		10-17
Chapter			
11	Network DDE		11-1
	About Network DDE		11-3
	Overview of Dynamic Data Exchange (DDE)		11-3
	Overview of Network DDE		11-11
	Network DDE Security		11-14
	Examples of How Network DDE Works.....		11-18
	Using Network DDE with Application Macro Languages		11-23
	Questions and Answers about Network DDE		11-24

Chapter 12	Mail	12-1
	About Mail	12-3
	The Mail Client	12-3
	The Mail Postoffice	12-8
	Interface Between the Mail Client and Postoffice	12-11
	Customizing the Mail Client	12-13
	Integrating Mail and Other Windows-Based Applications	12-27
	Differences Between Windows for Workgroups Mail and Mail 3.x	12-28
	Extending the Reach of Mail	12-29
	Tips for Using Mail	12-35
	Questions and Answers about Mail	12-38
Chapter 13	Schedule+	13-1
	Overview of Schedule+ Architecture	13-2
	Custom Message Types for Schedule+	13-7
	Schedule+ Interoperability	13-9
	Extending the Reach of Schedule+	13-17
	Questions and Answers about Schedule+	13-18

Part 6 Troubleshooting Windows for Workgroups

Chapter 14	Troubleshooting Windows for Workgroups	14-1
	About Troubleshooting	14-3
	Troubleshooting Setup	14-7
	Troubleshooting Windows for Workgroups Configuration	14-11
	Troubleshooting Windows for Workgroups Running in Standard Mode	14-25
	Troubleshooting Windows for Workgroups Running in 386 Enhanced Mode	14-27
	Troubleshooting for MS-DOS-Based Applications	14-30
	Troubleshooting Fonts	14-32
	Troubleshooting Printing	14-33
	Troubleshooting Networks	14-37
	Troubleshooting GP Faults	14-42
	Troubleshooting Multimedia	14-45

Part 7 References, Resources, and Appendixes

	Glossary	1
Appendix A	Windows for Workgroups Resource Directory	A-1
Appendix B	Microsoft Mail Configuration Guide	B-1
Appendix C	Flowcharts for Troubleshooting Windows for Workgroups	C-1
Appendix D	Accessibility for Individuals with Disabilities	D-1
Appendix E	Integrating Additional Protocols with Windows for Workgroups	E-1
	Index	1

Welcome

Welcome to the *Windows™ for Workgroups Resource Kit Version 3.1*. This manual is a companion guide to the *Windows Resource Kit for Microsoft Windows Operating System Version 3.1* and is designed for people who are, or who want to become, expert users of Microsoft® Windows™ for Workgroups version 3.1. The *Windows for Workgroups Resource Kit* consists of a manual containing technical reference information and a disk containing helpful utilities and accessories designed to help you get the most out of Windows for Workgroups.

The *Windows for Workgroups Resource Kit* presents a detailed, easy-to-read technical view of Windows for Workgroups, so that you can better manage how Windows for Workgroups is used at your site. The *Windows for Workgroups Resource Kit* also contains specific information for system administrators who are responsible for installing, managing, and integrating Windows for Workgroups in a network or multi-user environment.

This introductory chapter presents three kinds of information you can use to get started:

- The first section outlines the contents of the *Windows for Workgroups Resource Kit*, so you can quickly find technical details about specific elements of Microsoft Windows for Workgroups.
- The second section contains an overview of the conventions used to present information in the *Windows for Workgroups Resource Kit*.
- The third section presents a series of troubleshooting flowcharts, so you can quickly find details and procedures for solving problems you might have installing or running Windows for Workgroups.

The *Windows for Workgroups Resource Kit* is a technical supplement to the documentation that is included in your Windows for Workgroups product and does not replace that information as the source for learning how to use Windows for Workgroups features and Windows-based applications.

Contents

About the Windows for Workgroups Resource Kit	xii
Conventions in This Manual	xiv
Document Conventions	xiv
Syntax Conventions	xv

About the Windows for Workgroups Resource Kit

This guide is organized in seven parts that provide specific details about Windows for Workgroups architecture, installation and setup, networks, configuration, new and updated accessories, and troubleshooting.

Part 1: Technical Overview

- **Chapter 1, “Networking—A Technical Discussion,”** contains information targeted toward the support professional that may not have a local area network (LAN) background. This chapter provides a technical discussion of networking concepts and discusses the components that make up a LAN.
- **Chapter 2, “Windows for Workgroups Architecture,”** describes the architecture used by Windows for Workgroups and discusses the components of Windows for Workgroups that differentiate it from Windows version 3.1.

Part 2: Installation and Setup

- **Chapter 3, “Windows for Workgroups Installation,”** contains a technical discussion of the Windows for Workgroups Setup program, details about setting up Windows for Workgroups on a network, and instructions for creating a custom installation routine for automated setup.
- **Chapter 4, “Windows for Workgroups Files,”** describes the purpose for each file installed by Windows for Workgroups in the WINDOWS directory and the Windows SYSTEM subdirectory.
- **Chapter 5, “Windows for Workgroups Setup Information Files,”** contains the details you need to understand to create custom Windows for Workgroups setup information files (SETUP.INF, NETWORK.INF, CONTROL.INF, APPS.INF, and OEMSETUP.INF) for multiple installations.
- **Chapter 6, “Windows for Workgroups Initialization Files,”** describes the contents of Windows for Workgroups initialization files, including WIN.INI, SYSTEM.INI, PROTOCOL.INI, MSMAIL.INI, and SCHDPLUS.INI, and explains how you can change entries in these files.

Part 3: Special Topics

- **Chapter 7, “Additional Windows for Workgroups Information,”** contains tips about using Microsoft Windows for Workgroups. This chapter also discusses special topics related to the components of Windows for Workgroups.

- **Chapter 8, “Network Integration with Microsoft LAN Manager and Novell NetWare,”** contains information about integrating a Windows for Workgroups workstation into an existing LAN environment. Topics covered include integrating Windows for Workgroups with Microsoft LAN Manager and Novell® NetWare® networks.

Part 4: Configuring Windows for Workgroups

- **Chapter 9, “Tips for Configuring Windows for Workgroups,”** presents tips about configuring your system, both for gaining optimal performance and for creating custom Windows for Workgroups configurations.

Part 5: Using Windows for Workgroups

- **Chapter 10, “New and Updated Accessories,”** discusses accessories new for Windows for Workgroups and provides new information about Windows 3.1 accessories that have been updated for Windows for Workgroups.
- **Chapter 11, “Network Dynamic Data Exchange,”** discusses how Dynamic Data Exchange (DDE) functionality has been extended to the network environment to enable information to be exchanged dynamically with other workstations in your Windows for Workgroups workgroup.
- **Chapter 12, “Mail,”** includes information about the Mail application provided with Windows for Workgroups. The architecture of Mail, tips for customizing Mail, and information about integrating Mail with other Windows-based applications are discussed in this chapter.
- **Chapter 13, “Schedule+,”** includes information about the Schedule+ application provided with Windows for Workgroups. Schedule+ architecture and key features of interest to system administrators are discussed in this chapter.

Part 6: Troubleshooting Windows for Workgroups

- **Chapter 14, “Troubleshooting Windows for Workgroups 3.1,”** provides specific information for troubleshooting problems with Windows for Workgroups, showing the key steps for isolating and solving common problems.

Part 7: References, Resources, and Appendixes

This part of the *Windows for Workgroups Resource Kit* contains a glossary, a directory of information resources, a configuration guide for Mail, information about using additional protocols with Windows for Workgroups, and an index.

Conventions in This Manual

This document assumes that you have read the Windows for Workgroups 3.1 documentation set and that you are familiar with using menus, dialog boxes, and other Windows features. It also assumes that you have installed Windows for Workgroups on your system and that you are using a mouse with Windows. For keyboard equivalents to the actions described here, see the Microsoft Windows for Workgroups online Help.

This document uses several conventions to help you identify information.

Document Conventions

The following table describes the typographical conventions used in the *Windows for Workgroups Resource Kit*.

Type style	Used for
bold	MS-DOS® command names such as copy or dir , switches such as /? or /3 , section and entry names in .INI and .INF files such as [386Enh] or emmexclude= , and text that you type to carry out actions at the command prompt.
<i>italic</i>	Parameter values for which you can supply specific values. For example, to supply a value for a parameter that calls for a <i>filename</i> , you must type a specific filename such as MYFILE.EXE.
ALL CAPITALS	Directory names, filenames, and acronyms. For example, "WINDOWS" is used to represent the Windows main directory, and "SYSTEM" represents the Windows System subdirectory. When you type directory names and filenames at the command prompt or in a dialog box, lowercase letters may be used.

Other conventions in this document include:

- 
- “Windows” refers to Microsoft Windows version 3.1 or later (including Windows for Workgroups).
 - “MS-DOS” refers to Microsoft MS-DOS version 3.1 or later.
 - The Microsoft Windows logo appears in the margin to highlight specific features that are new to Windows for Workgroups 3.1 or updated from Windows 3.1.
 - “Windows-based application” is used as a shorthand term to refer to an application that is designed to run with Windows and does not run without Windows. All Windows applications follow similar conventions for the arrangement of menus, style of dialog boxes, and keyboard and mouse use.
 - “MS-DOS-based application” is used in this document as a shorthand term to refer to an application that is designed to run with MS-DOS but not specifically with Windows and is not able to take full advantage of Windows features (such as graphics or memory management).
 - “Command prompt” refers to the command line where you type MS-DOS commands. Typically, you see characters such as “C:\>” to show the location of the command prompt on your screen. When Windows is running, you can double-click the MS-DOS Prompt icon in Program Manager to use the command prompt.
 - An instruction to “type” any information means to press a key or a sequence of keys, and then press the ENTER key.
 - Mouse instructions in this document, such as “Click the OK button” or “Drag an icon in File Manager,” use the same meanings as the descriptions of mouse actions in the *Windows for Workgroups User’s Guide* and the Windows online tutorial.

Check the glossary at the end of the *Windows for Workgroups Resource Kit* for definitions of terms that you are unfamiliar with.

Syntax Conventions

“Syntax” refers to the order in which you must type an item such as an MS-DOS command with its switches or an entry in a Windows initialization (.INI) file. Elements that appear in bold must be typed exactly as they appear in the syntax example. Elements that appear in italic are placeholders for parameter values for which you must supply specific information.

Unless specified otherwise, you can type commands, keynames, parameters, and switches in either uppercase or lowercase letters.

This example shows the syntax for a SYSTEM.INI entry, with each item in the sample explained in the following table.

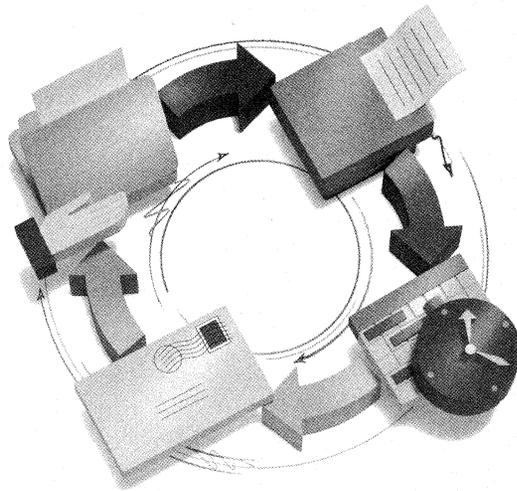
[*section*]
keyname=*value*, *string*, *source*, *destination*

Entry item	Definition
-------------------	-------------------

<i>[section]</i>	The name of a section in an initialization file or setup information file. The enclosing brackets ([]) are required, and the left bracket must be in the leftmost column on the screen—f example, [standard].
<i>keyname</i>	The name of an entry, which usually can consist of any combination of letters and digits. For many entries described in this document, the <i>keyname</i> must be followed immediately by an equal sign (=)—f example, run= .
<i>value</i>	An integer, a string, or a quoted string, depending on the entry.
<i>string</i>	A group of characters to be treated as a unit. A string can include letters, numbers, spaces, or any other characters. Sometimes the syntax definition will indicate that the string must be enclosed in double quotation marks (" ").
<i>source</i>	The location of data to be transferred to a specific destination or to be used as input to a command. <i>Source</i> can consist of a drive letter and colon, a directory name, a filename, or a combination of these elements.
<i>destination</i>	A location to which the data specified by <i>source</i> is to be transferred. <i>Destination</i> can consist of a drive letter and colon, a directory name, a filename, or a combination of these elements.

Note If you have MS-DOS version 5.0 or later on your system, you can get help for any MS-DOS commands, such as **mem**, and for many of the drivers, such as SMARTDRV.EXE, by typing the command name and **/?** at the command prompt (for example, **mem /?**). You can also type **help** plus the command name. Type **help** at the command prompt to see a list of all MS-DOS commands with a brief description of command syntax, parameters, and switches.

Technical Overview



Part

1

Technical Overview

Chapter 1 Networks—A Technical Discussion **1-1**

Overview of Networking.....	1-3
Physical Characteristics of a LAN.....	1-10
Popular LAN Architectures.....	1-19
Network Adapter Cards.....	1-25
Network Device Interface Specification (NDIS).....	1-29
Network Protocols and Transports.....	1-31
Communication Between Network Transports and Applications.....	1-34
Servers and Workstations.....	1-37
Additional Reading.....	1-40

Chapter 2 Windows for Workgroups Architecture **2-1**

Review of Windows 3.1 Architecture.....	2-2
Windows for Workgroups 3.1 Architecture.....	2-11

Networks—A Technical Discussion

This chapter describes the key components that make up a local area network. It also provides an overview of two key networking industry standards—the Open Systems Interconnection (OSI) reference model and the Institute of Electrical and Electronic Engineers (IEEE) 802 model—that define layers of network communication. This overview provides a foundation for the last part of the chapter, which details the networking components found in Windows for Workgroups.

Related information

- *Microsoft Windows for Workgroups version 3.1 Getting Started:* Chapter 2, “Installing a Network Adapter Card;” Chapter 10, “Configuring Hardware.”
- *Windows for Workgroups Resource Kit:* Chapter 2, “Windows for Workgroups Architecture;” Chapter 7, “Additional Windows for Workgroups Information;” Chapter 8, “Network Integration with Microsoft LAN Manager and Novell NetWare.”

Contents of This Chapter

Overview of Networking.....	1-3
OSI Reference Model.....	1-3
Sending Data Through the OSI Layers.....	1-7
IEEE 802 Model.....	1-8
Physical Characteristics of a LAN.....	1-10
LAN Topologies.....	1-10
LAN Cabling Media.....	1-13
Baseband and Broadband Transmission.....	1-17
Popular LAN Architectures.....	1-19
Ethernet.....	1-19
Token Ring.....	1-22
ArcNet.....	1-23
Summary of Popular LAN Architecture Specifications.....	1-24
Network Adapter Cards.....	1-25
Ethernet Cable Media Connector Types.....	1-25
Configuration Options.....	1-26
Network Device Interface Specification (NDIS).....	1-29
Network Protocols and Transports.....	1-31
NetBEUI.....	1-32
Dual-Net/Multiple Network Support.....	1-33
Communication Between Network Transports and Applications.....	1-34

NetBIOS.....	1-35
The Network Workstation Services.....	1-35
Servers and Workstations.....	1-37
Network Resource Management.....	1-37
LAN Security.....	1-39
Additional Reading.....	1-40

Overview of Networking

A *network* consists of two or more computers that are connected together by physical media and are running software which enables the computers to communicate with one another.

In the early years of networking, several large companies, including IBM®, Honeywell, and Digital Equipment Corporation (DEC®), each had their own standard for how computers could be connected together. These standards described the mechanisms necessary to move data from one computer to another. These early standards, however, were not entirely compatible. Networks adhering to IBM's Systems Network Architecture (SNA) could not directly communicate with networks using DEC's Digital Network Architecture (DNA), for example.

In later years, standards organizations, including the International Standards Organization (ISO) and the Institute of Electrical and Electronic Engineers (IEEE), developed models that became globally recognized and accepted as the standards for designing any computer network. Both models describe networking in terms of functional layers.

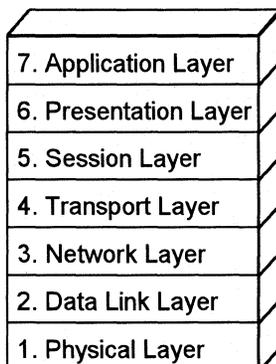
Before we examine the ISO and IEEE models, let's look at how the building blocks of a networking model, the functional layers, work together.

OSI Reference Model

In 1984, the International Organization for Standardization (ISO) developed a model called the *Open Systems Interconnection (OSI) reference model*. It is used to describe the flow of data between the physical connection to the network and the end-user application. This model is the best known and most widely used model to describe networking environments.

Figure 1.1

The OSI model defines seven levels of network communication.



As shown in Figure 1.1, the OSI layers are numbered from bottom to top. The most basic functions, such as putting data bits onto the network cable, are on the bottom, while functions attending to the details of applications are at the top.

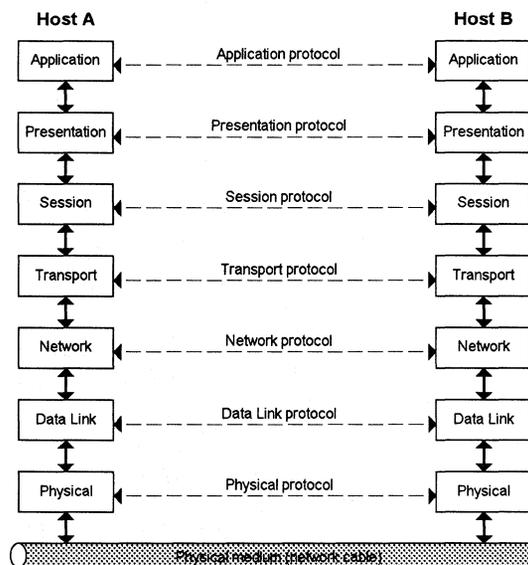
Relationship of OSI Model Layers

In the OSI model, the purpose of each layer is to provide services to the next higher layer, shielding the layer from the details of how the services are actually implemented. The layers are abstracted in such a way that each layer believes it is communicating with the associated layer on the other computer (labeled as *Host B* in Figure 1.2). In reality, each layer communicates only with adjacent layers on one computer.

Except for the lowest layer in the networking model, no layer can pass information directly to its counterpart on another computer. Information on the sending computer must be passed through all the lower layers. The information then moves across the networking cable to the receiving computer and up that computer's networking layers until arriving at the same level that sent the information. For example, if the Network layer sent information from Host A, it moves down through the Data Link and the Physical layers on the sending side, over the cable, and up the Physical and Data Link layers on the receiving side to its destination at the Network layer on Host B.

Figure 1.2

Relationship of the layers for the OSI reference model



Interaction between adjacent layers is called an *interface*. The interface defines which services the lower networking layer offers to the upper one and how those services will be accessed. In addition, each layer on one computer acts as though it is communicating directly with the same layer on another computer. The set of rules used to communicate between layers is called a *protocol*.

The following sections describe the purpose of each of the 7 layers of the OSI model and identify services that they provide to adjacent layers.

Physical Layer

The first and lowest layer of the OSI model is the *Physical layer*. This layer addresses the transmission of the unstructured raw bit stream over a physical medium (that is, the networking cable). The Physical layer relates the electrical/optical, mechanical, and functional interfaces to the cable. The Physical layer also carries the signals that transmit data generated by all the higher layers.

This layer defines how the cable is attached to the network adapter card. For example, it defines how many pins the connector has and what each pin is used for. It also defines which transmission technique will be used to send data over the network cable.

The Physical layer defines data encoding and bit synchronization, ensuring that when a transmitting host sends a 1 bit, it is received as a 1 bit, not a 0 bit. This also means defining how long each bit lasts and how each bit is translated into the appropriate electrical or optical impulse for the network cable.

This layer is responsible for getting bits (0s and 1s) from one computer to another. The bits themselves have no defined meaning at this level.

Data Link Layer

The second layer is called the *Data Link layer*. This is the layer where the bits have meaning on the networking model.

This layer packages raw bits from the Physical layer into *data frames*. A data frame is a logical, structured packet in which data can be placed. Figure 1.3 shows an example of a data frame. In this example, the sender ID represents the ID of the workstation that is sending the information, the destination ID represents the ID of the workstation that the information is being sent to, the frame type is used to identify the type of frame the data is representing, the data is the information itself, and the control represents error correction and verification information to ensure that the data frame is received properly.

Figure 1.3

Sender ID	Destination ID	Frame Type	Data	Control
-----------	----------------	------------	------	---------

A simple data frame

The Data Link layer is responsible for providing the error-free transfer of these frames from one computer to another through the Physical layer. This allows the Network layer to assume virtually error-free transmission over the network connection.

When the Data Link layer sends a frame, it waits for an acknowledgment from the recipient. As the recipient, the Data Link layer detects any problems with the frame that may have occurred during transmission. Frames that were not acknowledged, or frames that were damaged during transmission, are resent.

Network Layer

Layer 3, the *Network layer*, is responsible for addressing messages and translating logical addresses and names into physical addresses. This layer also determines the route from the source to the destination computer. It determines which path the data should take based on network conditions, priority of service, and other factors. It also manages traffic problems, such as switching, routing, and controlling the congestion of data packets, on the network.

The Network layer bundles small data frames together for transmission across the network. It also restructures large frames into smaller packets. On the receiving end, the Network layer reassembles the data packets into their original frame structure.

Transport Layer

The *Transport layer*, layer 4, takes care of error recognition and recovery. It also ensures reliable delivery of host messages originating at the Application layer. Similar to how the Network layer handles data frames, this layer repackages messages—dividing long messages into several packets and collecting small messages together in one packet—to provide for their efficient transmission over the network. At the receiving end, the Transport layer unpacks the messages, reassembles the original messages, and sends an acknowledgment of receipt.

Session Layer

The *Session layer* is the fifth layer of the OSI model. This layer allows two applications on different computers to establish, use, and end a connection called a *session*. This layer performs name recognition and the functions needed to allow two applications to communicate over the network, such as security functions.

The Session layer provides synchronization between user tasks by placing checkpoints in the data stream. This way, if the network fails, only the data after the last checkpoint has to be retransmitted. This layer also implements dialog control between communicating processes, regulating which side transmits, when, for how long, and so on.

Presentation Layer

The *Presentation layer* determines the form used to exchange data between networked computers. It can be called the network's translator. At the sending computer, this layer translates data from a format sent down from the Application layer into a commonly recognized, intermediary format. At the receiving end, this layer translates the intermediary format into a format useful to that computer's Application layer.

The Presentation layer also manages network security issues by providing services such as data encryption. It also provides rules for data transfer and provides data compression to reduce the number of bits that need to be transmitted.

Application Layer

The top layer of the OSI model is the *Application layer*. It serves as the window for application processes to access network services. This layer represents the services that directly support the user applications such as software for file transfers, for database access, and for electronic mail.

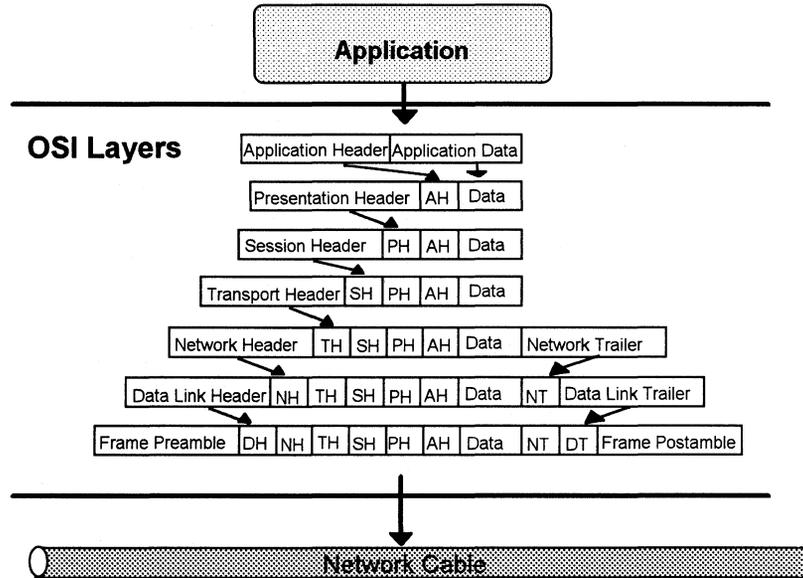
Sending Data Through the OSI Layers

As mentioned earlier, the OSI model regulates how data is passed from the end-user application to the networking cable. As data is passed from the application through the OSI layers, each layer wraps the data with layer-specific information. This information, in the form of headers and/or trailers, is read later by the corresponding layer on the receiving computer. The Physical layer adds header information called the *frame preamble* to the outside of the frame and adds trailer information called the *frame postamble* to the outside of the frame, but does not add to the data within the frame.

Passing through all seven layers of the OSI model, the data is wrapped with seven layers of information. Figure 1.4 illustrates how each OSI layer adds information to the data packet.

Figure 1.4

As data is sent down the OSI layers to the network, each layer wraps the data packet with its own header/trailer information.



When the data packet arrives at the destination computer, each layer “unwraps” the appropriate header and/or trailer, preamble, and postamble information. Each layer reads the information included by its corresponding layer on the sending computer and responds to it. Then, it passes the data packet to the next higher layer.

IEEE 802 Model

Another networking model was developed by the Institute of Electrical and Electronic Engineers (IEEE.) As a proliferation of local area network (LAN) products emerged, and with them a need for consistency, IEEE began to define LAN standards. Its project was called 802, for the year and month it began (February 1980).

Project 802 defined LAN standards for the Physical and Data Link layers of the OSI model. Although the published IEEE 802 standards actually predated the ISO standards, both were in development at roughly the same time and shared information, which resulted in two compatible models.

The 802 standards committee agreed with most of the OSI model but decided that more detail was needed at the Data Link layer. The 802 project divided the Data Link layer into two sublayers: the *Media Access Control* (MAC) and the *Logical Link Control* (LLC).

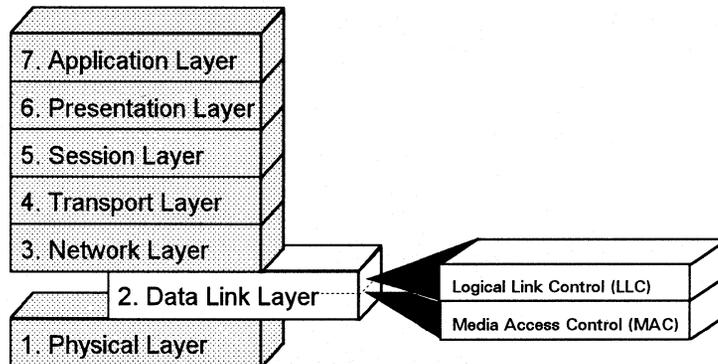
As Figure 1.5 indicates, the Media Access Control sublayer is the lower of the two sublayers, providing shared access for the computers' network adapter

cards to the Physical layer. The MAC layer communicates directly with the network adapter card and is responsible for delivering error-free data between two computers on the network.

The Logical Link Control sublayer, the upper sublayer, manages data link communication and defines the use of logical interface points (called Service Access Points—SAPs) that other computers can reference and use to transfer information from the LLC sublayer to the upper OSI layers.

Figure 1.5

IEEE Project 802 defined the Logical Link Control and Media Access Control sublayers.



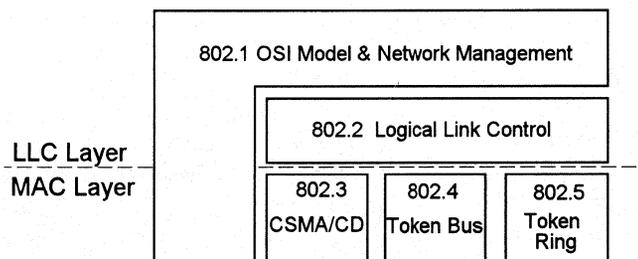
Project 802 resulted in a number of documents, including three key standards for network topologies:

- 802.3 defines standards for bus networks, such as Ethernet, that use a mechanism called Carrier Sense Multiple Access with Collision Detection (CSMA/CD).
- 802.4 defines standards for token-passing bus networks. (The ArcNet™ architecture is similar to this standard in many ways.)
- 802.5 defines standards for token-ring networks.

IEEE defined functionality for the LLC layer in standard 802.2 and defined functionality for the MAC and Physical layers in standards 802.3, 802.4, and 802.5.

Figure 1.6

Project 802 standards as related to LLC and MAC layers



The section in this chapter called “Popular LAN Architectures” describes three popular network types that are similar to those described by standards 802.3, 802.4, and 802.5.

The following section describes basic physical aspects of local area networks. This information is important as a background for a discussion of LAN architectures, including the one used by Windows for Workgroups.

Physical Characteristics of a LAN

At the Physical layer, each local area network has certain common characteristics. This section discusses LAN topologies, LAN cabling media, and transmission techniques used for LANs.

LAN Topologies

The way computers on a LAN are connected is called a *topology*. The three most widely used topologies are bus, star bus, and ring.

Bus Topology

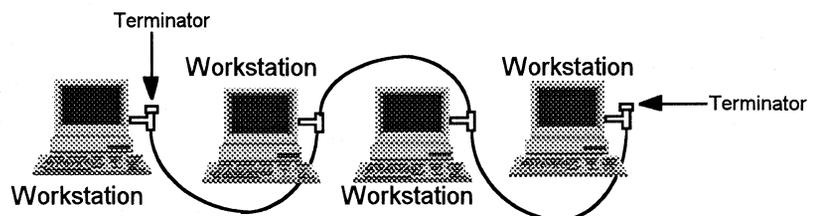
A bus topology connects each computer to a single cable. At each end of the cable is a terminating resistor, or a *terminator*. A signal is passed back and forth along the cable past the workstations and between the two terminators.

The “bus” carries a message from one end of the network to the other. As the bus passes each workstation, the workstation checks the destination address on the message. If the address in the message matches the workstation's address, the workstation receives the message. If the address doesn't match, the bus carries the message to the next workstation, and so on.

One type of bus network, called a local bus, is shown in Figure 1.7. A local bus uses a T-connector to connect the cable to the workstation's network adapter card. A terminator is connected to the last T-connector at each end of the network.

Figure 1.7

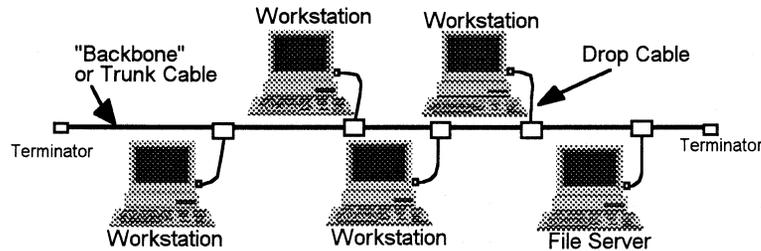
A local bus network connects the network cable directly to each computer using T-connectors.



A bus network, as shown in Figure 1.8, uses drop cables to connect each workstation to the main “backbone” cable.

Figure 1.8

A regular bus network uses drop cables to connect each computer to the network cable.



Bus topology is a passive network, meaning that workstations only listen for data being sent on the network and are not responsible for moving data from one workstation to the next. If one workstation fails, it doesn't affect the entire LAN. On the other hand, if the connection to one workstation comes loose, or if a cable breaks, the entire cable segment (the length between two terminators) loses its connectivity, causing the entire segment to be nonfunctional until it can be repaired.

The following table lists some advantages and disadvantages of bus networks.

Advantages	Disadvantages
Failure of a single workstation doesn't affect the entire LAN	Cable break can affect large number of users
Easy cable connections; flexible	Limited cable length and number of workstations
Inexpensive cable and connectors	Difficult to isolate network cabling errors
	Performance degradation is not graceful

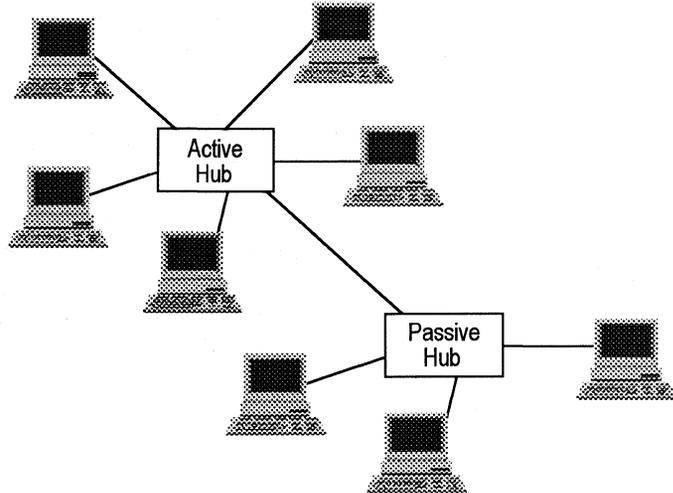
Star Bus Topology

In a network using a star configuration, each workstation is connected to a special unit called a *hub*. The hub provides a common connection so that all of the computers can communicate with one another.

Star bus topology uses signal splitters in the hub to send out signals in different directions on the cable connections. Both active and passive hubs are allowed. Active hubs can transmit a stronger signal to feed a longer cable and/or more signal splitters.

Figure 1.9

Each computer on a star network is connected to a hub. Hubs may be connected together to extend the network.



The following table lists some advantages and disadvantages of star bus topology.

Advantages

Easy to add new workstations

Central monitoring and network management

Disadvantages

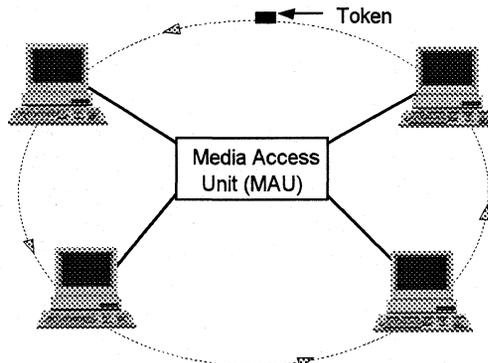
Failure of hub cripples the workstations connected to the hub

Token-Ring Topology

On a token-ring network, workstations are situated on a continuous network loop on which a "token" is passed from one workstation to the next. Although the name token ring implies a ring, the token ring is physically implemented as a star and electrically a ring. Workstations are centrally connected to a hub called a Media Access Unit (MAU) and are wired in a star configuration. Workstations use a token to transmit data and must wait for a free token in order to transfer messages.

Figure 1.10

A computer communicates on a token-ring network by "grabbing" the token and sending it around an electric ring.



The token contains the address of the sender and the address of the recipient. When the recipient has copied the information from the received message, it returns the token to the originating workstation to verify that the message was received. The original sender then passes the token to the next workstation in the ring so that workstation can send information over the network.

The following table lists some advantages and disadvantages of a token-ring topology.

Advantages	Disadvantages
Cable failures affect a small number of users	Costly wiring and connections
Equal access for all workstations	
Graceful performance degradation as the size of the network grows	

LAN Cabling Media

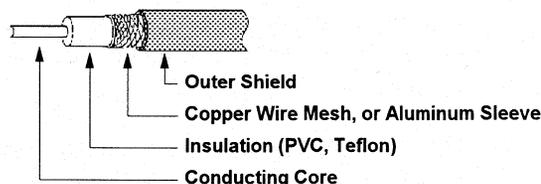
The LAN industry has standardized three primary physical media that can be used at the Physical layer: coaxial cable, twisted-pair cable, and fiber optic cable. Transmission rates that can be supported on each of these physical media are measured in *millions of bits per second*, or Mbps.

Coaxial Cable

Coaxial (or coax) cable is a conductive center wire surrounded by an insulating layer, a layer of wire mesh (shielding), and a non-conductive outer layer. Coaxial cable is resistant to interference and signal weakening that other cabling, such as unshielded twisted-pair cable, can experience. Coax is generally better than unshielded twisted-pair cable over longer distances and for reliably supporting higher data rates with less sophisticated equipment.

Figure 1.11

Coaxial cable



Coaxial cable for networking applications comes in varying types. The most popular are Thinnet and Thicknet.

Thinnet Coaxial Cable

Thinnet, or thin wire Ethernet, refers to RG-58 cabling and is a flexible coaxial cable about 1/4-inch thick. Thinnet is used for relatively short distance communication and is fairly flexible to facilitate routing between workstations. Thinnet connects directly to a workstation's network adapter card using a BNC T-connector and uses the network adapter card's internal *transceiver*. A transceiver is a device that receives and transmits signals.

Note The *Windows for Workgroups Starter Kit* uses Thinnet coaxial cable to interconnect workstations.

Thicknet Coaxial Cable

Thicknet, or Standard Ethernet, is a relatively rigid coaxial cable about 1/2-inch in diameter. Typically, Thicknet is used as a backbone to connect several smaller Thinnet-based networks because of its ability to support data transfer over longer distances. A transceiver is often connected directly to Thicknet cable using a connector known as a "piercing tap." Connection from the transceiver to the network adapter card is made using a drop cable (also called a transceiver cable) to connect to the Attachment Unit Interface (AUI) port connector.

Coaxial Cable Grades—Fire Codes

Coaxial cables come in two grades: PVC and plenum. The type of cable-grade that you should use to connect workstations depends on where the cables are going to be routed in your office.

PVC (poly-vinyl chloride) is a chemical used to construct the insulation and the cable jacket for most types of coaxial cable. PVC coaxial cable is flexible and can easily be routed in the exposed areas of an office.

A *plenum* is the short space in many buildings between the false ceiling and the floor above, and is used to circulate warm and cold air through the building. Fire codes are very specific on the type of wiring that can be routed through this area.

Plenum cabling refers to coax cable that contains special materials in its insulation and cable jacket certified to be fire-resistant and which produce a minimum amount of smoke in order to minimize poisonous chemical fumes. It can be used in the plenum area and in vertical runs (for example, in a wall) without conduit. Plenum cabling is not as flexible as PVC cable and is more expensive.

Important Please consult your local fire and electrical codes for specific regulations about running networking cable in your office.

The following table list some advantages and disadvantages of coaxial cabling.

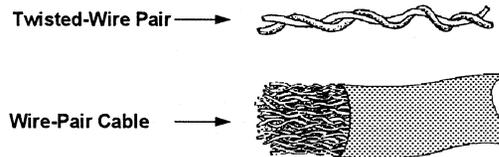
Advantages	Disadvantages
Low maintenance costs	Limited distance and topology
Simple to install and tap	Low security; easily tapped
Better resistance to signal noise over longer distances	Difficult to make major changes to the cabling topology
	Cable cost is higher than that of twisted pair, although electronic support components are less expensive

Twisted Pair

Twisted-pair cable consists of two insulated strands of copper wire twisted together. A number of twisted-wire pairs are often grouped together and enclosed in a protective sheath to form a cable. Unshielded twisted-pair wire is commonly used for telephone systems and is already installed in most office buildings.

Figure 1.12

Twisted-pair wire and cable



A shielded twisted-pair cable is less subject to electrical interference and supports higher transmission speeds over longer distances than unshielded twisted-pair cable.

The following table list some advantages and disadvantages of twisted-pair cabling.

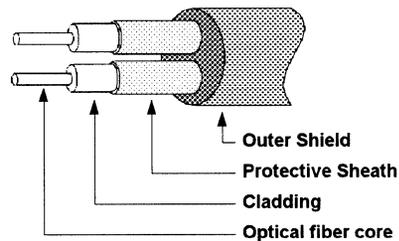
Advantages	Disadvantages
Well-understood technology	Susceptible to noise
Easy to add computers to network	Limited maximum bandwidth
Least expensive medium	Distance limitations
Same medium as telephones	Easiest to tap
Pre-existing phone wire may be in place to connect workstations	Requires expensive support electronics and devices

Optical Fiber

Optical fibers are used to carry digital data signals in the form of modulated pulses of light. An optical fiber consists of an extremely thin cylinder of glass, called the core, surrounded by a concentric layer of glass, known as the cladding.

Figure 1.13

Optical fiber cable



There are two fibers per cable—one to transmit and one to receive. Optical fiber cable transmissions are not subject to interference and are very fast (currently transmitting about 100 Mbps with demonstrated rates of up to 200,000 Mbps).

The following table shows some advantages and disadvantages of optical fiber cabling.

Advantages	Disadvantages
High bit rates	More expensive than other cabling types
Lowest transmission loss over longer distance	Cabling is inflexible; can't be bent sharply
Not subject to interference	Lack of standard components
Good for network "backbones"	Limited (practically) to high-traffic, point-to-point
Supports voice, data, video	Requires skilled installation and maintenance
Difficult to tap	High installation costs

Baseband and Broadband Transmission

Two techniques can be used to transmit the encoded signals over cable—baseband and broadband. *Baseband* transmission uses digital signaling and *broadband* transmission uses analog signaling.

Baseband Transmission

Baseband system's use digital signaling over a single frequency. Signals flow in the form of discrete pulses of electricity or light. With baseband transmission, the entire communication-channel capacity is used to transmit a single data signal.

As the signal travels along the network cable, it gradually decreases in strength and can become distorted. If the cable length is too long, resulting in a signal that is weak or distorted, the received signal may be unrecognizable or misinterpreted. As a safeguard, baseband systems sometimes use *repeaters* to receive an incoming signal and retransmit it at its original strength and definition to increase the practical length of a cable segment.

Broadband Transmission

Broadband systems use analog signaling and a range of frequencies. With analog transmission, the signals employed are continuous and non-discrete. Signals flow across the physical medium in the form of electromagnetic or optical waves.

Figure 1.14

Typical bandwidth allocation for broadband channels (mid-split configuration shown)

Transmit	Channel 1	252-258 MHz	▲
	Channel 2	258-264 MHz	
	Channel 3	264-270 MHz	
	Channel 4	270-276 MHz	
	Channel 5	276-282 MHz	
Guard Band		300 MHz	
Receive	Channel 5	84-90 MHz	 ▼
	Channel 4	78-84 MHz	
	Channel 3	72-78 MHz	
	Channel 2	66-72 MHz	
	Channel 1	60-66 MHz	

For baseband, a cable's total bandwidth is the difference between the highest and lowest frequencies that are carried over that cable. With broadband, if sufficient total bandwidth is available, multiple analog transmission systems (such as cable television and network transmissions) can be supported simultaneously on the same cable. Each transmission system is allocated a part of the total bandwidth. All devices associated with a given transmission system, such as all computers using a LAN cable, must then be tuned so that they use only the frequencies that are within the allocated range.

Native analog devices, such as some telephones and video equipment, can transmit their analog signal directly on the cable. However, digital systems, such as computers, require modems to translate the digital signal into an analog signal before transmission.

While baseband systems use repeaters, broadband systems use amplifiers to regenerate analog signals at their original strength.

With broadband transmission, signal flow is unidirectional. That is, in order for a signal to reach all devices, there must be two paths for data flow. There are two common ways to do this:

- Mid-split broadband configuration divides the bandwidth into two channels, each using a different frequency or range of frequencies. One channel is used to transmit signals, the other to receive signals.
- In dual-cable broadband configuration, each device is attached to two cables. One cable is used to send and the other to receive.

Popular LAN Architectures

In the previous sections, we discussed three physical components of a local area network—topology, cabling, and transmission technique. This section discusses how these components are used to implement some of today's most popular network architectures. LAN architecture, which refers to the overall design of a LAN, including the physical components, maps to the Physical layer of the OSI model. The architecture describes how to implement the networking concepts previously discussed.

Three popular LAN architectures used today are Ethernet, Token Ring, and ArcNet. This section describes the history and implementation requirements for each one.

Note The Windows for Workgroups Resource Kit concentrates on Ethernet. Ethernet is the most popular LAN architecture and is used for the Windows For Workgroups Starter Kit because of its low cost and ease of installation. This section also briefly discusses Token Ring and ArcNet networks to provide further understanding of LAN technology.

Ethernet

In the late 1960s, the University of Hawaii developed a *wide area network* called ALOHA. (A wide area network extends LAN technology across a larger geographical area.) The university needed to connect various computers that were spread throughout its campus. A key feature of that network design was called the Carrier-Sense Multiple Access with Collision Detection (CSMA/CD) access method. Carrier Sense means that the network card listens to the cable for a quiet period during which it can send messages. Multiple Access refers to multiple computers using the same network cable. Collision Detection is a safeguard against messages colliding in transit.

This early network design was the foundation for today's Ethernet. In 1972, Xerox® Corporation created Experimental Ethernet, and in 1975, it introduced the first Ethernet product. The original version of this networking product was designed as a 2.94-Mbps system to connect over 100 workstations on a 1-kilometer cable.

Xerox Ethernet was so successful that Xerox, Intel® Corporation, and Digital Equipment Corporation drew up a standard for a 10-Mbps Ethernet. This design became the basis for the IEEE's 802.3 specification. The Ethernet product conforms to most, but not to all, parts of 802.3.

Ethernet networks can be wired with different types of cable, each with its own benefits and drawbacks. Three popular specifications for Ethernet topologies are the following:

- The 10base2 specification (referred to in this document as Thinnet) refers to an Ethernet baseband LAN that uses Thinnet cabling. Thinnet coaxial cable can carry a signal up to approximately 185 meters (or about 607 feet). Beyond that, the signal must be regenerated by a repeater.
- The 10base5 specification (referred to in this document as Thicknet) is an Ethernet baseband LAN using Thicknet cabling. Thicknet can carry a signal for 500 meters (about 1640 feet) before needing a repeater.
- The 10baseT specification (referred to in this document as twisted pair) is an Ethernet baseband LAN using an unshielded twisted-pair (UTP) cable, commonly called *twisted pair*. This cabling medium can carry a message for 100 meters (about 328 feet) between a workstation and the hub that the workstation is connected to.

Note Network hardware components exist to easily connect the different Ethernet topologies discussed above, thus allowing a topology to be selected depending on the workgroup environment.

The following sections describe each of these three Ethernet topologies.

Thinnet (10Base2)

This network is called Thinnet, or thin wire, because of the thin, inexpensive coaxial cabling (RG-58) it uses. This cable can carry a signal for approximately 185 meters (about 607 feet). Beyond that, the signal must be regenerated by a repeater. Although there is a maximum length, there is also a minimum cable length. The cable between computers must be at least 0.5 meters (20 inches) long.

The reason this network is called 10Base2 by IEEE is because it transmits at 10 Mbps over a *baseband* wire and can carry a signal roughly 2 X 100 meters (actual distance is 185 meters).

Thinnet networks generally use a local bus topology, as described in “Bus Topology” earlier in this chapter. IEEE standards for Thinnet don't allow a drop cable to be used from the bus T-connector to a workstation. Instead, a T-connector fits directly on the network adapter card. A BNC barrel connector may be used to connect Thinnet cable segments together, thus extending a length of cable. For example, if you need a length of cable that is 30 feet long, but all you have is a 25 foot length and a 5 foot length of Thinnet cable, a BNC

barrel connector can be used to join the two cable segments together. The use of barrel connectors, however, should be kept to a minimum.

A Thinnet network was designed to be an economical way to support a small department or workgroup. The cable used for this type of network is relatively inexpensive. It is also easy to configure and to install.

A single Thinnet network can support a maximum of 30 nodes (workstations and repeaters) per cable segment. A *segment* is the length of cable between two terminators. A terminator must be connected to each end of a LAN in order for the network to function. Using repeaters, a Thinnet network can combine as many as five cable segments to support up to 150 computers (based on the IEEE 802.3 specification).

Note The Microsoft *Windows for Workgroups Starter Kit* contains a segment of Thinnet cable, two T-connectors, two terminators, and two Ethernet network adapter cards for connecting two workstations together. The Microsoft *Windows for Workgroups User Kit* contains components that will allow you to add one additional workstation to your Windows for Workgroups network (one segment of Thinnet cable, one T-connector, and one Ethernet network adapter card).

Thicknet (10Base5)

Although IEEE calls this 10Base5—referring to its main specifications of 10Mbps, baseband, and 500-meter (5 X 100) segments—it is commonly known as Thicknet for the cabling it uses. (It is also called Standard Ethernet.)

Thicknet generally uses a bus topology and can support as many as 100 nodes (workstations, repeaters, and bridges) per backbone segment. The backbone, or trunk segment, is the main cable from which drop cables are connected to workstations, repeaters, and bridges. Unlike Thinnet, this type of network uses drop cables connected to external transceivers to connect workstation network adapter cards to a cable segment.

Thicknet cable transmits at a rate of 10 Mbps and can carry a signal for 500 meters (1640 feet), excluding drop cables, before needing a repeater. A maximum of five backbone segments can be connected using repeaters (based on the IEEE 802.3 specification). The length of drop cables is not used to measure the distance supported on the Thicknet cable; only the end-to-end length of the Thicknet cable segment itself is used.

The minimum cable length between connections (or taps) on the Thicknet cable segment, excluding drop cables, is 2.5 meters (about 8 feet).

Drop cables are typically 3/8-inch shielded-pair cable and can be as long as 50 meters (about 164 feet). Drop cables are connected by AUI (Attachment Unit Interface) connectors on the network adapter card at one end and the main backbone cable at the other (through the use of an external transceiver).

Thicknet was designed to support a backbone for a large department or even all the workstations in an entire building.

Twisted-Pair (10BaseT)

10BaseT is an Ethernet baseband LAN that uses unshielded twisted-pair cable to connect workstations. Most networks of this type are configured in a star pattern but internally use a bus signaling system like other Ethernet configurations.

Typically, the hub of a 10BaseT network serves as a repeater and often is located in a work closet of the building (for example, in a phone closet). Each workstation or server is located at the end-point of a cable connected to the hub. Each computer has two pairs of wire—one pair is used to receive data and one pair is used to send data. Wires can be up to 100 meters (about 328 feet) from the repeater to computer. The minimum cable length between computers is 2.5 meters (about 8 feet).

Token Ring

A token-ring network is an implementation of IEEE standard 802.5, the standard for the token-ring access method. The access method, more than the ring shape, distinguishes token-ring networks from other LANs.

As described earlier in this chapter, a token is passed around the ring to which the computers are connected. The computer that grabs the token is able to send data over the network. When the token returns to its source with an acknowledgment from the destination, the source relinquishes the token and continues passing it around the ring, allowing another workstation to send data out on the network. (By contrast, the other common access method for LANs, the CSMA/CD access method, is described in the previous section on Ethernet.)

The first design of a network passing a token ring is attributed to E. E. Newhall in 1969. IBM first publicly supported a token-ring topology in March 1982, when the company presented papers for the IEEE Project 802.

IBM announced a token-ring network product in 1984, and in 1985, the IBM token ring became an ANSI/IEEE standard. The goal was to allow for a simple wiring structure using twisted-pair cable that connected a workstation to the network by a wall socket, with the main wiring in a centralized location.

Data on the ring is transmitted at either 4 or 16 Mbps, depending on the actual implementation. In order for workstations to communicate with each other, all network cards must be configured alike to communicate at either 4 or 16 Mbps on the network.

Network computers are connected by shielded and/or unshielded twisted-pair cable to a wiring concentrator. Each computer can be up to 100 meters (about 328 feet) from the Media Access Unit (MAU) using shielded wire, or 45 meters (about 148 feet) using unshielded wire. The minimum shielded or unshielded cable length is 2.5 meters (about 8 feet).

Each MAU can support as many as 72 workstations that use unshielded wire, or up to 260 workstations using shielded wire. Each ring can have as many as 33 MAUs.

A token ring is an efficient design for moving data on a network. On small- to medium-sized networks with heavy data traffic, token rings are more efficient compared to most Ethernet installations. On the other hand, the direct routing of data supported by Ethernet networks tends to fair better when a network includes a large number of computers with light to moderate traffic.

ArcNet

The Attached Resource Computer Network (ArcNet) was developed by Datapoint Corporation in 1977. It was designed as a token-passing bus architecture, transmitting at 2.5Mbps. A successor to the original ArcNet, ArcNetplus, supports data transmission rates of 20 Mbps.

ArcNet technology predates IEEE Project 802 standards, but most closely maps to 802.4, the standards for token-passing bus networks using broadband cable. ArcNet, however, is a baseband network and can have a star or bus topology.

ArcNet is an easy-to-install and inexpensive network. It typically uses coaxial cable and includes both active and passive hubs. Active hubs regenerate signals. Passive hubs split strong signals into multiple, weaker signals. Each workstation is connected by cable to a hub. The maximum cable length is from 120 meters (about 393 feet) to 606 meters (about 1988 feet), depending on the type of cable and hub connection.

Summary of Popular LAN Architecture Specifications

The following is a summary of the specifications for the popular architectures discussed in the previous sections.

Note The following information is the minimum standards required to conform to IEEE specifications. A particular implementation of the network architecture may differ from the information in this table.

	Ethernet (IEEE 802.3)			Token Ring	ArcNet
	Thinnet (10Base2)	Thicknet (10Base5)	Twisted-pair (10BaseT)		
Topology	Local bus	Bus	Star	Ring/Star	Series of stars
Cable type	RG-58	Thicknet; 3/8" shielded-pair drop cable	Unshielded twisted-pair	Shielded or unshielded twisted pair	RG-62 or RG-59
Impedance	50 Ω	50 Ω			
Terminator resistance	50 $\Omega \pm 2 \Omega$	50 $\Omega \pm 2 \Omega$	85-115 Ω UTP; 135-165 Ω TP	100-120 Ω UTP, 150 Ω TP	RG-59: 75 Ω RG-62: 93 Ω
Maximum cable segment length	185m (about 606 feet)	500m (about 1640 feet)	100m (about 328 feet)	From 45m to 200m (about 148 to 656 feet), depends on cable type	Depends on cable type, but <i>on average</i> : <i>W-W</i> : 120m (393 feet); <i>A-A</i> : 606m (1988 feet); <i>P-W</i> or <i>P-A</i> : 30m (98 feet) <i>A-A</i> : 0.3m (about 12 inches) jumper ¹
Minimum length between computers	0.5m (about 23 inches)	2.5m (about 8 feet)	2.5m (about 8 feet) between repeater hub and computer	2.5m (about 8 feet)	Depends on cable
Maximum connected segments	5	5	Star pattern doesn't support connected segments	33 Media Access Units (MAUs)	Doesn't support connected segments
Maximum computers per segment	30	100	N/A	Unshielded: 72 workstations per concentrator; Shielded: 260 workstations per concentrator	Depends on cable used

¹ *W-W* = workstation to workstation; *A-A* = active hub to active hub; *P-W* = passive hub to workstation; *P-A* = passive hub to active hub.

Network Adapter Cards

In the preceding section, we discussed Ethernet and its use of three cabling media—Thinnet, Thicknet, and twisted-pair. Each of these cabling media have a different connector to fit a different network adapter card. The *network adapter card* (sometimes called a network interface card or NIC) is installed in each computer on the LAN and includes a connector port on the back of the network card for connecting to a cabling medium.

Besides cabling, network adapter cards comprise the other key portion of the network's Physical layer. Network adapter cards do the main work of the LAN. They move data from the local computer to the network, and vice versa, by translating digital PC signals into analog RF signals, if necessary, for the network's cables. They also assemble data into packets for transmission and verify source-to-destination transmissions.

Ethernet Cable Media Connector Types

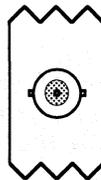
If you are unfamiliar with connector types used for connecting to the most popular cabling media, the following information will help you to easily identify the connector type on the back of your network adapter card.

Some network adapter cards have more than one interface connector. For example, it is not uncommon for a network adapter card to have both a Thinnet and Thicknet connector, or a twisted-pair and a Thicknet connector. If this is true in your case, the selection of which interface connector to use will be controlled either by setting jumpers or DIP switches on the card itself or by using a software-selectable option. Consult the documentation that came with your network adapter card for information on how to properly configure the card.

A Thinnet network connection uses a coaxial BNC connector as shown in Figure 1.15:

Figure 1.15

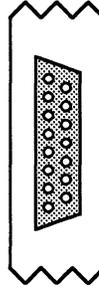
Coaxial BNC
connector for Thinnet



A Thicknet network connection uses a 15-pin Attachment Unit Interface (AUI) cable (as shown in Figure 1.16) to connect the 15-pin (DB15) connector on the back of the network adapter card to an external transceiver.

Figure 1.16

A 15-pin AUI
connector for
Thicknet

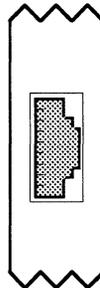


Note Be careful not to confuse a joystick port with an AUI network adapter port, as they both look alike.

A twisted-pair Ethernet network (10BaseT) connection uses an RJ-45 connector, as shown in Figure 1.17. The RJ-45 connector is similar to an RJ-11 telephone connector but has twice the number of conductors.

Figure 1.17

RJ-45 connector for
twisted-pair Ethernet



Configuration Options

Network adapter cards generally have three configurable options that must be set properly for the network adapter card to function in your computer: the interrupt (IRQ), the base I/O port address, and the base memory address. These settings may be changed by using the Network option in the Windows Control Panel.

Note It is necessary to match the network adapter card settings that you specify in the Network option of the Windows Control Panel with the jumper or DIP-switch settings configured on your network adapter card.

Interrupt (IRQ)

When your network card sends a request to your computer, it uses an *interrupt*—an electronic signal sent to the computer's central processing unit (CPU). Each device in your system must use a different *interrupt request line (IRQ)*, which is specified when the device is configured. (Interrupt is often used interchangeably with IRQ.)

In most cases, you can use IRQ3 or IRQ5 for your network card. IRQ5 is the recommended setting if it is available. Check the documentation that came with your computer and the other devices installed in your system to determine which IRQs are already being used.

If neither IRQ3 nor IRQ5 is available, refer to the following table for alternate values you may be able to use. The IRQs listed as “available” usually can be used for your network card. If your computer doesn't have the hardware listed for a specific IRQ, that IRQ should be available.

IRQ Computer with 80286 processor (or higher)

2 (9)	EGA/VGA (enhanced graphics adapter/video graphics adapter)
3	Available (unless used for second serial port [COM2, COM4] or bus mouse)
4	COM1, COM3
5	Available (unless used for second parallel port [LPT2])
6	Floppy-disk controller
7	Parallel port (LPT1)
8	Real-time clock
10	Available
11	Available
12	Mouse (PS/2)
13	Math coprocessor
14	Hard-disk controller
15	Available

Base I/O Port

The *base input/output (I/O) port* specifies a channel through which information is transferred between your computer's hardware (such as your network card) and its CPU. The port appears to the CPU as an address.

Each hardware device included in your system must have a different base I/O port number. The port numbers (in hexadecimal format) in the following table are usually available for you to assign to your network card. Those with a device listed next to them are addresses commonly used for the devices. Check the documentation that came with your computer and other devices to determine which addresses are already in use.

Port	Device	Port	Device
200 to 20F	Game port	300 to 30F	
210 to 21F		310 to 31F	
220 to 22F		320 to 32F	Hard-disk controller (for PS/2 Model 30)
230 to 23F	Bus mouse	330 to 33F	
240 to 24F		340 to 34F	
250 to 25F		350 to 35F	
260 to 26F		360 to 36F	
270 to 27F	LPT3	370 to 37F	LPT2
280 to 28F		380 to 38F	
290 to 29F		390 to 39F	
2A0 to 2AF		3A0 to 3AF	
2B0 to 2BF		3B0 to 3BF	LPT1
2C0 to 2CF		3C0 to 3CF	EGA/VGA
2D0 to 2DF		3D0 to 3DF	CGA/MCGA (also EGA/VGA, in color video modes)
2E0 to 2EF		3E0 to 3EF	
2F0 to 2FF	COM2	3F0 to 3FF	Floppy-disk controller; COM1

Base Memory Address

The *base memory address* defines the address of the location in your computer's memory (RAM) that will be used by the network card to exchange information between your computer and the other computers you are connected to. This setting is sometimes called the *RAM start address*. Often, the base memory address for your network card is D8000. (For some network cards, the final "0" is dropped from the base memory address—for example, D800.)

Note Some network cards don't have a setting for the base memory address because they don't use any RAM.

With some network cards, you may also need another setting that specifies the amount of memory to be "mapped" or set aside for this purpose. For example, for some cards you can specify either 16K, which leaves more memory available for other uses, or 32K, which provides better performance but leaves less memory available.

Additional Settings

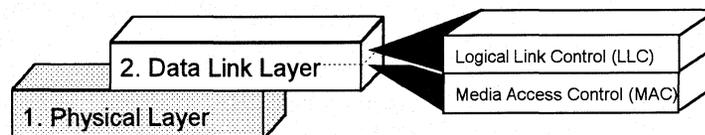
Your network card may have other settings you need to define during configuration. For example, some cards have more than one connector (where you connect the cable), providing you with flexibility in choosing a cable. If your card has multiple connectors, you usually need to specify which connector to use. This may be done by using a jumper or a software setting. See your network-card documentation for details about this and other settings.

Network Device Interface Specification (NDIS)

The second layer of the OSI model is the Data Link layer. As described earlier in this chapter, the Data Link layer is divided into two sublayers by the IEEE Project 802—Media Access Control (MAC) and Logical Link Control (LLC).

Figure 1.18

The Data Link layer is divided into LLC and MAC sublayers.



Device drivers at these sublayers move data received in the Physical layer by the network adapter card and pass it up the other OSI layers to the destination application.

A *MAC driver* is the device driver located at the MAC sublayer. It provides low-level access to network adapters by providing data transmission support

and some basic adapter management functions. MAC drivers are also known as network adapter card drivers.

Until the late 1980s, many of the implementations of transport protocols—that is, the lower layer protocols that reside on layers 3 and 4 of the OSI model and are used by networked computers to communicate with each other—were tied to a proprietary implementation of a MAC-layer interface defining how the protocol would converse with the network adapter card. This made it difficult for network adapter card vendors to support the different network operating systems available on the market. Each network adapter card vendor had to create proprietary interface drivers to support a variety of protocol implementations for use with several network operating system environments.

In 1989, Microsoft and 3Com® jointly developed a standard defining an interface for communication between the MAC layer and the transport protocol (or simply protocol) drivers that reside on layers 3 and 4 of the OSI model. This standard is known as the *Network Device Interface Specification (NDIS)*. NDIS allows for a flexible environment of data exchange. It defines the software interface—called the NDIS interface—used by network transport protocols to communicate with the network adapter card.

The flexibility of NDIS comes from the standardized implementation used by the network industry. Any NDIS-conformant protocol can pass data to any NDIS-conformant MAC driver, and vice versa. A process called *binding* is used to establish the initial communication channel between the protocol driver and the MAC driver. A software driver called the *Protocol Manager*, as defined by NDIS, is used to bind the MAC and transport protocol drivers. Transport protocol drivers are described in the next section and reside above the MAC-driver layer in the OSI model.

Version 2.0.1 of the NDIS, as used by Windows for Workgroups, allows a single computer to have as many as four network adapter cards installed in it. Each network adapter card can support a maximum of four protocols, with no more than eight transport protocols supported on a computer.

The advantage of supporting multiple protocol drivers on a single network card is that Windows for Workgroups workstations can have simultaneous access to different types of network servers, each using a different transport protocol. For example, a Windows for Workgroups workstation can have access to both a Microsoft LAN Manager server and a Novell NetWare server simultaneously.

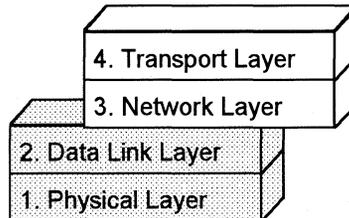
The Protocol Manager arbitrates among the multiple protocol and MAC drivers, routing incoming network requests to the appropriate driver.

Network Protocols and Transports

The following section discusses components that reside on the Network and Transport layers.

Figure 1.19

Network and Transport layers of the OSI model.



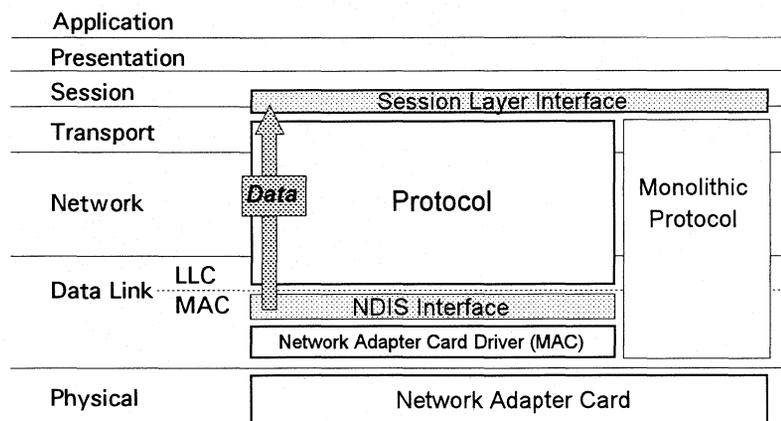
To briefly summarize, the packaging and routing of data and application messages occurs on these layers, by a component called a *protocol*.

Sometimes a separate component called a *transport* manages application messages while the protocol manages data. In other cases, as with Microsoft's NetBEUI protocol, both of these functions are handled by the protocol. A protocol driver that manages all functions of the MAC through the Transport layer in one protocol driver is referred to as a *monolithic protocol stack*—monolithic protocols do not need to bind to a separate MAC driver due to the internal network adapter card (MAC) driver. Network transports that conform with NDIS require an NDIS-compliant network adapter card driver to provide support for MAC layer communication.

The protocol is the network data organizer. It defines how data should be presented to the next receiving layer and packages the data accordingly. It passes data to the MAC driver through the NDIS interface at the Data Link layer, and to the application through the Session layer interface located at the Session layer, as shown in Figure 1.20.

Figure 1.20

The protocol is the network's data organizer. It sends data between the Session layer and NDIS interfaces.



Now that we've established the relationship between a network protocol and the NDIS MAC-layer driver, let's discuss two key protocols supported by Windows for Workgroups, NetBEUI and MSIPX.

NetBEUI

The primary protocol used by Windows for Workgroups is called NetBEUI (NetBIOS Extended User Interface). This protocol was first introduced by IBM in 1985. NetBEUI is a small and efficient protocol designed for use on a departmental LAN of 20 to 200 workstations. The original design assumed broader connectivity services could be added as the network grew by including gateways. A *gateway* is a device that allows one type of network to communicate with a different type of network.

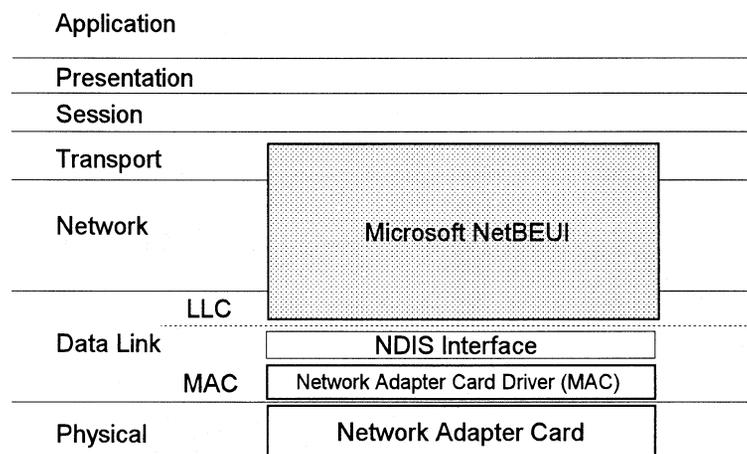
Microsoft has supported the NetBEUI protocol in all of its networking products since Microsoft's first networking product, MS-Net, was introduced in the mid-1980s.

Microsoft supports NetBEUI for several reasons. It has powerful flow control and tuning parameters and has robust error detection. It conforms to IBM's NetBEUI specifications and thus can communicate with IBM's NetBEUI protocol. It includes performance enhancements related to NetBIOS 3.0 and supports high data throughput rates. Finally, its small and efficient design is perfect for workgroup networking.

Figure 1.21 shows how NetBEUI fits into the OSI model.

Figure 1.21

Windows for Workgroups components shown in reference to the seven-layer OSI model



Dual-Net/Multiple Network Support

Through the use of the Network Device Interface Specification (NDIS), Windows for Workgroups supports the use of multiple protocol stacks on a single workstation, so long as each protocol stack complies with NDIS version 2.0.1.

Windows for Workgroups provides the necessary components to support loading both the Microsoft NetBEUI protocol and the Novell IPX protocol on a Windows for Workgroups workstation. Loading both protocols on a single workstation enables that workstation to connect to NetBEUI-compliant network servers (for example, Microsoft LAN Manager) and Novell NetWare servers.

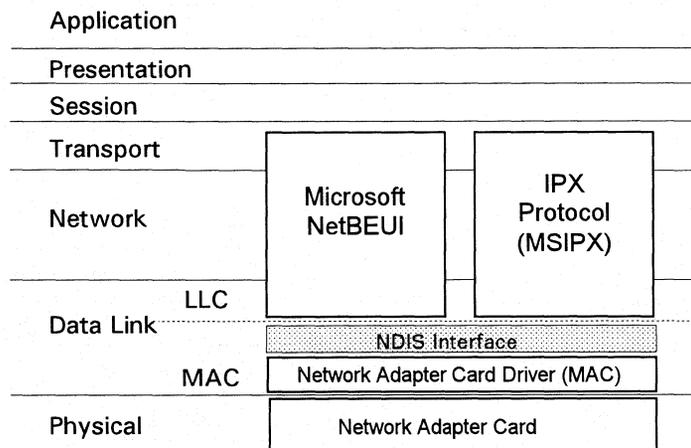
Configuring Windows for Workgroups to support a dual-net environment can occur either during the setup process or after Windows for Workgroups has been installed by using the interface provided in the Network option of the Windows Control Panel.

In order to support the dual-protocol configuration, Microsoft provides an NDIS-compliant IPX protocol stack called MSIPX. (The MSIPX protocol stack is briefly described in the next section and is discussed in-depth in Chapter 8, “Network Integration with Microsoft LAN Manager and Novell NetWare.”)

Figure 1.22 shows an example of how a computer configured with both the NetBEUI and MSIPX transport protocols would appear.

Figure 1.22

Windows for Workgroups supports up to four protocol stacks.



MSIPX and NetWare Protocols

Windows for Workgroups provides NetWare connectivity by including a version of Novell's IPX protocol, called MSIPX, that supports NDIS-compliant network drivers.

The Novell IPX (Internetwork Packet Exchange) protocol provides connectionless communication between computers. The commonly used Novell IPX implementation is a monolithic stack (where the network adapter card driver is internal to the protocol stack). Each workstation's copy of IPX.COM protocol driver is customized to suit the network card, interrupts, memory, and other characteristics of that workstation. The IPX must be modified each time the workstation's configuration changes, and IPX.COM can't be copied from one workstation to the next unless the workstations are configured exactly the same.

In order for Windows for Workgroups to support the NetBEUI and IPX protocols on the same network adapter card, an NDIS-compliant version of the IPX protocol stack is needed. The MSIPX protocol stack is a full implementation of the IPX protocol, but one that is NDIS-compliant and thus can also take advantage of the many available NDIS network adapter card drivers.

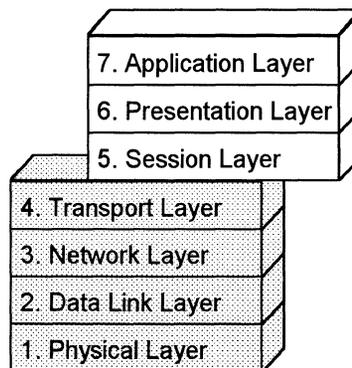
For more information about implementation and integration of the MSIPX protocol, see Chapter 8, "Network Integration with Microsoft LAN Manager and Novell NetWare."

Communication Between Network Transports and Applications

In this section, we conclude our discussion of the OSI model by identifying Windows for Workgroups components that run on the Session, Presentation, and Application layers.

Figure 1.23

*Session,
Presentation, and
Application layers of
the OSI model*



These three layers support communication between applications on different computers by creating sessions, defining data-exchange formats, and providing application-support services.

NetBIOS

NetBIOS is the Network Basic Input/Output System—a high-level interface used by applications to communicate with NetBIOS-compliant transports such as NetBEUI. The network redirector, discussed in the next section, is an example of a NetBIOS application. The NetBIOS interface is responsible for establishing logical names on the network, establishing connections (called *sessions*) between two logical names on the network, and supporting reliable data transfer between computers that have established a session.

This Session-layer interface was developed by Sytek, Inc., for IBM's broadband computer network. At that time, NetBIOS was included on a ROM chip on the network adapter card. Sytek also developed a NetBIOS for IBM's token-ring network, this time implemented as a device driver. Several other vendors have since produced versions of this interface.

In order to support the emerging network industry standard, Microsoft developed the NetBIOS interface for MS-Net and LAN Manager products. This interface is included with the Windows for Workgroups product.

NetBIOS uses a unique logical name to identify a workstation for handling communications between nodes. A NetBIOS name is a unique alphanumeric name consisting of one to 15 characters (no other workstation on the network may have the same name as another computer or workgroup). Windows for Workgroups prompts the user to enter the name for his/her computer at the time the user runs Setup (referred to as the *computer name*). The *computer name* for the workstation may be changed by using the Network option of the Windows Control Panel (the *computer name* is defined in the **[network]** section of the SYSTEM.INI file—see Chapter 6 “Windows for Workgroups Initialization Files,” for more information).

Communication between networked computers can also be in the form of a conversation. To carry on two-way communication between computers, NetBIOS establishes a logical connection, or *session*, between them. Once a logical connection is established, computers can then exchange data in the form of NetBIOS requests or in the form of a Server Message Block (SMBs).

The next section explains how the redirector uses the NetBIOS interface to communicate with other workstations and servers on the network. It also describes how the redirector uses SMBs to send data requests to remote devices.

The Network Workstation Services

Three classes of services are provided by the Windows for Workgroups network workstation software (the version of this software for MS-DOS is called the Microsoft *Workgroup Connection*): redirector, receiver, and server. A fourth

class of service, the messenger service, is supported by the Microsoft LAN Manager network client software, but is not supported by the Windows for Workgroups network software. See Chapter 8 for more information.

The services supported by the workstation software rely on the use of the Server Message Block (SMB) protocol to communicate with other workstations through the NetBIOS interface. The SMB protocol operates at the Application layer of the OSI model and was jointly developed by Microsoft, Intel, and IBM.

The network redirector is a NetBIOS application that translates requests from logical devices and routes them to the appropriate destination on the network. The redirector intercepts DOS file I/O calls (DOS INT 21h) and determines whether the request is for a local physical device or a logical remote device. If the device is a local device, the redirector passes the request to MS-DOS. If the request is for a remote device, the redirector translates the request into an SMB protocol message.

SMB protocol defines a series of commands used to pass information between workstations and can be broken into four message types: session (connection) control, file, printer, and message. Session control consists of commands that start and end a redirector connection to a shared resource at the server. The file SMB messages are used by the redirector to gain access to files at the server. The printer SMB messages are used by the redirector to send data to a print queue at a server and to get status information about the print queue. The message SMB type allows an application to send messages to or receive messages from another workstation.

The redirector packages SMB requests meant for remote workstations in a structure known as a Network Control Block (NCB). NCBs can be sent over the network to a remote device. The redirector also uses NCBs to make requests to the protocol stack of the local computer, such as "Create a session with the file server."

The receiver workstation service provides the support for a workstation to listen for SMB messages destined for it and removes the data portion of the SMB request in order for it to be processed by a local device. The server workstation service implements the full set of SMB protocols and manages local devices for shared use by other computers on the network.

Because Windows for Workgroups supports the SMB protocol, a Windows for Workgroups workstation can communicate with any other SMB-compliant workstation or server running a compatible transport protocol (for example, NetBEUI), both as a server and as a workstation. Other SMB-compliant network operating systems include Microsoft LAN Manager, Microsoft LAN Manager for UNIX, Microsoft MS-Net, Microsoft Windows NT, 3Com 3+Open™, DEC Pathworks™, and IBM LAN Server. See Chapter 8, "Network Integration with Microsoft LAN Manager and Novell NetWare" for more information.

Servers and Workstations

This section discusses the role of servers on a network and describes two models of network security.

In local area network architecture, a computer that provides shared resources to other computers on the network is called the *server*. A server shares resources, such as data files and printers, by making the resources available for use by other workstations on the network. Through the server, the network administrator grants other computers access to specific resources, such as a directory on the server's hard disk, a CD-ROM player connected to the server, or a printer connected locally to the server.

Network Resource Management

Part of a network's overall design includes how network resources will be shared and managed. Early networks used central file servers to manage access to shared files and printers. These servers were quite frequently workstations dedicated for the purpose of sharing resources or running network file server-related applications (for example, database engines). Examples of network operating systems (NOS) that support the central file server model include Microsoft LAN Manager and Novell NetWare. Lately, the idea of peer-to-peer networking has been introduced. Peer-to-peer networking means that network workstations can both use resources shared by another user and share their computer's resources with other workgroup members. Windows for Workgroups supports the peer-to-peer networking model.

The central file server model requires that all file sharing take place on the LAN server itself. If the file server is the central repository of information, then all exchanges of information between workstations on a LAN must first pass through the file server. This model is commonly implemented in an environment in which the workstations don't have the processing power or the resources capable of supporting networking server functionality. For example, the workstation may not have enough memory or may have a small-capacity hard disk.

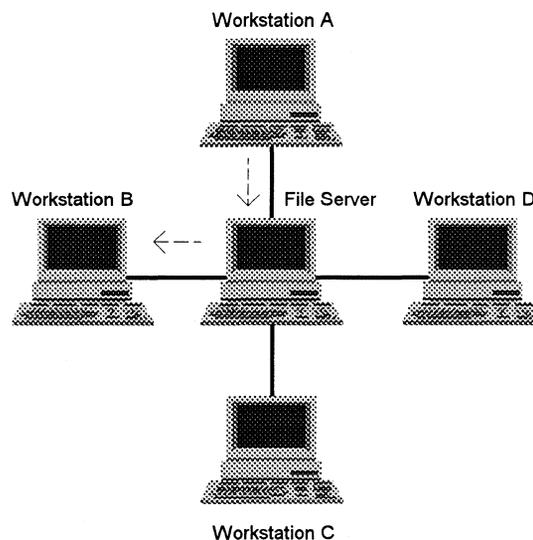
In the peer-to-peer networking model, each workstation (or at least one workstation) in a file transfer scenario acts as both a client and a server. As a client, the workstation can access the network resources shared by another workstation. This model is commonly implemented in an environment in which workstations have the processing power and resources to provide network server functionality, or when the cost is prohibitively high to dedicate a workstation as a server. This model is more flexible than the central file server model for a workstation user because the user is able to directly access shared resources on another computer.

To illustrate the differences between the central file server model and the peer-to-peer networking model, let's discuss a sample scenario. Two workstations, Workstation A and Workstation B, want to share a document that Workstation A has created. The document resides on the hard disk of Workstation A.

In the central file server model, Workstation A must connect to the server, place the document on the server's hard disk, then inform Workstation B that the document is available. Workstation B can then connect to the server and use the document left there by Workstation A. Unless Workstation A is working with the specified document directly on the file server (that is, making modifications to the version of the document that resides on the file server), the version of the document that may reside on Workstation A's hard disk will likely become out of sync with the version of the document Workstation B is now using. Figure 1.24 illustrates the role of the central file server in this sample scenario.

Figure 1.24

The role of a file server in the central file server model

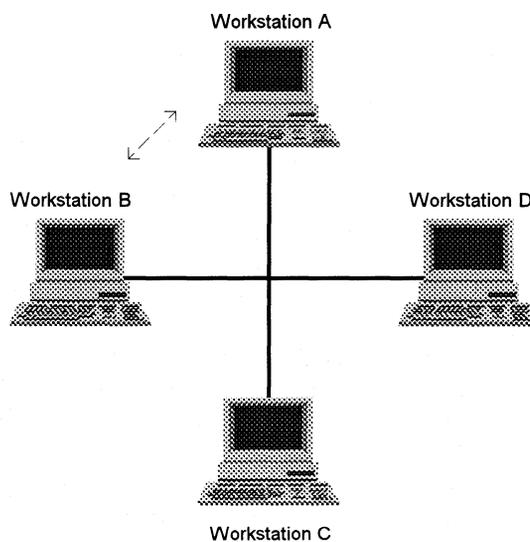


In the peer-to-peer networking model, Workstation A may simply grant Workstation B permission to access the directory where the files are located on Workstation A's local hard disk. Workstation B can then directly connect to Workstation A and use the document. Another way to provide Workstation B with the document that Workstation A created is to have Workstation B give Workstation A access to a shared directory on Workstation B. Then Workstation A can copy the document from its hard disk to the shared directory on Workstation B. The latter procedure however, suffers from the same difficulty as the central file server model because the versions of the documents may get out of sync. Because both workstations are considered to be peers, both support client and server functionality.

Figure 1.25 illustrates the role of Workstation A and Workstation B in this sample scenario.

Figure 1.25

The role of a workstation in the peer-to-peer networking model



LAN Security

There are two primary security models for granting permission to allow workstations access to shared resources—user-level security and share-level security.

User-level security involves assigning certain rights on a user-by-user basis. A user types a password when he or she logs on to the network. The server validates this user name/password combination and uses it to allow or disallow access to shared resources by checking access to the resource against a user-access database resident on the server. Microsoft LAN Manager supports both user-level and share-level security and Novell NetWare supports only user-level security.

Share-level security involves assigning a password to each shared resource. Access to the shared resource is granted to a user if he or she enters the appropriate password when attempting to access the shared resource. Because of its simplicity, Windows for Workgroups uses share-level security.

Additional Reading

For additional information on topics discussed in this chapter, see the following resources:

ANSI/IEEE standard 802.2 - 1985 (ISO/DIS 8802/2): *IEEE Standards for Local Area Networks—Logical Link Control Standard*.

ANSI/IEEE standard 802.3 - 1985 (ISO/DIS 8802/3): *IEEE Standards for Local Area Networks—Carrier Sense Multiple Access with Collision Detection (CSMA/CD) Access Method and Physical Layer Specifications*; American National Standards Institute; January 12, 1989.

ANSI/IEEE standard 802.4 - 1985 (ISO/DIS 8802/4): *IEEE Standards for Local Area Networks—Token-Passing Bus Access Method and Physical Layer Specifications*; American National Standards Institute; December 17, 1984.

ANSI/IEEE standard 802.5 - 1985 (ISO/DIS 8802/5): *IEEE Standards for Local Area Networks—Token-Ring Access Method and Physical Layer Specifications*; American National Standards Institute; June 2, 1989.

The Ethernet. A Local Area Network. (Data Link Layer and Physical Layer Specifications); version 2.0, November 1982. Also known as the “Ethernet Blue Book.”

Haugdahl, J. Scott. *Inside NetBIOS*. Minneapolis: Architecture Technology Corporation, 1990.

IBM Token-Ring Network PC Adapter Technical Reference (69X7830).

IBM Token-Ring Network Architecture Reference (6165877), November 1985.

International Standard 7498: *Information processing systems—Open Systems Interconnection—Basic Reference Model (First edition)*; American National Standards Institute, November 15, 1984. The OSI model.

Microsoft Corporation. *Microsoft LAN Manager Resource Kit*. Microsoft Corporation, 1992.

Microsoft Corporation, 3Com Corporation. *SMB Specification*. This may be obtained from the files library in the Microsoft Client Server Computing forum on CompuServe (GO MSNETWORK).

**Chapter
2**

Windows for Workgroups Architecture

This chapter provides an overview of the architecture used by Windows for Workgroups version 3.1. Windows for Workgroups builds on the architecture used by Windows version 3.1 to provide integrated peer-to-peer networking functionality.

Related information

- *Windows Resource Kit*: Chapter 5, “Windows 3.1 and Memory Management.”

Contents of This Chapter

Review of Windows 3.1 Architecture	2-2
Windows Components	2-2
Windows Dynamic Link Libraries (DLLs)	2-4
Windows Operating Modes	2-5
Windows for Workgroups 3.1 Architecture	2-11
Windows for Workgroups Network Drivers	2-11
Standard Mode	2-13
386 Enhanced Mode	2-15

Review of Windows 3.1 Architecture

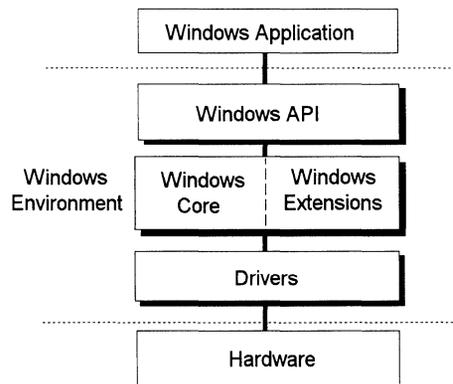
In order to discuss the architecture used by Windows for Workgroups, we'll first begin by reviewing the architecture used by Windows 3.1. The concepts presented in this section will serve as the basis for the discussion of the components that make up Windows for Workgroups.

Windows Components

Generally, Windows architecture can be divided into three primary components, as illustrated in Figure 2.1: the Windows Application Programming Interface (API), the Windows core and Windows extensions, and the Windows drivers.

Figure 2.1

Generic Windows architecture



The goal of the Windows architecture is to abstract the components in such a way that software developers need only understand the published behavior of the exposed, programming interfaces. That is, a software developer can use the Windows APIs to write a Windows-based application without knowing the details about how the Windows core routines work internally. The developer also doesn't need to know the implementation details of how Windows device drivers communicate with the hardware. Likewise, a hardware vendor could write a Windows device driver for communicating with a hardware device using the published development interfaces without knowing the details of the Windows core internals or the internals of a Windows-based application.

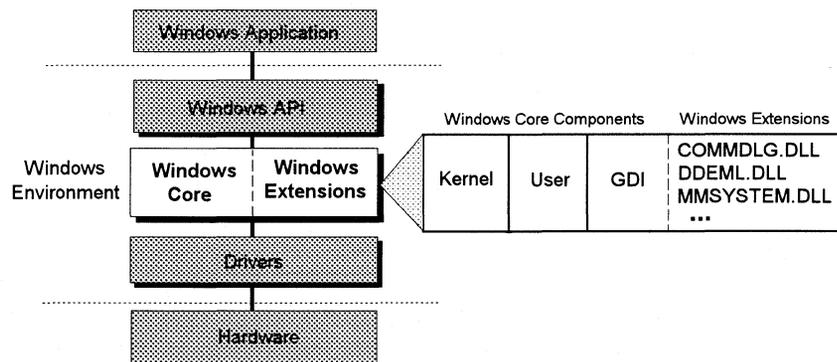
Windows-based applications use the published Windows application programming interface (API) to call routines present in the Windows core or in Windows extensions. For example, an application may call the `CreateWindow` function to tell Windows to create a new window, or may call the `DrawText` function to tell Windows to display text in a window or on a given output device. At the Windows Application layer, the calls to the Windows core and extension routines are independent of the hardware configuration. The

Windows API is documented in the *Microsoft Windows Software Development Kit (SDK)*.

The second highest layer, the Windows core, is the heart of the Windows environment. This layer contains the operating system kernel and the operating system support routines. Extensions to the Windows environment are also present in this layer. Windows extensions are provided in the form of Dynamic Link Libraries (DLLs) and provide functionality that can be used by both Windows-based applications and by elements of Windows itself. (DLLs are explained in the next section.)

Figure 2.2

Windows core consists of the Kernel, User, GDI, and extension routines.



The Windows core consists of the kernel, user, and GDI routines. Kernel (KRNLX86.EXE) provides the operating system kernel functionality and is responsible for handling file I/O, managing memory, and performing system task scheduling in the Windows environment. The version of Kernel used for standard mode and 386 enhanced mode depends on the processor type used in your computer and the amount of memory you have installed.

User (USER.EXE) is responsible for handling user input and output, including managing the keyboard, mouse, sound driver, timer, in addition to communications ports, and managing and keeping track of elements of the user interface (which includes components such as windows, icons, menus, and dialog boxes).

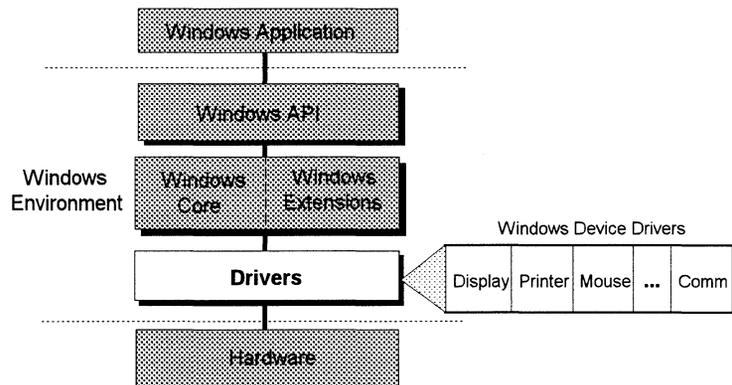
The Graphics Device Interface, GDI (GDI.EXE), is responsible for managing the drawing of graphic primitives, manipulating bitmaps, and handling interaction with the device-independent drivers layer (including display and printer output devices).

The Windows extensions provided with Windows 3.1 include additional support routines for supporting common functionality universal to all Windows-based applications. For example, standardized dialog boxes for common user interface operations are supported by the common dialogs extension (COMMDLG.DLL), a high-level way for managing Dynamic Data Exchange sessions is supported by the DDE management library

(DDEML.DLL), and a library providing services for applications to support multimedia functionality that includes playing and recording audio with waveform and MIDI audio devices is supported by the multimedia systems extension (MMSYSTEM.DLL).

Figure 2.3

Windows Driver layer consists of drivers for hardware devices such as display adapters and printers.



The lowest layer of the Windows environment consists of the Windows device drivers. Each device driver tells Windows how to communicate with a particular hardware device (such as a display adapter, printer, network, or audio device). For example, a video display driver tells Windows how to manipulate bits on a video adapter card to display images, a keyboard driver tells Windows how to interpret signals received from the keyboard, and a mouse driver tells Windows how to interact with the mouse to detect when the mouse is moved and when the user presses a button on the mouse. Windows includes device drivers for many popular hardware devices. Hardware device manufacturers can write additional device drivers to support their particular hardware device. Device drivers are documented in the *Microsoft Windows Device Driver Kit (DDK)*.

Windows Dynamic Link Libraries (DLLs)

The Windows environment is extensible, and thus extremely flexible. As was mentioned in the previous section, many of the components of Windows are written as *dynamic link libraries (DLLs)*. A DLL is an executable file that enables applications to share code and other resources necessary to perform particular tasks. Software developers can extend the Windows environment by creating a DLL that contains routines for performing operations, and then make the DLL available to other Windows-based applications (in addition to internal Windows routines) running on the workstation.

DLLs most often appear as files with a .DLL filename extension; however, they may also have an .EXE or other filename extension. For example, SHELL.DLL provides the drag-and-drop routines that Windows File Manager uses and has a

.DLL filename extension. KERNEL.EXE, USER.EXE, and GDI.EXE are examples of DLLs with .EXE filename extensions. DLLs provide code, data, or routines to applications running in the Windows operating system. For example, User (USER.EXE) provides the code for the “CreateWindow” function, which applications use when they want to create a new window on the screen.

There are two mechanisms for telling an application program how to interact with a software function library. A mechanism called *static binding* is used when a software developer wants to incorporate software routines into his/her application program but keep the function routines internal to the program—that is, the software functions are only available to the application that contains the library. When the software developer creates the final executable program, the language linker statically resolves the function calls in the main application program with the function routines in the function library. This happens only once—at the time the executable program is being built. This method is typically the way that MS-DOS-based applications (that is, non-Windows-based applications) are created.

Another mechanism for telling application programs how to interact with a software function library is called *dynamic binding*. Dynamic binding differs from static binding in that the function-call references in the main application program are dynamically resolved with the function routines in the function library at the time the application runs. This happens each and every time the application program is run. Windows uses dynamic binding to tie function references in a Windows-based application program to functions in a DLL.

One advantage of a DLL, and the way that it interacts with the Windows environment, is that Windows will only load into memory the components of the function library that it needs at a given time. When the application that called the DLL function is finished with the function, Windows may remove from memory the components of the DLL that it doesn't need. This results in efficient memory management within the Windows environment.

Windows Operating Modes

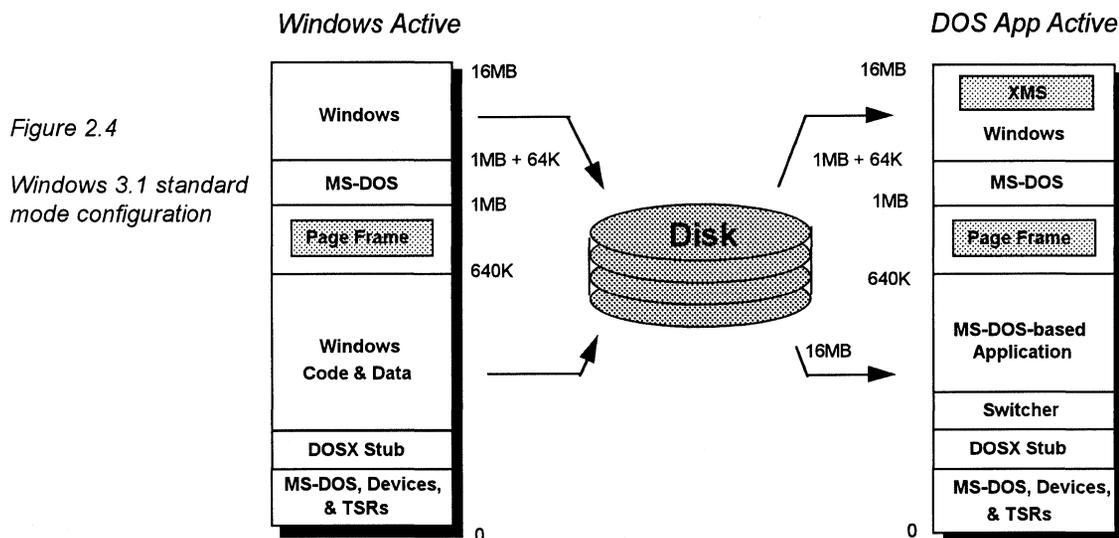
Windows 3.1 supports two operating modes—standard mode and 386 enhanced mode. The default operating mode for your computer is determined by the computer's hardware, the amount of memory it has, and any startup switches you specify. The following sections describe these two operating modes.

Standard Mode

Windows 3.1 standard mode requires a computer with an 80286 processor or higher. This mode provides access to extended memory for Windows-based

applications. Windows doesn't use virtual memory in this mode. MS-DOS-based applications run in a full screen rather than in windows.

In standard mode, Windows does not provide the ability to multitask with MS-DOS-based applications. When an MS-DOS-based application (that is, a non-Windows-based application) is invoked, Windows swaps the components that reside in conventional memory (except for the DOSX stub and the switcher code) to disk in order to allow the MS-DOS-based application to run, and suspends any running Windows-based applications. When the user switches back to the Windows environment (or quits the MS-DOS-based application), Windows loads its components back from the disk and resumes running any Windows-based applications that were previously running. Figure 2.4 illustrates the relationship between these components when running Windows 3.1 in standard mode.



A standard mode Windows configuration consists of the extended memory driver (HIMEM.SYS), the extended memory support routines for Windows (DOSX.EXE), the Kernel (KRNLx86.EXE), and User (USER.EXE), and GDI (GDI.EXE) functions.

386 Enhanced Mode

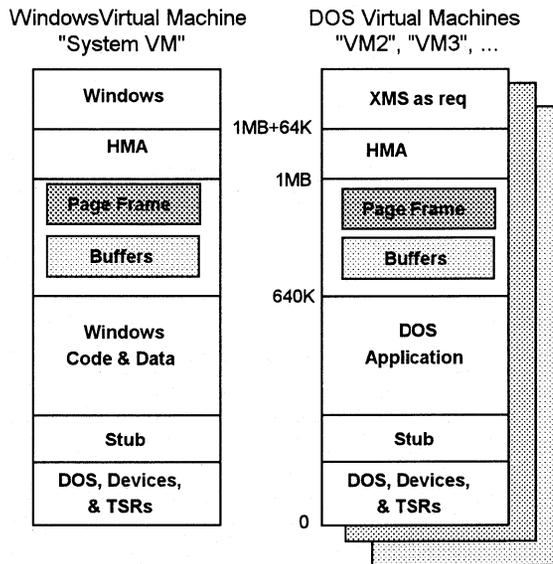
Windows 3.1 also runs in 386 enhanced mode. This mode is designed for a computer with an 80386 processor or higher. Windows 386 enhanced mode allows multitasking of MS-DOS-based applications and enables you to have more control over those applications than is provided in standard mode. For example, in 386 enhanced mode you can run an MS-DOS-based application in

a window, whereas in standard mode, MS-DOS-based applications can run only in a full screen.

A 386 enhanced mode Windows configuration consists of the extended memory driver (HIMEM.SYS), the Windows enhanced mode kernel (WIN386.EXE) providing the virtual machine manager (VMM) and various internal virtual device drivers, the Windows 386 kernel (KRNL386.EXE), the User (USER.EXE) functions, the GDI (GDI.EXE) functions, and other miscellaneous external virtual device drivers.

Figure 2.5

Windows 3.1
386 enhanced mode
configuration



Most importantly, 386 enhanced mode supports the virtual 8086 mode (V86) of 80386 (and higher) CPUs. The V86 mode of the 80386 CPU allows an application running in this mode to run as if it were on its own self-contained computer, when in fact it may be multitasking with other concurrently running applications.

Virtual Machines

The 80386 CPU separates each V86 mode task into its own *virtual machine*. A virtual machine (VM) is supported directly by the Intel 80386 (or higher) CPU and is an executable task consisting of an application, supporting software (such as ROM BIOS and MS-DOS), memory, and CPU registers. Windows uses a single virtual machine, called the *system VM*, in which to run the Windows kernel, other Windows core components and extensions, and all Windows-based applications. As illustrated in Figure 2.5, Windows creates a separate virtual machine called a *DOS VM*, for each running MS-DOS-based application.

The Virtual Machine Manager (VMM) is used to manage the different VMs running in the Windows environment. It provides the services needed to create and manage the virtual machines where Windows and MS-DOS-based applications run. This allows MS-DOS-based applications to run concurrently with Windows when they are launched by Program Manager or another Windows-based application.

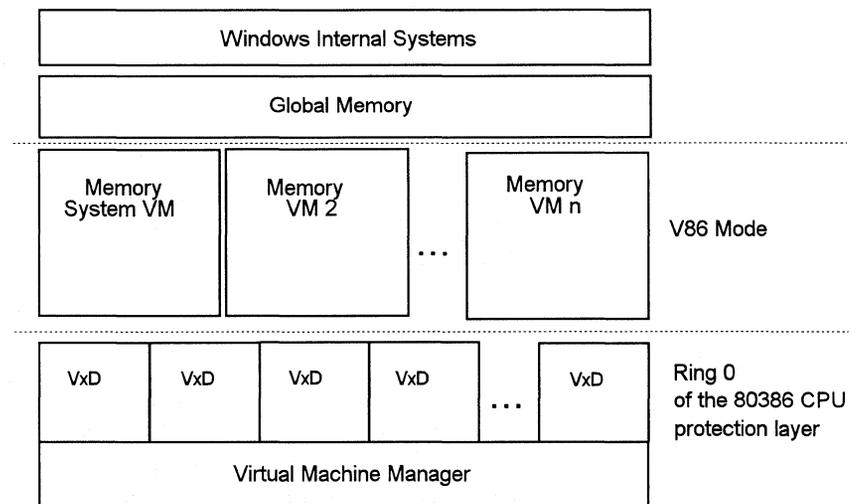
In order to provide support for the Windows environment and for the virtual machines while Windows is running in 386 enhanced mode, Windows uses virtual device drivers to communicate between the system and DOS VMs and hardware devices.

Virtual Device Drivers (VxDs)

A virtual device driver (VxD) is a 32-bit, protected-mode, dynamic link library (DLL) that manages a system resource, such as a hardware device or installed software, so that more than one application can use the resource at the same time. The term *VxD* is used to refer to a general virtual device driver—the *x* represents the type of device driver. For example, a virtual device driver for a display device is known as a VDD. Windows running in 386 enhanced mode uses virtual devices to support multitasking for MS-DOS-based applications. Virtual devices work in conjunction with Windows to process interrupts and carry out I/O operations for a given application without disrupting how other applications run.

Figure 2.6

Relationship of the Virtual Machine Manager, virtual device drivers (VxDs), and virtual machines (VMs) in memory



Virtual devices support all hardware devices for a typical computer, including the programmable interrupt controller (PIC), timer, direct-memory-access (DMA) device, disk controller, serial ports, parallel ports, keyboard device, math coprocessor, and display adapter. A virtual device can contain the device-specific code needed to carry out operations on the device. A virtual device is

required for any hardware device that has settable operating modes or retains data over any period of time. In other words, if the state of the hardware device can be disrupted by switching between multiple applications, the device must have a corresponding virtual device. The virtual device keeps track of the state of the device for each application and ensures that the device is in the correct state whenever an application continues.

Although most virtual devices manage hardware, some manage only installed software, such as an MS-DOS device driver or a terminate-and-stay-resident (TSR) program. Such virtual devices often contain code that either emulates the software or ensures that the software uses data that applies only to the currently running application. ROM BIOS, MS-DOS, MS-DOS device drivers, and TSRs provide device-specific routines and operating system functions that applications use to indirectly access the hardware devices. Virtual devices are sometimes used to improve the performance of installed software; the 80386 and compatible microprocessors can run the 32-bit protected-mode code of a virtual device more efficiently than the 16-bit real-mode code of an MS-DOS device driver or TSR.

Windows 3.1 includes a number of virtual device drivers to support common hardware devices and installable software. The following list describes some of the virtual devices that are provided with Windows 3.1 and which may be present in the **[386Enh]** section of the SYSTEM.INI (listed as an entry beginning with **device=**). Windows provides many other virtual devices that help support other virtual devices. For example, many virtual devices use the services provided by the V86 memory manager (V86MMGR) to reserve V86 mode memory.

<i>Virtual Device</i>	<i>Description</i>
biosxlat	The virtual BIOS device maps the ROM BIOS APIs for protected-modes applications, allowing Windows-based applications and device drivers access to ROM BIOS services.
blockdev	The virtual block device coordinates devices calling block I/O services and FastDisk devices that provide those services for specific hard-disk controllers.
combuff	The virtual communications buffer device works in conjunction with the virtual communications device (VCD) to buffer serial port input.
dosmgr	This virtual device is responsible for management of the interaction between a virtual machine (VM) and MS-DOS. Among the services it provides is the API translation of the Interrupt 21h MS-DOS API for protected-mode applications.
ebios	The virtual extended BIOS device reserves the memory occupied by the EBIOS on the given computer.
int13	The virtual Interrupt 13h device traps and emulates Interrupt 13h BIOS functions. It is used in conjunction with the virtual block device.

Virtual Device	Description	<i>(continued)</i>
pagefile	The virtual swap-file device manages the virtual memory paging file and carries out all read and write operations requested by the virtual paging device (PAGESWAP). This virtual device can use 32-bit disk access, such as provided by the Western Digital 1003 (WDCTRL) device, or use either MS-DOS system functions (Interrupt 21h) or ROM BIOS routines (Interrupt 13h) to carry out read and write operations.	
pageswap	The virtual paging device provides demand-paging support for Windows. The virtual device reads and writes pages using the virtual swap-file device.	
parity	This virtual device installs an NMI (Non-Maskable Interrupt) handler to detect memory and bus parity errors that are signaled via NMI.	
vcd	The virtual communications device virtualizes the standard serial ports on Industry Standard Architecture (ISA) computers, supporting ports COM1 through COM4.	
vdd	The virtual display device virtualizes the display adapter, including video memory, registers, and ROM BIOS interrupts. This virtual device also provides APIs to permit an accompanying 386-enhanced-mode grabber to draw screen output for a MS-DOS-based applications when run in a window, and to retrieve screen contents for cut and paste operations. Windows includes virtual display devices for CGA, EGA, VGA, 8514/a, Hercules, and Video 7 VGA.	
vdmad	The virtual direct memory access (DMA) device virtualizes DMA devices. This virtual device provides services that other virtual devices, such as the virtual floppy disk device, use to carry data transfers using DMA channels.	
vfd	The virtual floppy drive device removes special timer port trappings to ensure that copy protection schemes work properly and communicates with the virtual DMA device to synchronize DMA channel use.	
vkcd	The virtual keyboard device virtualizes keyboard input, supports Windows shell hot keys, provides for pasting text into a virtual machine, and manages keyboard input when in message mode.	
vmcpd	The virtual math coprocessor device is used to virtualize access to the math coprocessor.	
vmmd	The virtual mouse device virtualizes mouse input and maps the Interrupt 33h API for use by protected-mode applications.	
vmpoll	The virtual machine polling device is responsible for detecting when a virtual machine is idle waiting for keyboard input. The "Detect Idle Time" bit in MS-DOS-based application PIF files is a behavior bit that affects the operation of the VMPOLL virtual device driver (VxD).	
vnetbios	The virtual NETBIOS device maps the NETBIOS API for protected-mode applications, allowing Windows-based applications to gain access to the network. It also handles asynchronous network transactions by mapping the application's buffer into global memory so the network software can access the buffer even if another application is running at the time.	
vpd	The virtual printer device virtualizes the parallel ports. If a second virtual machine tries to access one of the ports while it is being used by another application, a contention dialog box is displayed to the user, allowing the user to resolve the conflict.	
vpicd	The virtual programmable interrupt controller device is used to enable interrupt requests (IRQs).	
vsd	The virtual sound device is used to virtualize access to a computer's speaker.	

<i>Virtual Device</i>	<i>Description</i>	<i>(continued)</i>
vtd	The virtual timer device is used to virtualize the timer services provided by the 8253 Programmable Interval Timer chip.	
wdctrl	The 386-enhanced-mode block device provides direct access to hard-disk controllers that are compatible with Western Digital 1003 devices or ST506 hard disk controllers. Direct access allows virtual devices to carry out input and output operations on disk entirely in ring 0 protected mode, bypassing the real mode BIOS and improving performance.	
wshell	The virtual Win386 shell interface device is a communication device used for the management of MS-DOS applications. This VxD provides “glue” code for the WINOLDAP (WINOA386.MOD) Windows-based application that is the “windows agent” for all of the MS-DOS applications being run. The WSHELL VxD also provides system level services for other VxDs. In particular, the WSHELL VxD contains the code that understands how to display “full screen message boxes,” and waits for and interprets the user’s response. An example of one of these is the blue background restart screen that is displayed when the user presses Ctrl+Alt+Del.	

Windows for Workgroups 3.1 Architecture

The Windows for Workgroups architecture builds on the architecture implemented in Windows 3.1 by providing integrated network functionality in the base operating system. The incorporation of networking components in Windows for Workgroups helps to define a standard platform for workgroup members to communicate and exchange information.

Windows for Workgroups Network Drivers

The Windows for Workgroups network functionality consists of the following components:

- Protocol Manager (PROTMAN.DOS)
- NDIS-compatible network adapter card (MAC) driver (*.DOS)
- Workgroups driver (WORKGRP.SYS)
- NetBEUI transport protocol
- NETBIOS interface
- Network redirector
- Windows for Workgroups Winnet driver
- Windows for Workgroups Server virtual device driver

The implementation of the Protocol Manager, the NDIS network adapter card driver, and the Workgroups driver is the same whether running Windows for Workgroups in standard or 386 enhanced mode. The implementations of the

NetBEUI transport protocol, the NETBIOS interface, and the network redirector are different depending on whether you are running Windows for Workgroups in standard or 386 enhanced mode.

As described in Chapter 1, the Protocol Manager (PROTMAN.DOS) is used to bind the network transport protocols to an NDIS-compliant network adapter card (MAC) driver. PROTMAN.DOS is a real-mode MS-DOS device driver, is specified in the CONFIG.SYS file (before the other network device drivers), and is loaded during the MS-DOS system initialization process. The Protocol Manager examines information specified in the PROTOCOL.INI file and uses the information to bind the network transport protocols identified in the PROTOCOL.INI to the specified NDIS-compliant network adapter card driver.

The NDIS network adapter card driver is used to define how the network protocol will communicate with a network adapter card. The NDIS drivers (*.DOS) used on your workstation depend on the network adapter card(s) you have installed. They are real-mode MS-DOS device drivers that are specified in the CONFIG.SYS file and are loaded during the MS-DOS system initialization process.

The Workgroups driver (WORKGRP.SYS) is the real-mode stub for the NetBEUI protocol and the network redirector components. This driver is used to provide the real-mode component support for accessing network functionality in protected-mode. WORKGRP.SYS is a real-mode MS-DOS device driver, is specified in the CONFIG.SYS file, and is loaded during the MS-DOS system initialization process.

Sample CONFIG.SYS file entries from a Windows for Workgroups installation showing the specification of the aforementioned drivers would look something like the following:

```
device=C:\WINDOWS\protman.dos /i:C:\WINDOWS      Protocol Manager
device=C:\WINDOWS\exp16.dos                    NDIS driver for
                                                Intel EtherExpress
                                                16
device=C:\WINDOWS\workgrp.sys                  Real-mode stub for
                                                NetBEUI and/or
                                                network redirector
```

Note The Protocol Manager (PROTMAN.DOS) must be specified as the first network driver in the CONFIG.SYS file (that is, before the NDIS drivers and the Workgroups driver). If PROTMAN.DOS doesn't precede the other driver files, the driver files will display an error message stating that the Protocol Manager wasn't found during system initialization.

In order to start the network software and to bind the real-mode network drivers specified in the CONFIG.SYS file, the **net start** command needs to be entered before starting Windows for Workgroups. The Windows for Workgroups Setup program includes the **net start** command in the AUTOEXEC.BAT file when Windows for Workgroups is installed.

The discussion of other Windows for Workgroups network components and their implementation characteristics is provided in the following sections.

Standard Mode

To run Windows for Workgroups in standard mode, you need an Intel 80286 (or higher) computer with a minimum of 2MB of memory. The minimum memory requirements for running Windows for Workgroups is slightly higher than that required by Windows 3.1 in order to support the network functionality.

A Windows for Workgroups workstation running in standard mode can act as a client in the workgroup, and thus can use shared resources. However, a workstation running in standard mode can't share resources, due to the implementation of the Windows for Workgroups server components as virtual device drivers (VxDs), which can only run in 386 enhanced mode. (See the next section on 386 enhanced mode for more information.)

A workstation running in standard mode has access to other network functionality common to all Windows for Workgroups workstations including the use of Network DDE. (For more information about Network DDE, see Chapter 10.)

In standard mode, the NetBEUI transport protocol, the NETBIOS interface, and the network redirector are loaded and run in conventional memory. Before running Windows for Workgroups in standard mode, you can start the network redirector by typing the following command at the MS-DOS command prompt:

```
NET LOGON
```

This command loads the transport protocols identified in the PROTOCOL.INI file, loads the NETBIOS interface routines (present in NET.EXE), and starts the network redirector. You'll then be prompted to log on to the network. If you log on to the network successfully, the persistent network connections in use during the last Windows for Workgroups session are automatically reestablished. If you don't log onto the network before starting Windows for Workgroups, the Windows for Workgroups Winnet driver prompts you to do so when it loads.

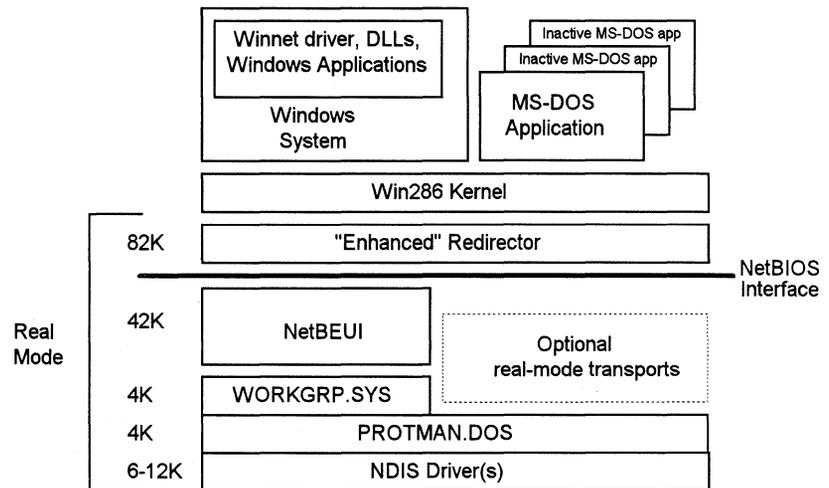
The Windows for Workgroups Winnet driver (WFWNET.DRV) is loaded by the kernel and provides Windows support routines for accessing the Windows for Workgroups network. This driver is responsible for handling user interaction with the network (for example, logging on to or off of the network) and for accessing shared resources (for example, connecting to or disconnecting from shared files and printers).

Summary of Windows for Workgroups Standard Mode Components

Figure 2.7 illustrates the standard mode configuration for Windows for Workgroups. As described in the previous section, the networking components available to a Windows for Workgroups workstation in standard mode are real-mode implementations and reside in conventional memory. The memory values listed in Figure 2.7 are approximate and will vary depending on your system configuration.

Figure 2.7

Windows for Workgroups standard mode configuration



Standard Mode Startup Sequence

This section describes the sequence in which Windows for Workgroups loads the Windows standard mode components at the time the user starts Windows for Workgroups. Windows for Workgroups runs in standard mode either because the user has an Intel 80286-based computer, because it doesn't have enough memory on an Intel 80386-based computer to run in 386 enhanced mode, or because the user explicitly starts standard mode with the /s or /2 Windows startup option. Since the networking components are started before starting Windows, the startup process below is similar to that of standard mode in Windows 3.1 (except for the Windows for Workgroups network driver).

Here is the series of events that makes up the startup process:

1. The user starts the network software by typing **net logon** at the MS-DOS prompt. If the user logs onto the network successfully at this point, the persistent network connections are restored (for example, file and printer connections active during the last Windows for Workgroups session).
2. The user invokes WIN.COM by typing **win** at the MS-DOS prompt.
3. WIN.COM invokes the MS-DOS *exec* function to load DOSX (to provide extended memory access support).
4. DOSX loads the standard mode kernel (KRNLx86.EXE).
5. KRNLx86.EXE loads the following files:
 - The Windows drivers (identified as *.DRV in the SYSTEM.INI file)
 - GDI (GDI.EXE)
 - User (USER.EXE)
 - Supporting files (for example, fonts)
 - The Windows for Workgroups network driver (WFUNET.DRV)
6. WFUNET.DRV loads the Network DDE background application (NETDDE.EXE), and the ClipBook Server background application (CLIPSRV.EXE).
7. WFUNET.DRV prompts the user to log on to the network if the user has not done so already. If the user logs on to the network successfully, the WFUNET.DRV then restores the persistent network connections (for example, file and printer connections active during the last Windows for Workgroups session).
8. KRNLx86.EXE launches the Windows shell identified by the **shell=** entry in the **[boot]** section of the SYSTEM.INI file. By default, this is the Windows Program Manager.

386 Enhanced Mode

To run Windows for Workgroups in 386 enhanced mode, you need an Intel 80386 (or higher) computer with a minimum of 3MB of memory (4MB of memory is recommended). The minimum memory requirements for running Windows for Workgroups is slightly higher than that required by Windows 3.1 in order to support the network functionality.

A Windows for Workgroups workstation running in 386 enhanced mode can act as both a client and a server in the workgroup, and therefore can both

access and share resources. A workstation running in 386 enhanced mode can also use the Network DDE functionality provided with Windows for Workgroups to access and share information with remote workstations. (For a detailed explanation of Network DDE, see Chapter 10.)

Windows for Workgroups running in 386 enhanced mode takes advantage of the Windows 3.1 architecture and implements many of its networking components as VxDs.

Virtual Devices Used By Windows for Workgroups

Windows for Workgroups takes advantage of the faster, more efficient 32-bit virtual device driver (VxD) architecture of Windows to provide network functionality in 386 enhanced mode. The VxD implementation of the network components doesn't use conventional memory. This means that Windows for Workgroups can provide more available memory for MS-DOS-based applications than can be made available by Windows 3.1 running with a non-integrated network. As much as 600K of free conventional memory is available to run a MS-DOS-based application in a DOS virtual machine under MS-DOS 5.0 and Windows for Workgroups with the networking components loaded.

The following table summarizes the virtual device drivers provided with Windows for Workgroups to support the network functionality in 386 enhanced mode.

Virtual device driver	Function
VNB.386	Virtual device driver version of the NetBEUI transport
VNETBIOS.386	Virtualizes the NETBIOS interface
VREDIR.386	Virtual device driver version of the network redirector
VSERVER.386	The file and print server (SMB-based)
VBROWSE.386	Provides browsing services to the network
VNETSUP.386	Provides various network support functions for VREDIR.386, VSERVER.386, VNB.386, and VBROWSE.386
VSHARE.386	Virtual device driver version of SHARE.EXE
VWC.386	Virtual workgroup client device. Provides support for mapping redirected network devices across multiple virtual machines when the real-mode redirector is used

The virtual device driver versions of the NetBEUI transport (VNB.386) and the network redirector (VREDIR.386) are analogous to their real-mode counterparts. The VNETBIOS.386 virtual device driver virtualizes the NETBIOS interface so that Windows for Workgroups workstations may concurrently run network-aware Windows and MS-DOS-based applications that use NETBIOS to communicate.

The Windows for Workgroups file and print server functionality is implemented in the VSERVER.386 virtual device driver. This virtual device driver provides other workstations in your workgroup access to files and printers on your local workstation. Because the server functionality is implemented as a virtual device driver, a Windows for Workgroups workstation running in standard mode can use only the network redirector and is only able to use shared resources. The VSERVER.386 virtual device driver implements the server functionality using the SMB protocol as discussed in Chapter 1. A Windows for Workgroups workstation in 386 enhanced mode can be accessed by other network redirectors that also support the SMB protocol. See Chapter 8 for more information about integrating with other local-area networks.

The VBROWSE.386 virtual device driver provides the browsing services for the network. It supports other Windows for Workgroups networking components to help identify other Windows for Workgroups workstations on the network and supports browsing of network resources in Windows-based applications.

The VNETSUP.386 virtual device driver provides various network support functions for the virtual device driver implementation of the network redirector (VREDIR.386), the network server (VSERVER.386), the NetBEUI transport (VNB.386), and the network browsing support routines (VBROWSE.386). VNETSUP.386 is responsible for pulling the information contained in the **[network]** section of the SYSTEM.INI file and passing it to other Windows for Workgroups network components.

The VWC.386 virtual device is responsible for mapping network devices that are connected or disconnected in a virtual machine (for example, from an MS-DOS command prompt or from the Windows system VM using File Manager or Print Manager) across the multiple virtual machines executing in the system when the real-mode redirector is used. For example, if a DOS VM is executing in the background of the system and a network connection is made in File Manager to a network file share, the network connections displayed by typing **net use** at the MS-DOS prompt in the DOS VM will reflect the new network connection made in File Manager.

VSHARE Virtual Device Driver

MS-DOS includes a terminate-and-stay-resident (TSR) program called SHARE.EXE that supports file-sharing and locking in a network or a multitasking environment in which programs share files. MS-DOS uses the code loaded by SHARE.EXE to validate all read and write requests from programs. SHARE.EXE keeps track of files that are opened by applications and will prevent two or more applications or processes from modifying the same file at the same time (which could result in the original file being damaged or destroyed if SHARE.EXE was not loaded). SHARE.EXE is typically specified in the AUTOEXEC.BAT file and is loaded during system initialization.

With Windows 3.1, in order for Windows and MS-DOS-based applications to support the file-sharing and locking capabilities offered by Share, SHARE.EXE needs to be loaded before Windows is started. This results in Share consuming approximately 6K of conventional memory. The memory used by Share before starting Windows is also reflected by each virtual machine in which MS-DOS-based applications are running. Thus, valuable conventional memory is used even if a given MS-DOS-based application doesn't require file-sharing or locking support.

To support the networking environment of Windows for Workgroups, the Share TSR was rewritten as a virtual device driver for use with Windows for Workgroups 386 enhanced mode. The VSHARE.386 virtual device driver provides full compatibility with the version of SHARE.EXE provided with the version of MS-DOS that you are using. One key difference between SHARE.EXE and VSHARE.386 is that SHARE.EXE statically allocates the file-sharing table size and the number of simultaneous file region locks allowed at startup, and thus these settings cannot change until the workstation is rebooted. VSHARE.386, on the other hand, dynamically allocates memory for the file-sharing functionality, and thus can increase (or decrease) its allocation when additional resources are requested by the system.

Since the VSHARE.386 virtual device driver virtualizes the sharing functionality, file-sharing and locking are available to all Windows-based applications and MS-DOS-based applications within the Windows environment without the conventional memory requirement of the MS-DOS Share TSR.

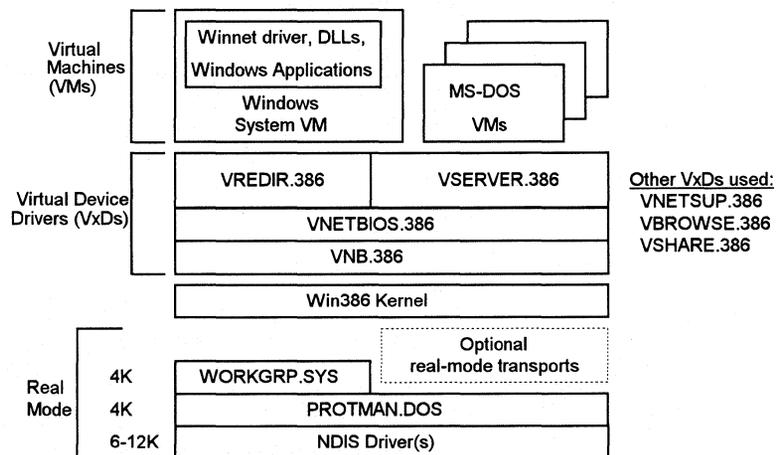
If the MS-DOS Share TSR is loaded before you run Windows for Workgroups, the VSHARE.386 will stop the Share TSR, but won't remove it from memory. VSHARE.386 will assume the file-sharing and locking responsibility from SHARE.EXE.

Note If you don't need to support file-sharing and locking outside of the Windows environment (that is, you run only MS-DOS applications that require Share from within Windows), you can remove SHARE.EXE from your AUTOEXEC.BAT and recover the memory previously used by the MS-DOS Share TSR (approximately 6K). However, if you are using MS-DOS 4.0 and presently have SHARE.EXE in your AUTOEXEC.BAT to support disk partitions greater than 32MB in size, you should leave SHARE.EXE in your AUTOEXEC.BAT file. This is to ensure support for large disk partitions when you run an MS-DOS application outside of Windows for Workgroups.

Summary of Windows for Workgroups 386 Enhanced Mode Components

Figure 2.8 illustrates the 386 enhanced mode configuration for Windows for Workgroups. As described in the previous section, the networking components used by a Windows for Workgroups workstation in 386 enhanced mode are protected-mode virtual device drivers and reside in extended memory. The memory values listed in the figure are approximate and will vary depending on your system configuration.

Figure 2.8
Windows for
Workgroups 386
enhanced mode
configuration



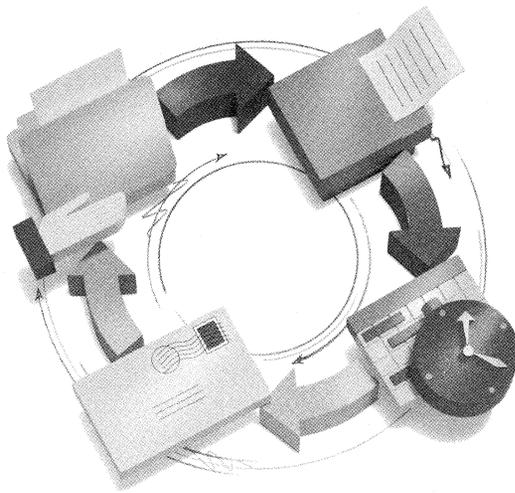
386 Enhanced Mode Boot Sequence

When a user starts Windows for Workgroups in 386 enhanced mode, the following startup process is performed. Due to the integrated network functionality present in Windows for Workgroups, the boot process below is different than that of 386 enhanced mode in Windows 3.1.

1. The user invokes WIN.COM by typing **WIN** at the MS-DOS prompt
2. WIN.COM invokes the MS-DOS *exec* function to load the Windows 386 enhanced mode system kernel (WIN386.EXE).
3. WIN386.EXE loads the following:
 - The Virtual Machine Manager (VMM)
 - All virtual device drivers (VxDs) specified in the SYSTEM.INI file
4. The network support VxD (VNETSUP.386) initializes the Windows for Workgroups VxDs. The NetBEUI protocol (VNB.386) and the NETBIOS interface (VNETBIOS.386) are bound together. The VNB.386 and VREDIR.386 are bound to the WORKGRP.SYS device driver.
5. The network redirector VxD (VREDIR.386) starts the redirector workstation service (this is like typing **net start workstation** at the MS-DOS command prompt).
6. WIN386.EXE loads the 386 enhanced mode kernel (KRNL386.EXE).
7. KRNL386.EXE loads the following files:
 - The Windows drivers (identified as *.DRV in the SYSTEM.INI file)
 - GDI (GDI.EXE)
 - User (USER.EXE)
 - Supporting files (for example, fonts)
 - The Windows for Workgroups network driver (WFVNET.DRV)
8. The network server VxD (VSERVER.386) starts the server service.
9. The persistent network shares as configured by the Windows for Workgroups workstation are shared on the network.
10. WFVNET.DRV loads the Network DDE background application (NETDDE.EXE) and the ClipBook Server background application (CLIPSRV.EXE).

11. WFWNET.DRV prompts the user to log on to the network if the user has not done so already. If the user logs on to the network successfully, the WFWNET.DRV then restores the persistent network connections (for example, printer connections and file connections made during the last Windows for Workgroups session).
12. KRNL386.EXE launches the Windows shell as identified by the **shell=** entry in the **[boot]** section of the SYSTEM.INI file. By default this is the Windows Program Manager.

Installation and Setup



Chapter 3 Windows for Workgroups Installation **3-1**

About Windows for Workgroups Setup	3-2
Windows for Workgroups Setup: A Technical Discussion.....	3-7
Setting Up Windows for Workgroups for a Network Installation.....	3-17
Using Automated Setup: Setup /h	3-22
Maintaining Windows for Workgroups with Setup	3-31

Chapter 4 Windows for Workgroups Files **4-1**

About the Windows for Workgroups Files.....	4-2
WIN.COM.....	4-2
The Core Files	4-3
Drivers, Fonts, and International Support Files.....	4-3
MS-DOS Support Components of Windows for Workgroups	4-12
Windows for Workgroups Applications, Setup, and Other Files.....	4-17
Files You Can Delete.....	4-22

Chapter 5 Windows for Workgroups Setup Information Files **5-1**

About the Setup Information Files	5-3
SETUP.INF: Setup Information File	5-6
APPS.INF: Application Information File	5-42
CONTROL.INF: The Windows Control Panel Information File.....	5-48
NETWORK.INF: Network Information File	5-50
OEMSETUP.INF: OEM Network Setup Information File	5-64
Modifying .INF Files for Custom Installations	5-69

About the Initialization Files	6-2
WIN.INI: Windows for Workgroups Initialization File	6-6
SYSTEM.INI: System Initialization File	6-29
PROTOCOL.INI: Network Protocol Initialization File.....	6-73
PROGMAN.INI: Program Manager Initialization File.....	6-76
CONTROL.INI: Control Panel Initialization File	6-79
WINFILE.INI: File Manager Initialization File	6-80
MSMAIL.INI: Microsoft Mail Initialization File	6-82
SHARED.INI: Shared Mail Settings Initialization File	6-94
SCHDPLUS.INI: Schedule+ Initialization File	6-95

Windows for Workgroups Installation

This chapter contains an overview of the installation of Microsoft Windows for Workgroups 3.1, plus a detailed technical discussion of the Windows for Workgroups Setup program. This chapter concludes with details about setting up Windows for Workgroups 3.1 for a network and creating a custom installation routine for automated setup.

Related information

- *Microsoft Windows for Workgroups User's Guide: Chapter 1, "Windows Basics"*
- *Microsoft Windows for Workgroups Getting Started*
- *Windows for Workgroups Resource Kit: Chapter 5, "Windows for Workgroups Setup Information Files"; Chapter 6, "Windows for Workgroups Initialization Files"; Chapter 14, "Troubleshooting Windows for Workgroups"*

Contents of This Chapter

About Windows for Workgroups Setup	3-2
Running Setup	3-3
Choosing Setup Options	3-4
Choosing Express Setup vs. Custom Setup	3-6
Windows for Workgroups Setup: A Technical Discussion	3-7
About the Windows Operating Modes	3-7
MS-DOS Mode Setup	3-8
Windows Mode Setup	3-12
Setting Up Windows for Workgroups for a Network Installation	3-17
General Guidelines	3-17
Placing Windows for Workgroups Files on a Network Server: Setup /a	3-19
Setting Up a Shared Copy of Windows for Workgroups: Setup /n	3-21
Using Automated Setup: Setup /h	3-22
Creating the System-Settings File	3-23
Sample System-Settings File	3-29
Maintaining Windows for Workgroups with Setup	3-31

About Windows for Workgroups Setup

Note In this manual, *WINDOWS directory* refers to the directory where you choose to install Windows for Workgroups, and *SYSTEM subdirectory* refers to the subdirectory where the Windows for Workgroups system files are stored.

The Windows for Workgroups Setup program is on Disk #1 of the Microsoft Windows for Workgroups disk set. The Setup program guides you through a straightforward series of steps in two basic stages:

1. In the first stage of installation, called MS-DOS mode setup, Setup evaluates your computer system and asks you to verify this information. Then Setup copies the basic Windows for Workgroups files to the system's hard disk.
2. In the second stage of installation, called Windows mode setup, Setup loads Windows for Workgroups and installs the remaining files. Setup then prompts you to make a series of choices about installing printer drivers and other applications.

Information about Setup for Windows for Workgroups 3.1

Look for these features in Windows for Workgroups Setup:

- Setup warns you about the presence of any drivers known to be incompatible with either Setup or Windows for Workgroups. Setup searches for earlier versions of Windows 3.x and various drivers to update driver files. For details about related changes to SYSTEM.INI and WIN.INI, see Chapter 6, "Windows for Workgroups Initialization Files."
- Whether you choose express setup or custom setup, Setup detects the computer for installation, so you seldom have to choose from the list of specific computers and devices. For details about the computers and drivers that Setup detects, see the entries for the **[display]**, **[keyboard.types]**, **[pointing.device]**, and **[computer]** sections in SETUP.INF, as described in Chapter 5, "Windows for Workgroups Setup Information Files."
- Setup will detect the type of network adapter card you have installed in your computer. For details about the network adapter cards that Setup detects, see the entries for the **[netcard]** section in NETWORK.INF, as described in Chapter 5, "Windows for Workgroups Setup Information Files."
- Setup automatically creates a Startup group in Program Manager, which can be used to start a number of selected applications each time Windows for Workgroups runs.



- You can automate Windows for Workgroups installation by creating a script. For details, see “Using Automated Setup: Setup /h” later in this chapter.

Running Setup

You can run Setup in two ways, for two different purposes:



- Type **setup** at the MS-DOS command prompt to install and configure Windows for Workgroups from floppy disks or from a network drive. Do this to upgrade your Windows installation or to install Windows for Workgroups for the first time on your system.
- Run Setup from Program Manager after Windows for Workgroups is installed. Do this to change your system configuration or to add or delete files.

Follow these basic steps to install Windows for Workgroups 3.1 from the installation disks.

To install Windows for Workgroups 3.1 from the disk drive

1. Insert the Windows for Workgroups 3.1 Disk #1 in a floppy-disk drive, and change to the drive letter of the drive.

For example, to switch to the A drive, at the command prompt type **A:** and press ENTER.
2. At the command prompt, type **setup** and press ENTER.

You can include any switches with the **setup** command, as described in “Choosing Setup Options” later in this chapter.
3. Follow the instructions on your screen.

If you have questions about any procedures or options, press F1 for online Help.

Important Don't use the MS-DOS **copy** command to copy the Windows for Workgroups files from the Setup disks. These files are compressed and can't be used unless you run the Setup program. Setup renames the compressed files during installation. If you have files on your system that end with an underscore (_), that file is compressed and has not been installed properly.

For details about installing Windows for Workgroups over a network, see “Setting Up Windows for Workgroups for a Network Installation” later in this chapter.

Choosing Setup Options

You can include various switches when you type **setup** at the command prompt, as described in the following table.

Switch	Purpose
<code>/i</code>	Ignores automatic hardware detection. The user will have to check settings and possibly make corrections on the System Information screen during setup.
<code>/n</code>	Sets up a shared copy of Windows for Workgroups from a network server. For details, see “Setting Up Windows for Workgroups for a Network Installation” later in this chapter.
<code>/a</code>	Begins administrative setup by expanding and copying all files from the Windows for Workgroups installation disks onto a network server and marking the files as read-only. For details, see “Setting Up Windows for Workgroups for a Network Installation” later in this chapter.
<code>/b</code>	Sets up Windows for Workgroups with monochrome display attributes.
<code>/t</code>	Searches the drive for incompatible software that should not run at the same time as Setup or Windows for Workgroups 3.1. (For maintenance only.)
<code>/h:filename</code>	Runs batch mode setup to install Windows for Workgroups with little or no user interaction. The <i>filename</i> is the name of the system-settings file that contains information about the user’s configuration. Include the path if <i>filename</i> is not in the directory that contains the Windows for Workgroups Setup files. For details, see “Using Automated Setup: Setup /h” later in this chapter.
<code>/o:filename</code>	Specifies the SETUP.INF file, including the path if necessary.
<code>/s:filename</code>	Specifies the SETUP.INF file, including a path for the Windows for Workgroups installation disks.

Tip To see a list of the switches available for Windows for Workgroups Setup, at the command prompt type **setup /?** and press ENTER.

You can customize the installation procedure for Windows for Workgroups using the command-line switches. You can also customize the Windows for Workgroups information (.INF) files and the initialization (.INI) files to install a custom version of Windows for Workgroups at your site.

The following decision tree will help you decide how you might want to customize Windows for Workgroups for installation at your site.

Windows for Workgroups custom installation decision tree for a single system

- **You want to customize how Windows for Workgroups is installed:**
 - Choose Custom Setup at the first Setup prompt.
- **You want to customize how Windows for Workgroups works after installation:**
 - Use Control Panel, Program Manager, and Windows for Workgroups Setup.

Windows for Workgroups custom installation decision tree for multiple systems

- **You want to set up Windows for Workgroups on a network:**
 - Use `setup /a` to copy files to the network server, then use `setup /n` to install a shared copy of Windows for Workgroups on the workstations. See “Setting Up Windows for Workgroups for a Network Installation” later in this chapter.
- **You want to automate installation for multiple users:**
 - Create custom system-settings files and use `setup /h`. See “Using Automated Setup: Setup /h” later in this chapter.
- **You want custom choices available in the System Information screen:**
 - Edit device entries in `SETUP.INF` for hardware, and edit printer entries in `CONTROL.INF`. See “Modifying .INF Files to Install Custom Applications” in Chapter 5, “Windows for Workgroups Setup Information Files.”
- **You want custom choices available (or to minimize choices) in the network card selection screen:**
 - Edit network adapter card entries in `NETWORK.INF`. See “Modifying .INF Files to Install Custom Applications” in Chapter 5, “Windows for Workgroups Setup Information Files.”
- **You want custom choices available (or to minimize choices) in the network protocol selection dialog:**
 - Edit network protocol entries in `NETWORK.INF`. See “Modifying .INF Files to Install Custom Applications” in Chapter 5, “Windows for Workgroups Setup Information Files.”

- **You want to customize the applications installed with Windows for Workgroups:**
 - Install custom applications, as described in “Modifying .INF Files to Install Custom Applications” in Chapter 5, “Windows for Workgroups Setup Information Files.”
 - Customize the Windows for Workgroups applications installed, as described in “Modifying .INF Files to Install Custom Windows for Workgroups” in Chapter 5, “Windows for Workgroups Setup Information Files.”
 - Install custom PIFs during installation, as described in “Adding Custom PIF Information in APPS.INF” in Chapter 7 of the *Windows Resource Kit*.
- **You want to customize Program Manager for multiple systems:**
 - Restrict user capabilities or set up shared Program Manager groups as described in “PROGMAN.INI” in Chapter 6, “Windows for Workgroups Initialization Files.”
- **You want a custom Windows for Workgroups startup environment for multiple systems:**
 - Edit WIN.SRC and SYSTEM.SRC before installing Windows for Workgroups, as described in “Editing .INI Source Files” in Chapter 6, “Windows for Workgroups Initialization Files.”

Choosing Express Setup vs. Custom Setup

At the beginning of setup, you are prompted to choose express setup or custom setup. Choose express setup for most standard installations. You only have to provide information about the printers and port connections for your system.

Choose custom setup for precise control over how Windows for Workgroups is installed. Setup prompts you to verify information it detects about the hardware and software in your system, such as the following:

- The directory where you want Windows for Workgroups files installed.
- The computer system hardware, including the monitor, mouse, keyboard, and network adapter card. Windows for Workgroups setup detects most kinds of hardware, but with custom setup you can make more detailed choices about your system configuration.
- The preferred language (you can run Windows for Workgroups in several languages, including French, German, and Spanish). This setting specifies the sorting order used by Windows for Workgroups and does not install any additional language support.
- The printers and printer ports for the computer system.
- The other applications you want to run with Windows for Workgroups.

- The Windows-based applications and supporting files that you want to install.
- The specific Windows for Workgroups components you want to install.

Windows for Workgroups Setup: A Technical Discussion

This section is a step-by-step examination of how the Windows for Workgroups Setup program installs Windows for Workgroups. This information will help you to diagnose and solve problems with a Windows for Workgroups installation, and will also help you to make custom modifications. For additional up-to-date information, see the README file in the WINDOWS directory.

During the installation procedure (if you choose custom setup), setup asks you to confirm that the hardware settings are correct or to modify the settings if necessary. At a more technical level, setup continually checks the system to make sure it can complete the installation procedure. It also performs several automatic steps to upgrade from a previous version of Windows that may be installed on your system. Setup installs printer drivers, applications in the Program Manager group, and third-party drivers. It also performs various maintenance functions.

The Setup program has two parts: MS-DOS mode setup and Windows mode setup.

About the Windows Operating Modes

Windows for Workgroups can run in two different operating modes: standard mode and 386 enhanced mode. For both of these operating modes, the processor runs in protected mode. When you type **win** at the command prompt, Windows for Workgroups automatically runs in the appropriate mode, depending on the computer hardware and the amount of memory the system has:

- **Standard mode** requires a computer with an 80286 processor (or higher) and 2MB of memory. This mode provides access to extended memory using application swap files. Windows for Workgroups does not use virtual memory. MS-DOS-based applications run in a full screen.

If you have an 80386 or higher processor, you can force Windows for Workgroups to run in standard mode by typing **win /s** at the command prompt.

- **386 enhanced mode** requires a computer with an 80386sx processor (or higher) and 3MB of memory (4MB is recommended).

Windows for Workgroups 386 enhanced mode provides access to the virtual memory capability of the 80386sx processor, so that Windows-based applications can use more memory than is physically available. Windows for Workgroups 386 enhanced mode also allows multitasking of MS-DOS-based applications and allows you to have more control over those applications. For example, you can run an MS-DOS-based application in a window.

For technical details about the Windows operating modes, see Chapter 2, “Windows for Workgroups Architecture,” in this manual, and see Chapter 5, “Windows 3.1 and Memory Management,” in the *Windows Resource Kit*.

MS-DOS Mode Setup

In the MS-DOS mode part of setup, the Setup program:

- Confirms the directory where Windows for Workgroups files will be stored.
- Identifies the type of hardware installed in the system.
- Installs Windows for Workgroups files required for Windows mode setup.

Note Throughout the setup procedure, Setup checks that the files it copies are newer than any previous versions on the system. Setup won't copy the version of a file on the installation disks over a newer version of a file that's already on your system, unless you approve the upgrade.

Getting Ready for MS-DOS Mode Installation

MS-DOS mode setup begins by drawing the setup welcome screen, then saves for later use any command-line parameters (such as */n*), the type of processor, and the free memory available. Setup also checks to see if it can read the SETUP.INF file and that the file isn't corrupted. It also checks whether the MS-DOS version is greater than or equal to MS-DOS 3.10. If either the MS-DOS version check or the SETUP.INF check fails, Setup displays a message explaining the failure, and then returns to MS-DOS.

If the MS-DOS version and the SETUP.INF are okay, Setup checks the path for a previous version of Windows. If it can't find Windows on the path, Setup searches every local hard disk for Windows. If Setup finds that the same version of Windows for Workgroups has already been installed on the system, Setup switches to maintenance mode and performs updates of the system drivers.

If the same version of Windows for Workgroups isn't found, Setup checks that at least 376K of conventional memory is available. If the required memory is not free, Setup fails. If the memory is free, then Setup begins installation.

Setting the Windows Directory and Updating the Version

For express setup, if no previous version of Windows is found on the local drives, Setup creates the default directory listed in SETUP.INF (the **defdir**= entry in **[data]**) and begins installation.

For custom setup, Setup displays a prompt showing the suggested directory to install Windows for Workgroups (usually either C:\WINDOWS or the directory containing the previous version of Windows). If you modify the directory for installing Windows for Workgroups, Setup checks to make sure that the path is valid and that you're not installing Windows for Workgroups in the same directory as the source files. If you try to do this, a warning prompts you to modify the directory path.

If Windows 3.0 or Windows 3.1 is in the destination directory, Setup asks if you want to upgrade the earlier Windows installation or enter a different destination directory.

If you answer Yes to upgrade an earlier version of Windows, Setup performs this version update:

- The files listed in the **[win.copy]**, **[win.copy.net]**, or **[win.copy.win386]** section and the **[update.files]** section of SETUP.INF are copied to the appropriate directories.
- Files listed in the **[DelFiles]** section of SETUP.INF are deleted.
- Various entries in WIN.INI and SYSTEM.INI are replaced, based on entries in the **[ini.upd.31]** section of SETUP.INF.
- Program Manager groups are redefined based on entries in the **[new.groups]** section of SETUP.INF.
- The old WINVER.EXE file is deleted and replaced with the current version.
- The closing screen is displayed, asking you to reboot the computer to run Windows for Workgroups or return to MS-DOS.

If you specify another directory, Setup continues installing Windows for Workgroups 3.1 as though it were a first-time installation.

Identifying Hardware

If you run Setup with the /i switch (**setup /i**), Setup displays the System Information screen so that you can choose the computer type, display adapter, mouse, and keyboard driver used in your system. For details about making choices in this screen, press F1 during setup, or see *Microsoft Windows for Workgroups Getting Started*.

If Setup isn't performing an update and if you didn't use the **setup /i** switch, then Setup performs Auto-detection to identify the specific type of computer, display, keyboard, mouse, and network devices. After the detection routines, Setup checks the type of display against the type of processor to make sure the display type is supported for running Windows 386 enhanced mode. (For example, EGABW and EGAMONO aren't supported in 386 enhanced mode.)

After the detection routines are completed, custom setup displays the configuration information so you can modify the settings if necessary. For example, you might want to specify a display type that requires an OEM-supplied disk with the necessary drivers.

Figure 3.1

System Configuration
screen for Windows
for Workgroups setup

```

Windows for Workgroups Setup
=====
Setup has determined that your system includes the following hardware
and software components. If your computer appears on the
Hardware Compatibility List with an asterisk, press F1 for help.

      System Information
      Computer:          MS-DOS System
      Display:           VGA
      Mouse:             Microsoft, or IBM PS/2
      Keyboard:         Enhanced 101 or 102 key US and Non US keyboards
      Keyboard Layout:  US
      Language:         English: (American)

No Changes: The above list matches my computer.

If all the items in the list are correct, press ENTER to indicate
"No Changes." If you want to change any item in the list, press the
UP or DOWN ARROW keys to move the highlight to the item you want to
change. Then press ENTER to see alternatives for that item.

ENTER=Continue      F1=Help      F3=Exit

```

Setup also searches for TSRs and drivers that are known to interfere with installation. Where possible, Setup automatically removes these drivers. You may see a warning at this point that you must remove incompatible drivers or terminate-and-stay-resident (TSR) programs for Setup to proceed. Setup also checks for disk compression software such as Stacker. For more information, see "Troubleshooting TSRs During Setup," in Chapter 14, "Troubleshooting Windows for Workgroups."

When you confirm the system configuration settings, Setup copies the appropriate files required to start Windows for Workgroups in standard mode and prepares a minimum SYSTEM.INI file to start Windows mode setup. If

you are installing Windows for Workgroups on a local drive, Setup copies SETUP.EXE to the WINDOWS directory and SETUP.INF to the Windows SYSTEM subdirectory. Next, it copies files listed in the **[winexec]**, **[windows]**, and **[windows.system]** sections of SETUP.INF and the appropriate Windows for Workgroups logo code and the logo data files. Then Setup renames the SYSTEM.SRC file as SYSTEM.INI.

Modifying SYSTEM.INI to Support Your Hardware

To prepare for Windows mode setup, two sections in the SYSTEM.INI file are modified to support the hardware configuration:

- The **[boot]** section of SYSTEM.INI is modified to update the following entries: **system.driv=**, **display.driv=**, **keyboard.driv=**, **mouse.driv=**, **fonts.fon=**, **oemfonts.fon=**, **sound.driv=**, **comm.driv=**, **fixedfon.fon=**, **language.dll=**, and **network.driv=**.
- The **[keyboard]** section of SYSTEM.INI is modified, updating the following entries: **keyboard.dll=**, **oemansi.bin=**, **type=**, and **subtype=**. For most U.S. installations, the **type=** entry is the only entry that is modified.

The **shell=** entry in the **[boot]** section of SYSTEM.INI becomes **shell=setup** (which is specified in the **[shell]** section of SETUP.INF). The **[boot.description]** section of SYSTEM.INI is filled with the text strings used by Setup to show the current configuration in a format that can be understood by non-technical Windows for Workgroups users.

The files listed in the **[boot]** section of SYSTEM.INI are now copied to the Windows for Workgroups SYSTEM subdirectory. The names of the actual drivers depend on the computer configuration. Usually, the following files are copied: SYSTEM.DRV, DISPLAY.DRV, KEYBOARD.DRV, MOUSE.DRV, FONTS.FON, OEMFONTS.FON, SOUND.DRV, COMM.DRV, and FIXEDFON.FON.

After these system files are copied, the KEYBOARD.DLL file and the OEMANSI.BIN file (if necessary) are copied to the Windows for Workgroups SYSTEM subdirectory.

MS-DOS mode setup next modifies SYSTEM.INI for settings to complete installation. The **286grabber=** entry in the **[boot]** section is modified first, and if the system configuration will support 386 enhanced mode, the **386grabber=** entry is modified. For 80386sx and higher processors, entries are placed in the **[386enh]** section of SYSTEM.INI to support the virtual display device, EBIOS, and any computer-specific settings such as **emmexclude=** values.

Last Steps for MS-DOS Mode Setup

Before Windows for Workgroups runs, Setup builds WIN.COM in the WINDOWS directory by joining the WIN.CNF file with the correct logo code and logo data files from the Windows for Workgroups installation disks.

The last step before starting the Windows mode phase of setup is to build the command line to pass to Setup. (Command-line parameters are switches following an executable name that tell a program to run in a specific way.) First, the name of the application to run (**win.com**) is read from the **startup=** entry in the **[data]** section of SETUP.INF.

Next, Setup reads the default command line from the **execcmd=** entry in the **[winexec]** section of SETUP.INF, which specifies the Windows kernel to use for Windows mode setup.

Finally, Setup calls the MS-DOS **exec** command to start WIN.COM using the Setup command-line parameters. At the completion of MS-DOS mode setup, WIN.COM loads, first displaying the logo screen and determining whether Windows for Workgroups can run in standard mode to complete setup. If HIMEM.SYS is present and there is at least 256K of conventional memory and 192K of extended memory, Windows for Workgroups starts in standard mode. If HIMEM.SYS is not present, then XMSMMGR.EXE is also run, to serve as the extended memory manager for setup.

If MS-DOS mode setup fails, or if Windows mode setup hangs, Setup probably failed to detect the correct hardware on your system (or you provided Windows for Workgroups with incorrect hardware settings). For more information about Setup failure, see Chapter 14, "Troubleshooting Windows for Workgroups."

Windows Mode Setup

When MS-DOS mode setup starts Windows, all of the essential Windows for Workgroups files are loaded, including the core executable files (KRNL286.EXE or KRNL386.EXE, User, and GDI), dynamic-link libraries, the drivers, and the fonts. The Windows background color is drawn on the screen, followed by the hourglass mouse cursor (if a mouse or pointing device is present), and finally by the Windows setup screen. During Windows mode setup, Windows for Workgroups automatically runs in standard mode.

Getting Ready for Windows Mode Installation

While Windows for Workgroups is starting for the first time, Setup is loading SETUP.INF into memory and searching for the **[blowaway]** section. Setup trims down the file at that section, discarding unnecessary sections to minimize the memory overhead for the file. For more information on **[blowaway]**, see Chapter 5, "The Windows for Workgroups Setup Information Files."

Setup then checks the **shell=** entry in the **[boot]** section of SYSTEM.INI to make sure that it is the Windows boot shell. If Setup isn't the boot shell, then Setup performs a series of maintenance tasks and prompts you to change the system settings. If Setup is the boot shell, then it identifies the path of the WINDOWS directory, reads the command-line parameters from MS-DOS mode setup, and sets flags for later use.

Next, Setup prompts you for a user name and company name, which is the identifying name that will appear with the registration serial number in the About Program Manager dialog box. If you have previously installed Windows for Workgroups using the same disks, a reminder appears instead of the registration dialog box.

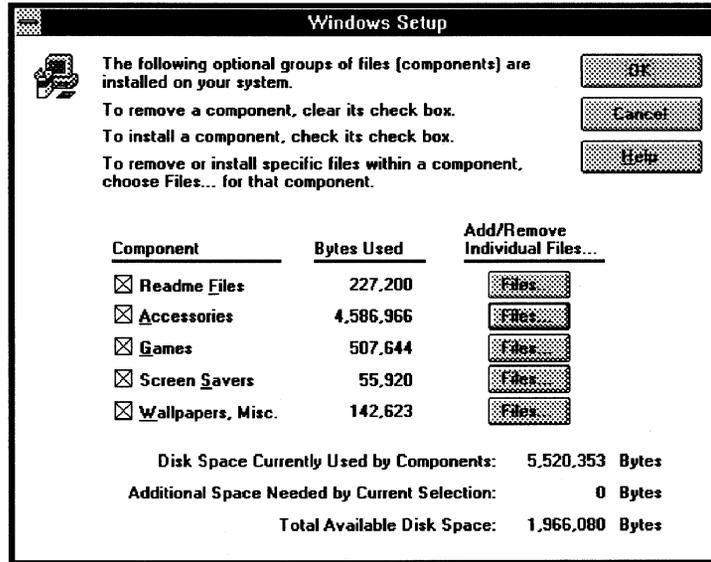
Windows for Workgroups Setup displays the Windows Setup dialog box and prompts you for your full name, the name for your computer, which is used to uniquely identify the computer in a workgroup, and the name of the workgroup that you want your computer to belong to.

Completing Custom Application Installation

During custom setup, the Add/Remove Files dialog box is displayed at this point, asking if Setup should install printers and optional applications. You are then presented with the Virtual Memory dialog box if your computer has an Intel 80386 (or higher) CPU, and asked to create a swap file to use for virtual memory in 386 enhanced mode. Setup saves the response to these options for later use. Windows mode setup then displays a dialog box that shows the total amount of disk space required for each Windows for Workgroups component option, allowing you to further customize each choice.

Figure 3.2

Add/Remove dialog box for Windows Setup. This dialog box does not appear if you use `setup /n` for installation.



Setup now uses information from the “copy files” sections of SETUP.INF to copy the Windows for Workgroups files and applications to the hard disk. The following sections in SETUP.INF list the files to be copied, depending on the system:

- **[win.copy]** if the system has an 80286 processor
- **[win.copy.net]** if installing from a network drive
- **[win.copy.win386]** if the system has an 80386 processor

Updating the System Files

Setup updates the system files at this point. Depending on settings in SETUP.INF, Setup will do the following:

- Copy the virtual mouse drivers and update the virtual mouse driver entry in the **[386enh]** section of SYSTEM.INI.
- Copy the virtual network drivers and the related Help files, plus any optional network files.
- Update system support sections of the WIN.INI and SYSTEM.INI files.
- Copy the virtual display device files, if required, along with the matching application fonts (such as EGA80WOA.FON), and modify SYSTEM.INI, if necessary. For more information, see “Display Driver Sections” in Chapter 5, “Windows for Workgroups Setup Information Files.”

Setup also copies the TrueType® fonts and other screen fonts, matching the resolution of the display driver with those of a given font set. The font sets you choose are copied to the Windows for Workgroups SYSTEM subdirectory, and the WIN.INI font entries are updated, if necessary. For details, see “Fonts Files” in Chapter 5, “The Windows for Workgroups Files.” When all other files are copied, Setup installs the specified applications and printers.

Setup displays the Printers dialog box and prompts you to add printers to your Windows for Workgroups configuration (if you told Setup that you wanted to configure printers). Setup then adds the selected printer drivers and configure them.

Setup then configures the Windows for Workgroups networking components. If you are upgrading a previous configuration, Setup upgrades the networking components, upgrades the network adapter card drivers, and upgrades a secondary net configuration (if defined).

Next, Setup displays the Network Adapters dialog box and prompts you for information to configure your workstation to use a network adapter card. This dialog box can be used to add or remove network adapter cards from your configuration, and to assign or add additional transport protocols to the configured network adapter cards.

Windows for Workgroups Setup then displays the Compatible Networks dialog box and prompts you to select one of the supported compatible network configurations, if applicable to your workgroup environment. Windows for Workgroups 3.1 provides drivers to integrate Windows for Workgroups with a Microsoft LAN Manager network and a Novell NetWare network. Selecting a compatible network configuration instructs Setup to copy additional components necessary to provide interoperability with Windows for Workgroups. For more information on the network integration components provided with Windows for Workgroups, see Chapter 8, “Network Integration with Microsoft LAN Manager and Novell NetWare.”

After completing these steps, Setup modifies the **shell=** entry in the **[boot]** section of SYSTEM.INI, changing it from **setup** to **progmán** (the Windows Program Manager). Then Setup starts Program Manager and sends DDE messages to it to create groups and install group items as defined in the **[progmán.groups]** and **[new.groups]** sections of SETUP.INF.

If you told Windows for Workgroups to install applications, Setup displays the Set Up Applications dialog box and prompts you to choose the location where Setup should search for applications to install.

Setup then displays a dialog box from which you can run the Windows tutorial, if you want to.

Checking for SMARTDrive and Incompatible Drivers

Setup searches CONFIG.SYS and removes any incompatible drivers and adds or updates entries for drivers such as HIMEM.SYS to load the updated drivers provided with Windows for Workgroups 3.1.

If SMARTDrive is not installed in the system and the system has at least 512K of extended memory, Setup installs SMARTDRV.EXE in the AUTOEXEC.BAT file. If your system requires double buffering, Setup also installs SMARTDRV.EXE in your CONFIG.SYS file. If RAMDRIVE.SYS is installed in CONFIG.SYS, Setup updates it; but Setup won't install RAMDrive if it isn't already installed.

Setup also checks the list of incompatible drivers in SETUP.INF. If any of these incompatible drivers are listed in CONFIG.SYS, or if the MS-DOS **append**, **subst**, or **join** commands are in AUTOEXEC.BAT, Setup displays a message explaining that incompatible drivers or commands have been found. Also, Setup checks the **files=** entry in CONFIG.SYS to make sure that the number of file handles is at least 30.

After these modifications in custom setup, the Setup program displays a message asking if the changes to CONFIG.SYS and AUTOEXEC.BAT should be made automatically or if you want to review and edit the files. Depending on your response, Setup takes the appropriate action and automatically saves the previous versions as backup files. In express setup, the changes are always made automatically.

Updating from Earlier Versions of Windows

For updating from Windows 3.0, Setup makes changes to WIN.INI and SYSTEM.INI based on entries in **[ini.upd.31]** and other sections of SETUP.INF. For more information, see "Miscellaneous Sections" in Chapter 5, "Windows for Workgroups Setup Information Files."

Completing Setup

To complete the installation of the Windows for Workgroups files, Setup renames WINVER to WINVER.EXE. Windows for Workgroups is now installed on your system.

Setup displays a final message asking you to restart your computer so that all the changes can take effect or to return to the MS-DOS command prompt.

Setting Up Windows for Workgroups for a Network Installation

The rest of this chapter describes how to place Windows for Workgroups files on a network server, how to set up workstations to access a shared copy of Windows for Workgroups, and how to use the automated setup option to efficiently set up Windows for Workgroups on multiple computers.

If you want to set up Windows for Workgroups over a network, set up shared copies of Windows for Workgroups, or you are setting up multiple workstations, read these sections.

Note If you want to use the Windows for Workgroups Setup disks to set up Windows for Workgroups on a single workstation's hard disk, you don't need to read the following sections.

General Guidelines

You have many options for setting up Windows for Workgroups on a network, depending on your environment:

- Do you want to set up a separate copy of Windows for Workgroups on each workstation? (This requires 10.5 to 15 MB on each hard disk.) Or do you want users to share most of the Windows for Workgroups files on a network server? (This requires 21 MB on the shared network server disk and 1.2MB on each workstation hard disk.)
- Do you want to run setup for each workstation from the Windows for Workgroups setup disks or from a network server?
- How interactive do you want the setup process to be? Do you want the user to configure options, or do you want to automate the process so that all options are predetermined?
- Do you want to set up a customized version of Windows for Workgroups? You can first copy Windows for Workgroups files to the network, then make modifications as described in Chapter 5, and finally set up Windows for Workgroups for users' workstations. For details, see "Modifying .INF Files for Custom Installations" in Chapter 5, "Windows for Workgroups Setup Information Files."

Note You must have a license for each workstation on which you want to use Microsoft Windows for Workgroups.

After answering the preceding questions, you can decide which setup options to use:

- **setup /a** (Windows administrative setup) copies the Windows for Workgroups files to a network server so that users can run the Setup program from the network. This option replaces EXPALL.BAT that was used in Windows 3.0.
- **setup /n** (Windows network setup) sets up a workstation to run a shared copy of Windows for Workgroups. Only a few Windows for Workgroups files are copied to the workstation's hard disk (or the user's personal network directory) so that the user can customize the Windows for Workgroups environment.
- **setup /h** (Windows automated setup) uses an automated Setup routine for easy installation. For more information, see "Using Automated Setup: Setup /h" later in this chapter.

Here are a couple of other points that you should keep in mind when considering your Windows for Workgroups installation choices:

- *Microsoft Workgroup Connection* can be used to bootstrap workstations that don't presently have network client software installed, but that are configured with a network adapter card. Once *Workgroup Connection* is installed on a workstation, Windows for Workgroups can be installed over the network from a configuration that was created by using the **setup /a** administrative setup option.
- Running a shared copy of Windows for Workgroups from a network file server requires that the network redirector be started on the workstation prior to running Windows for Workgroups. Starting the network software outside Windows for Workgroups will use more conventional memory on an Intel 386-based (or higher) computer, than loading the protected-mode networking components from inside Windows for Workgroups.

The rest of this section describes administrative setup and network setup. The next section in this chapter describes automated setup.

Placing Windows for Workgroups Files on a Network Server: Setup /a

You can use **setup /a** (the Windows administrative setup option) to place Windows for Workgroups files on a network server. The administrative option does not set up a usable copy of Windows for Workgroups, but simply transfers the files from the Windows for Workgroups Setup disks to a network drive. The files are automatically expanded, renamed, and marked as read-only when they are copied to the network drive. The shared Windows for Workgroups files must be marked as read-only (even on a read-only network drive) so that they can be used by more than one user or application at a time.

Important Before you run the Setup program, turn off any network messaging services or TSRs that allow the network or another user to send a message to your screen. These popup programs can cause Windows for Workgroups Setup to fail if a message is received while Windows for Workgroups is being installed.

To place Windows for Workgroups files on a network server

1. From your workstation, connect to the network drive where you want to copy the Windows for Workgroups files. Ensure that you have write access in that directory.
2. Insert the Windows for Workgroups 3.1 Disk #1 in a floppy-disk drive, and change to the drive letter of the drive.
3. At the command prompt, type **setup /a** and press ENTER.
4. Follow the instructions on your screen.

You are prompted to specify the network drive and directory you want the Windows for Workgroups files copied to (for example, W:\WINSHARE). You're also prompted to specify group registration information (group name and company name), which is stored and used when individual workstations are set up.

5. If your network requires it, make sure that all of the Windows for Workgroups files in this directory are marked as shareable. For more information, see your network documentation.

If you want to copy Windows for Workgroups files to other shared network servers in your organization, run **setup /a** from the shared directory where you copied Windows for Workgroups files. This copies files and mark them as read-only in the new directory. You'll be prompted for the company name and group name again, allowing you to specify a different company or group name for each shared WINDOWS directory in your organization.

Note Don't use the MS-DOS `copy *.*` command at the command prompt to copy Windows for Workgroups files to another network server, and don't delete the 8-byte files named **disk1**, **disk2**, **disk3**, and so forth, from the shared network directory. These files are required to run `setup /a` from the network.

After you place Windows for Workgroups files on a network server, users can connect to this server and run `setup /n` to set up Windows for Workgroups for their workstations.

Before you instruct users to set up Windows for Workgroups, decide whether you want them to share Windows for Workgroups files on the network server or copy all Windows for Workgroups files to their workstations, and whether you want them to use automated setup. You may also want to customize setup to add custom Program Manager groups and applications to the users' Windows for Workgroups environment. Or you might customize setup to modify the Program Manager capabilities available to users. For more information, see "PROGMAN.INI: The Program Manager Initialization File" in Chapter 6, "Windows for Workgroups Initialization Files."

Installing Windows for Workgroups over a Network

It is possible to install Windows for Workgroups on workstations from a network shared directory (versus using the floppy disks provided in the box) to improve the speed at which Windows for Workgroups is set up. In order to do this, however, the Windows for Workgroups files must first be expanded on an accessible file share (using the `setup /a` option), and the workstation where Windows for Workgroups is to be installed must be set up to have access to a network file share where the Windows for Workgroups files have been placed.

If the workstation that you wish to install Windows for Workgroups isn't presently running any network operating system, but has a properly configured network adapter card, the *Microsoft Workgroup Connection* product may be used to load the minimal networking components on an MS-DOS workstation using a one-disk installation procedure. Once *Workgroup Connection* is installed, the user can connect to the file share where the Windows for Workgroups files are stored.

The user may then run the Setup program from the network to install the Windows for Workgroups files to the local workstation. The Setup program will install the Windows for Workgroup networking components on the local workstation and will update the CONFIG.SYS file and the AUTOEXEC.BAT files to refer to the network files provided by Windows for Workgroups. After Windows for Workgroups is completely installed, the user can choose to reboot the computer and to remove *Workgroup Connection* from memory in preparation for loading Windows for Workgroups.

Setting Up a Shared Copy of Windows for Workgroups: Setup /n

Users can set up their workstations to access a shared copy of Windows for Workgroups by running **setup /n** (the network setup option). **Setup /n** copies some of the Windows for Workgroups files to the user's personal WINDOWS directory. The user can customize Windows for Workgroups according to personal preferences while still sharing most of the Windows for Workgroups files with other users.

To set up a shared copy of Windows for Workgroups on a workstation

1. From the workstation, connect to the network drive that contains the Windows for Workgroups files.
2. Change to the directory where the Windows for Workgroups files are located.
3. Type **setup /n** at the command prompt and press ENTER.
4. Follow the instructions on the screen.

You are prompted for the directory where you want to place the user files. This directory can either be on the local hard disk (for example, C:\WINDOWS) or on a private network directory (for example, U:\USERNAME). Specify a private directory on a network server if you are setting up a diskless workstation.

Setup copies only the files that are relevant to the user's system and desktop preferences—for example, the group and initialization files (.GRP and .INI files). All other Windows for Workgroups files are located in the shared network directory. Setup adds both the private WINDOWS directory and the shared network directory to the path in the user's AUTOEXEC.BAT file.

Note You can't run **setup /n** from the installation disks. You can modify setup so that it always sets up a shared copy of Windows for Workgroups on a workstation, even if a user specifies a different setup option. To do this, specify **netsetup=true** in the [data] section of SETUP.INF. For more information, see Chapter 5, "Windows for Workgroups Setup Information Files." There are no custom options available to the user when you force network setup in this way.

Using Automated Setup: Setup /h

The Windows automated setup option (also called batch mode setup) uses information defined in a system-settings file to install Windows for Workgroups quickly and easily, with little or no user interaction. Automated setup is useful if you have to set up many workstations or if you want users to be able to set up on their own workstations without having to make any system configuration choices about hardware components and printers, or choices about which optional Windows for Workgroups components to install.

Automated setup can be used for setting up a shared copy of Windows for Workgroups or a nonshared copy. You can use it if you are setting up Windows for Workgroups from a network drive or from the Windows for Workgroups Setup disks.

Note Before you run the Setup program, turn off any network messaging services or TSRs that allow the network or another user to send a message to your screen. These popup programs can cause Windows for Workgroups Setup to fail if a message is received while Windows for Workgroups is being installed.

To use the automated setup option

1. Create a system-settings file for each workstation configuration and place the system-settings files in a directory where users can open and read files or copy a file to the workstation disk.

For details, see the following section, "Creating the System-Settings File."

2. From the workstation, either you or a user can set up Windows for Workgroups by using the automated setup option. If you are setting up Windows for Workgroups from a network, connect to the network drive and change to the directory where the Windows for Workgroups setup files are located. If you are setting up Windows for Workgroups from the Windows for Workgroups setup disks, change to the letter of the floppy disk drive.
3. If you want to run a shared copy of Windows for Workgroups from the network, type **setup /h:[drive:\path]filename /n**.

If you want to run a nonshared copy of Windows for Workgroups from a local drive or a network drive, type **setup /h:[drive:\path]filename**.

The *filename* is the name of the system-settings file that contains details about the system configuration. Include a path if the system-settings file is not in the directory from which you are setting up Windows for Workgroups.

For example, you might create a system-settings file named SETUPVGA.SHH in the same directory as the Setup program. To set up a complete copy of Windows for Workgroups on the workstation, at the command prompt type `setup /h:setupvga.shh` and press ENTER.

If you want to run a shared copy of Windows for Workgroups from the network, using this same example, connect to the network drive and directory where the Windows for Workgroups setup files are located. At the command prompt, type `setup /h:setupvga.shh /n` and press ENTER.

Creating the System-Settings File

Windows for Workgroups comes with a system settings template file called SETUP.SHH, which you can copy or modify to create a custom system-settings file. The SETUP.SHH file is located on Windows for Workgroups 3.1 Disk #1 and is copied to the shared Windows for Workgroups directory when you use `setup /a`. You can modify this file with any text editor that can save unformatted text (ASCII) files.

Note When you create the .SHH file, use only ANSI characters. The .SHH file will be interpreted as containing characters from the ANSI character set, not from the code page character set currently installed on your system.

The system-settings file is summarized in the following table.

Section	Purpose
[sysinfo]	Specifies whether System Information screen will appear during setup.
[configuration]	Specifies the various devices on the user's system.
[windir]	Specifies where to put Windows for Workgroups files.
[userinfo]	Specifies the user and company name.
[network]	Specifies Windows for Workgroups network options and network adapters.
[NetcardProtocol]	Specifies PROTOCOL.INI parameters for the network adapter specified in the [network] section.
[TransportProtocol]	Specifies PROTOCOL.INI parameters for the network transport specified in the [network] section.
[dontinstall]	Specifies components that should not be installed.
[options]	Specifies various Setup options such as viewing README files, setting up existing applications, and starting the Windows tutorial.
[printers]	Specifies printers to set up.
[endinstall]	Specifies whether Setup modifies the CONFIG.SYS and AUTOEXEC.BAT files and whether the system is rebooted after Windows for Workgroups is installed.



The entries for these sections are described later in this section. Many of the entries require values from the Windows for Workgroups SETUP.INF, CONTROL.INF, NETWORK.INF, and WIN.INI files. When you use **setup /a** to place Windows for Workgroups files on a network server, these files are copied to the shared WINDOWS directory. The master WIN.INI and SYSTEM.INI files are called WIN.SRC and SYSTEM.SRC in this directory. You can also find SETUP.INF on Windows for Workgroups 3.1 Disk #1, and NETWORK.INF on Disk #2. You can print or view all of these text files with any text editor.

For details about specific items in SETUP.INF, WIN.SRC, and SYSTEM.SRC, see the related entries in Chapter 5, “Windows for Workgroups Setup Information Files.”

[sysinfo]

This section specifies whether the System Information screen is displayed during MS-DOS mode setup so the user can confirm or change the settings before continuing with Setup. Specify **showsysinfo=yes** to display the screen, or **showsysinfo=no** if you don't want to display the screen. (The default is **no**.)

For details about the System Information screen, see “MS-DOS Mode Setup” earlier in this chapter.

[configuration]

This section specifies the various devices on the user's system. If an entry for **[configuration]** is not included in the system-settings file, the detected or default device will be used.

If you are updating Windows for Workgroups, Setup ignores some of these entries and uses the devices that are already installed. If you want to force the update and override the installed device, type an exclamation point (!) before the **computer=**, **display=**, **mouse=**, and **network=** values, for example, **display=!vga**. Any **keyboard=**, **language=**, or **kblayout=** values specified will automatically override the installed device.

Use	To
computer=	Specify the type of computer, using a profile string from the [computer] section of SETUP.INF—for example: computer=ibm_compatible .
display=	Specify the type of display adapter, using a profile string from the [display] section of SETUP.INF—for example: display=vga .
mouse=	Specify the type of pointing device using a profile string from the [pointing.device] section of SETUP.INF—for example: mouse=ps2mouse .
network=	Specify the type of network and the version number using a profile string from the [network] section of SETUP.INF to specify the network—for example: network=wfnwnet sets up the detected version of Windows for Workgroups. If there is a [network_version] section in SETUP.INF, for example [wfnwnet.versions] , then use a number from that section to specify the version of the network you want to set up. Separate the version number from the network type with a slash (/)—for example: network=wfnwnet/00031000 sets up Windows for Workgroups version 3.10. Note: Windows for Workgroups 3.1 ignores this entry. Windows for Workgroups 3.1 is ALWAYS added. Use the MultiNetID= entry in the [network] section to specify a compatible network to use in addition to the Windows for Workgroups network.
keyboard=	Specify the type of keyboard using a profile string from the [keyboard.types] section of SETUP.INF—for example: keyboard=t4s0enha .
language=	Specify the language using a profile string from the [language] section of SETUP.INF—for example: language=enu . This setting defines the sort order used by Windows for Workgroups.
kblayout=	Specify the keyboard layout using a profile string from the [keyboard.tables] section of SETUP.INF—for example: kblayout=nodll .

[windir]

This section specifies where to put the user's Windows for Workgroups files. If Windows for Workgroups is already installed in the specified directory, Setup updates it. If you don't specify a directory or if the specified directory isn't valid, a dialog box appears during setup, asking for the directory in which to set up Windows for Workgroups.

For example, if you want to install Windows for Workgroups in the WINDOWS directory on drive C, specify **c:\windows** in this section. This setting overrides the value of **defdir=** in the **[data]** section of SETUP.INF.

[userinfo]

This section specifies the user and company name. The first line specifies the user name and is required unless you are setting up a shared copy of Windows for Workgroups (**setup /n**). The second line specifies the company name and is optional. Both names can be up to 30 characters long and must be enclosed in double quotation marks (" ") if they include blank spaces—for example:

```
"John Smith"
"My Company"
```

If you don't specify a user name, a dialog box appears during setup to ask for the name.

If you use **setup /n** to set up a shared copy of Windows for Workgroups, the **[userinfo]** section is ignored because the information was supplied when you used **setup /a** to copy Windows for Workgroups files to the server.

[network]

This section specifies the Windows for Workgroups network options and network adapters. Components of the Windows for Workgroups network configuration can be specified by changing the appropriate options in this section.

Use	To
ComputerName=	Specify the name of the computer to assign to this workstation (15 characters maximum). If you specify an asterisk (*) for the computer name, Setup retrieves the computer name from the current SYSTEM.INI file entry. If no computer name exists, Setup generates a unique computer name. If you specify a question mark (?) for the computer name, Setup generates a unique computer name. If you don't specify a computer name, a dialog box appears during setup prompting you for the computer name.
WorkgroupName=	Specify the name of the workgroup that the computer is to belong to (15 characters maximum). If you specify an asterisk (*) for the workgroup name, Setup retrieves the workgroup name from the current SYSTEM.INI file entry. If no workgroup name exists, Setup uses the default workgroup name WORKGROUP. If you specify a question mark (?) for the workgroup name, Setup generates the default workgroup name WORKGROUP. If you don't specify a workgroup, a dialog box appears during setup prompting for the workgroup name.
NetcardID=	Specify a network adapter driver. The value for the NetcardID= number comes from an entry in the [netcard] section of NETWORK.INF for the network adapter identified just after the descriptive text. If you don't specify a value for NetcardID= , Setup doesn't perform any network adapter card installation. Any previous network adapter card configuration remains unchanged.
TransportID=	Specify a network protocol for the network adapter. The transport ID number comes from an entry in the [transport] section of NETWORK.INF for the network transport identified just after the descriptive text. If you don't specify a value for TransportID= , or if TransportID=0 , Setup uses the default transport specified in NETWORK.INF.
MultiNetID=	Specify a secondary network. The MultiNet ID name comes from an entry in the [MultiNet] section of NETWORK.INF for the secondary network identified just after the description text. If you don't specify a value for the MultiNetID= entry, setup doesn't perform any secondary network installation. Any previous secondary network configuration will remain unchanged. The secondary network won't be installed unless a valid network adapter card was specified for the NetcardID= entry above.

[NetcardProtocol]

This section specifies PROTOCOL.INI parameters for the network adapter card specified in the **[network]** section. Parameters in this section will be merged into the PROTOCOL.INI file and will overwrite any default values and values that already exist in PROTOCOL.INI. The keywords **drivername=** and **lanabase=** are reserved for setup and will be ignored. Valid keywords for this section are the *key* field values from the PROTOCOL.INI section in NETWORK.INF. The specific entries for this section are dependent on the network adapter card being used. See the “PROTOCOL.INI” section in Chapter 6, “Windows for Workgroups Initialization Files,” for more information on the format for options in this section.

[TransportProtocol]

This section specifies PROTOCOL.INI parameters for the network transport specified in the **[network]** section. Parameters in this section will be merged into the PROTOCOL.INI file and will overwrite any default values and values that already exist in PROTOCOL.INI. The keywords **drivername=** and **lanabase=** are reserved for setup and will be ignored. Valid keywords for this section are the *key* field values from the PROTOCOL.INI section in NETWORK.INF. The specific entries for this section are dependent on the transport protocol being used. See the “PROTOCOL.INI” section in Chapter 6, “Windows for Workgroups Initialization Files,” for more information on the format for options in this section.

[dontinstall]

This section specifies optional Windows for Workgroups components that you don't want to set up. If this section is not present or is empty, all optional components are set up. You can choose not to install the components listed in the following table.

Specify	If you don't want to install
----------------	-------------------------------------

readmes	Readme files
accessories	Accessories
games	Games
screensavers	Screen savers
bitmaps	Background wallpapers

[options]

This section specifies whether a user can set up applications during setup and whether to start the Windows for Workgroups Tutorial at the end of setup. If you don't want any of these options, omit this section. You can specify in this section the entries shown in the following table.

Specify	To
setupapps	Set up applications interactively during setup. The Add/Remove File dialog box appears during setup so the user can choose which applications to install.
autosetupapps	Automatically set up all applications found on the user's hard disk.
tutorial	Start the Windows for Workgroups Tutorial at the end of setup.

You should not specify both **setupapps** and **autosetupapps**, but if you do, all applications on the hard disk will be set up.

[printers]

This section specifies which printers to install during setup. To install a printer, you specify a printer name and a port. Use one of the descriptive strings found in the **[io.device]** section of the CONTROL.INF file to specify the printer name. The printer name must be enclosed in double quotation marks (" ") if it contains blank spaces. Use one of the values found in the **[ports]** section of WIN.INI to specify the port. The printer name and the port must be separated by a comma. For example, to specify a HP LaserJet III printer connected to port LPT1, include this entry:

```
"HP LaserJet III",LPT1:
```

If you don't want to install any printers during setup, omit this section.

[endinstall]

This section specifies what should happen after Windows for Workgroups is successfully installed. You can specify whether Setup should modify CONFIG.SYS and AUTOEXEC.BAT and return to MS-DOS, restart Windows for Workgroups, or restart the system.

Specify	If you want Setup to
configfiles=modify	Make all changes to CONFIG.SYS and AUTOEXEC.BAT for the user. When Setup changes these files, the previous CONFIG.SYS and AUTOEXEC.BAT files are saved as CONFIG.OLD and AUTOEXEC.OLD in the root directory. (If CONFIG.OLD or AUTOEXEC.OLD are already present, the previous file will be renamed CONFIG.000 or AUTOEXEC.000, or the next number that does not conflict with an existing file.) If you are setting up Windows for Workgroups on a diskless workstation, Setup disregards this entry.
configfiles=save	Save proposed changes to the CONFIG.SYS and AUTOEXEC.BAT files in alternate files (CONFIG.WIN and AUTOEXEC.WIN) stored in the WINDOWS directory. The CONFIG.SYS and AUTOEXEC.BAT files are not changed; you must make these changes later. (If CONFIG.WIN or AUTOEXEC.WIN are already present, the previous file will be renamed CONFIG.000 or AUTOEXEC.000, or the next number that doesn't conflict with an existing file.)
endopt=exit	Return to MS-DOS on completion of setup.
endopt=reboot	Restart the user's computer on completion of setup.

Note **endopt=restart**, as supported in Windows 3.1, is not a valid option for Windows for Workgroups.

Sample System-Settings File

The following system-settings file installs Windows for Workgroups on a PC-compatible computer with these characteristics:

- A VGA display, Microsoft mouse, LAN Manager 2.0 Enhanced network, and 101-key U.S. keyboard.
- The System Information screen is displayed for confirmation, and Windows for Workgroups files are put in the C:\WINDOWS directory.
- The user will be prompted for the computer name to be used for the workstation and the computer will be assigned to the "Marketing" workgroup.
- The Intel EtherExpress™ 16 network adapter card driver is installed and configured for the NetBEUI transport protocol. Novell NetWare is chosen as the secondary network configuration, and the Windows for Workgroups NetWare integration components are installed.
- The default IRQ setting for the network adapter card is set to INT 5.
- The NetBEUI transport parameters, **sessions=** and **ncbs=**, are modified from their default values.
- The games are not installed.
- An HP LaserJet™ III printer is installed on LPT1.

- The CONFIG.SYS and AUTOEXEC.BAT files are modified by Setup.
- After Windows for Workgroups is set up, the computer is rebooted for the changes to take effect.

Sample Settings-File Contents:

```
[sysinfo]
showsysinfo=yes
```

```
[configuration]
computer=ibm_compatible
display=vga
mouse=ps2mouse
keyboard=t4s0enhha
language=enu
kblayout=nodll
```

```
[windir]
c:\windows
```

```
[userinfo]
"user's name"
"company name"
```



```
[network]
ComputerName=           ; Prompt user for computer name to assign
WorkGroup="Marketing"   ; Add computer to "Marketing" WorkGroup
NetcardID=301           ; Intel EtherExpress 16 network adapter
TransportID=1           ; NetBEUI transport protocol as the primary transport
MultiNetID=0x300        ; secondary network: Novell NetWare
```



```
[NetcardProtocol]
IRQ_LEVEL=5             ; Sets the default IRQ value to be 5 (LPT2)
```



```
[NetcardTransport]
sessions=10             ; these entries are used by the real-mode NetBEUI protocol
ncbs=20                 ; changes the default number of sessions for NetBEUI
                        ; changes the default number of ncbs (2 x sessions)
```

```
[dontinstall]
games
```

```
[options]
```

```
[printers]
"HP LaserJet III",LPT1:
```

```
[endinstall]
configfiles=modify
endopt=reboot
```

Maintaining Windows for Workgroups with Setup

You can use Setup to maintain Windows for Workgroups after the initial installation. When you run Setup from within Windows for Workgroups, you can review the same options as during original installation: choosing system settings, installing Windows-based applications, and adding or removing Windows for Workgroups components.

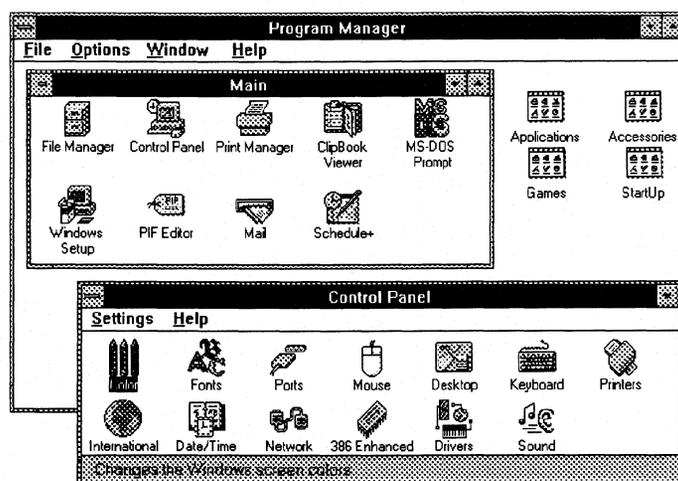
To make other changes to a system from within Windows for Workgroups you can:

- Choose the 386 Enhanced icon in Control Panel to define device contention settings, task scheduling for multitasking in 386 enhanced mode, and virtual memory settings.
- Choose the Ports icon or Printers icon in Control Panel to define settings for ports and devices.
- Choose the Network icon in Control Panel to define settings for the network configuration of your Windows for Workgroups workstation. This includes configuring network adapter cards and transport protocols.
- Drag icons to the Startup group in Program Manager to specify which applications will run whenever you start Windows for Workgroups.
- Use the Program Manager commands to create new groups. For example, use the Properties command in the File menu to change icons and specify other elements of an application's appearance on the desktop.

For details about using these Windows for Workgroups components, see the related chapters in the *Windows for Workgroups User's Guide*.

Figure 3.3

Program Manager
and Control Panel



**Chapter
4**

Windows for Workgroups Files

This chapter describes the purpose for each file installed by Windows for Workgroups 3.1 in the WINDOWS directory and the SYSTEM subdirectory.

For information about how to add to the list of files that are installed automatically with Windows for Workgroups, see “Modifying .INF Files for Custom Installations” in Chapter 5, “Windows for Workgroups Setup Information Files.”

Related information

- *Windows for Workgroups Resource Kit*: Chapter 5, “Windows for Workgroups Setup Information Files;” Chapter 6, “Windows for Workgroups Initialization Files.”

Contents of This Chapter

About the Windows for Workgroups Files.....	4-2
WIN.COM.....	4-2
The Core Files	4-3
Drivers, Fonts, and International Support Files.....	4-3
Driver Files.....	4-3
Font Files.....	4-9
International Support Files.....	4-12
MS-DOS Support Components of Windows for Workgroups	4-12
MS-DOS Driver Files	4-13
WinOldAp and the Grabber Files.....	4-14
Files for Standard Mode.....	4-16
Files for 386 Enhanced Mode.....	4-16
Windows for Workgroups Applications, Setup, and Other Files.....	4-17
Files for Windows for Workgroups Applications.....	4-17
Setup-Related Files	4-19
Other Files.....	4-20
Files You Can Delete.....	4-22

About the Windows for Workgroups Files

When Microsoft Windows for Workgroups runs, it performs all operating system duties except file system management, which MS-DOS still performs. Windows for Workgroups calls functions that are stored in a variety of executable files, driver files, and other dynamic-link libraries to manage the display, keyboard, and other devices, to manipulate the network, to manage memory, and to execute programs.

The kinds of files that make up Windows for Workgroups include:

- The WIN.COM file.
- The core dynamic-link libraries (kernel files, USER, and GDI) that contain the code and data for the Windows functions.
- The font files and the drivers for keyboard, display adapter, system, mouse, printers, networks, multimedia, and other devices.
- The files that provide MS-DOS support components for Windows.
- The files that provide the network support functionality—including the ability to share files and printers with other workstations.
- The Windows applications files and other files such as shells, utilities, and accessories.

For instructions on how to expand any files from the Windows installation disks, see Flowchart 1.7 in Appendix C, “Flowcharts for Troubleshooting Windows for Workgroups.” For additional information about the Windows files, see the manuals for the Microsoft Windows *Software Developer's Kit* and Microsoft Windows *Device Driver Kit*.

WIN.COM

WIN.COM loads Windows for Workgroups. It checks the computer type, memory configuration, and device drivers to determine which mode is appropriate to start Windows for Workgroups. To start Windows for Workgroups, there needs to be sufficient memory, an XMS driver present (such as HIMEM.SYS), and processor support for standard mode (80286 or higher) or 386 enhanced mode (80386 or higher).

After WIN.COM determines the appropriate operating mode, it uses the MS-DOS `exec` command to execute one of the following files, which in turn loads Windows for Workgroups:

- DOSX.EXE for standard mode
- WIN386.EXE for 386 enhanced mode

To build Windows for Workgroups, WIN.COM brings together a number of files:

- Core files
- Drivers
- Fonts and language support files
- Support files for MS-DOS-based applications
- MS-DOS support and various mode-specific files
- Windows for Workgroups network support files

The Core Files

Three files make up the Windows core components: Kernel, User, and GDI.

- The kernel files (KRNL286.EXE or KRNL386.EXE) control and allocate all the computer resources to manage memory, load applications, and schedule program execution and other tasks.
- USER.EXE creates and maintains windows on the screen, carrying out all requests to create, move, size, or remove a window. USER.EXE also handles requests regarding the icons and other components of the user interface. USER.EXE directs input to the appropriate application from the keyboard, mouse, and other input sources.
- GDI.EXE controls the Graphics Device Interface, which executes graphics operations that create images on the system display and other devices.

Drivers, Fonts, and International Support Files

Driver Files

Drivers make device independence possible for Windows-based applications, providing the hardware-specific interface between physical devices and Windows. Setup can install several kinds of drivers for Windows, such as:

Comm drivers	Mouse drivers	Printer drivers
Display drivers	Multimedia drivers	Sound drivers
Keyboard drivers	Network drivers	System drivers

The multimedia and printer drivers are optional. Also, Setup can install drivers to support virtual machines in 386 enhanced mode, as described in “Files for 386 Enhanced Mode” later in this chapter.

System Driver Files

The system driver provides support for the system timer, information about system disks, and access to OEM-defined system hooks. There are two system drivers shipped with Windows for Workgroups:

- SYSTEM.DRV, the driver for most hardware systems
- HPSYSTEM.DRV, the HP Vectra® system driver for standard mode

Keyboard Driver Files

The keyboard drivers shipped with Windows for Workgroups support keyboard input:

- KEYBOARD.DRV for standard keyboards, installed by default
- KBDHP.DRV for all Hewlett-Packard® machines

The keyboard driver is a standard driver for all systems worldwide. Windows for Workgroups is also compatible with international keyboards, including foreign symbols, by using keyboard tables to refer to a language library.

Keyboard table	Language library
KDBBE.DLL	Belgian keyboard
KBDCA.DLL	French-Canadian keyboard
KBDDA.DLL	Danish keyboard
KBDDV.DLL	U.S.-Dvorak keyboard
KBDFC.DLL	Canadian multilingual keyboard
KBDFI.DLL	Finnish keyboard
KBDFR.DLL	French keyboard
KBDGR.DLL	German keyboard
KBDIC.DLL	Icelandic keyboard
KBDIT.DLL	Italian keyboard
KBDLA.DLL	Latin American keyboard
KBDNE.DLL	Dutch keyboard
KBDNO.DLL	Norwegian keyboard
KBDPO.DLL	Portuguese keyboard
KBDSF.DLL	Swiss-French keyboard
KBDSG.DLL	Swiss-German keyboard
KBDSP.DLL	Spanish keyboard
KBDSW.DLL	Swedish keyboard
KBDUK.DLL	British keyboard
KBDUS.DLL	U.S. keyboard
KBDUSX.DLL	U.S.-International keyboard

The .DLL filename extension indicates that the file is a dynamic-link library.

Mouse Driver Files

The mouse drivers shipped with Windows for Workgroups support pointing devices for use with Windows for Workgroups and Windows-based applications.

Driver	Supported mouse or pointing device
HPMOUSE.DRV	Hewlett-Packard mouse (HP-HIL)
LMOUSE.DRV	Logitech™ Serial mouse
MSC3BC2.DRV	Mouse Systems COM2/3 button mouse
MSCMOUSE.DRV	Mouse Systems Serial/Bus mouse
MOUSE.DRV	Logitech Bus or PS/2 style, Microsoft, or IBM PS/2 mouse
NOMOUSE.DRV	No mouse attached to system

For information about the related MS-DOS mouse drivers, see “MS-DOS Support Components of Windows for Workgroups” later in this chapter.

Display Driver Files

The display drivers shipped with Windows for Workgroups support the system display adapter and the cursor for the pointing device. The display driver, however, does not support MS-DOS-based applications running in a full screen, because these applications write directly to video.

Driver	Supported display adapter
8514.DRV	8514/a
EGA.DRV	EGA
EGAHIBW.DRV	EGA with 128K RAM
EGAMONO.DRV	EGA monochrome display
HERCULES.DRV	Hercules® monochrome display
MMTLLO.DRV	ET4000 (small fonts)
MMTLHI.DRV	ET4000 (large fonts)
OLIBW.DRV	Olivetti/AT&T monochrome or PVC display
PLASMA.DRV	COMPAQ® Portable plasma
SUPERVGA.DRV	Super VGA (800x600 - 16 colors)
TIGA.DRV	TIGA
VGA.DRV	VGA
VGAMONO.DRV	VGA monochrome display
V7VGA.DRV	Video Seven VGA with 512K (FastWrite, VRAM, 1024i, and compatibles)
XGA.DRV	XGA

Other Driver Files

The communications driver, COMM.DRV, supports serial and parallel device communications.

The Advanced Power Management device driver, POWER.DRV, supports the power management features of laptop and notebook personal computers.

Printer Driver Files

Printer drivers support output to the printer. Some of the printer drivers shipped with Windows for Workgroups have a soft-font installation utility. Related files also include help files for the printer drivers and soft-font installers. In Windows for Workgroups, many dot-matrix drivers have been replaced by a universal printer driver. Other drivers have been updated for performance and to support TrueType fonts.

Printer driver	Representative printer
CANON10E.DRV	Canon® Bubble-Jet BJ-10e
CANON130.DRV	Canon Bubble-Jet BJ-130e
CANON330.DRV	Canon Bubble-Jet BJ-300/330
CIT24US.DRV	Citizen 24-pin
CIT9US.DRV	Citizen 9-pin
CITOH.DRV	C-Itoh 8510 or AT&T® 470/475
DICONIX.DRV	Kodak® Diconix
DM309.DRV	Olivetti® DM 309
DMCOLOR.DLL	Universal color printing support library
EPSON24.DRV	Epson® 24-pin
EPSON9.DRV	Epson 9-pin
ESCP2.DRV	Epson ESCP2 dot matrix
EXECJET.DRV	IBM ExecJet®
GENDRV.DLL	Generic library
HPDSKJET.DRV	Hewlett-Packard DeskJet® Series
HPPCL.DRV	HP LaserJet II Series
HPPCL5MS.DRV	HP LaserJet III Series
HPLOT.DRV	HP Plotter
IBM4019.DRV	IBM Laser Printer 4019
IBM5204.DRV	IBM Quickwriter® 5204
IBMCOLOR.DRV	IBM Color
LBPII.DRV	Canon LBP-8 II
LBPIII.DRV	Canon LBPIII
NEC24PIN.DRV	NEC® 24-pin
OKI24.DRV	Okidata® 24-pin
OKI9.DRV	Okidata 9-pin
OKI9IBM.DRV	Okidata 9-Pin IBM Model
PAINTJET.DRV	HP PaintJet®
PANSON24.DRV	Panasonic® 24-pin
PANSON9.DRV	Panasonic 9-pin
PROPRINT.DRV	IBM Pro series
PROPRN24.DRV	IBM Pro 24 pin series
PS1.DRV	IBM PS/1
PSCRIPT.DRV	Postscript (PSCRIPT.HLP is the Help file)
QWIII.DRV	IBM QuietWriter® III
THINKJET.DRV	HP ThinkJet® (2225 C-D)
TI850.DRV	TI 850/855
TOSHIBA.DRV	Toshiba® p351/1351

Printer driver	Representative printer	(continued)
-----------------------	-------------------------------	--------------------

TTY.DRV	Generic / Text only (TTY.HLP is the Help file)
UNIDRV.DLL	Microsoft universal library (UNIDRV.HLP is the Help file)

The following files are soft-font installers for specific printers.

Soft-font installer	Related printer
----------------------------	------------------------

CAN_ADF.EXE	Canon LBP-8 II or LBP111
FINSTALL.DLL	HPPCL5/MS (FINSTALL.HLP is the Help file)
SF4019.EXE	IBM Laser Printer 4019

The following files provide additional PostScript® description information for specific printers.

PostScript description	Related printer
-------------------------------	------------------------

40291730.WPD	IBM LaserPrinter 4029 (17 fonts)
40293930.WPD	IBM LaserPrinter 4029 (39 fonts)
DEC1150.WPD	Digital DEClaser 1150
DEC2150.WPD	Digital DEClaser 2150
DEC2250.WPD	Digital DEClaser 2250
DEC3250.WPD	Digital DEClaser 3250
DECCOLOR.WPD	Digital ColorMate PS
DECLPS20.WPD	Digital LPS Print Server
EPL75523.WPD	Epson EPL-7500
HERMES_1.WPD	Hermes H 606 PS (13 Fonts)
HERMES_2.WPD	Hermes H 606 PS (35 Fonts)
HPELI523.WPD	HP LaserJet IIISi PostScript
HPIID522.WPD	HP LaserJet IID PostScript
HPIII522.WPD	HP LaserJet III PostScript
HPIIP522.WPD	HP LaserJet IIP PostScript
HP_3D522.WPD	HP LaserJet IIID PostScript
HP_3P522.WPD	HP LaserJet IIIP PostScript
IBM17521.WPD	IBM 4019 (17 fonts)
IBM39521.WPD	IBM 4019 (39 fonts)
L100_425.WPD	Linotronic™ 100 v42.5
L200230&.WPD	Linotronic 200/230
L300_471.WPD	Linotronic 300 v47.1
L300_493.WPD	Linotronic 300 v49.3
L330_52&.WPD	Linotronic 330
L500_493.WPD	Linotronic 500 v49.3
L530_52&.WPD	Linotronic 530
L630_52&.WPD	Linotronic 630
MT_TI101.WPD	Microtek TrueLaser
N2090522.WPD	NEC Silentwriter2 90
N2290520.WPD	NEC Silentwriter2 290
N2990523.WPD	NEC Silentwriter2 990
N890X505.WPD	NEC Silentwriter LC890XL
N890_470.WPD	NEC Silentwriter LC890

PostScript description	Related printer	<i>(continued)</i>
NCM40519.WPD	NEC Colormate PS/40	
NCM80519.WPD	NEC Colormate PS/80	
O5241503.WPD	OceColor G5241 PS	
O5242503.WPD	OceColor G5242 PS	
OL840518.WPD	Oki OL840/PS	
OLIVETI1.WPD	Olivetti PG 306 PS (13 fonts)	
OLIVETI2.WPD	Olivetti PG 306 PS (35 fonts)	
P4455514.WPD	Panasonic KX-P4455	
PHIIPX.WPD	Phaser II PX	
Q2200510.WPD	QMS-PS 2200	
Q820_517.WPD	QMS-PS 820	
SEIKO_04.WPD	Seiko ColorPoint PS Model 04	
SEIKO_14.WPD	Seiko ColorPoint PS Model 14	
TIM17521.WPD	TI MicroLaser PS17	
TIM35521.WPD	TI MicroLaser PS35	
TKPHZR21.WPD	Phaser II PX I	
TKPHZR31.WPD	Phaser III PX I	
TRIUMPH1.WPD	Triumph® Adler SDR 7706 PS (13 fonts)	
TRIUMPH2.WPD	Triumph Adler SDR 7706 PS	
U9415470.WPD	Unisys® AP9415	

Network Driver Files

Network drivers provide a network interface to the Windows for Workgroups File Manager, Control Panel, Print Manager, and system utilities.

Driver	Support file	Supported network
	WFWNET.DRV	WFWNET.HLP NETAPI.DLL PMSPL.DLL
		LMScript.EXE LMScript.PIF
	MSNET.DRV	Generic network driver*
	NETWARE.DRV	NETWARE.HLP NWPOPUP.EXE NETX.COM ROUTE.COM
		Novell NetWare 2.10 or above; Novell NetWare386 Supports pop-up messages Workstation shell Token Ring IPX source routing support
		MSIPX.COM MSIPX.SYS
		NDIS-compliant IPX protocol NDIS shim for MSIPX.COM to use NDIS drivers

* MSNET.DRV supports 3Com 3+Share®, 3Com 3+Open LAN Manager (XMS only), Banyan® VINES® 4.0, Microsoft LAN Manager 1.x (and compatibles), Microsoft LAN Manager 2.0 Basic (and compatibles), Microsoft Network (and compatibles), and IBM PC LAN Program.

For a list of supporting virtual device files for network drivers, see “Files for 386 Enhanced Mode” later in this chapter.

Multimedia Driver Files

The following drivers support the multimedia capabilities of Windows for Workgroups.

Filename	Purpose
MCICDA.DRV	MCI CD-audio driver
MCISEQ.DRV	MCI driver for MIDI driver
MCIWAVE.DRV	MCI driver for waveform audio
MIDIMAP.DRV	Driver for MIDI Mapper Control Panel extension
MMSOUND.DRV	Multimedia sound driver
MPU401.DRV	MIDI driver for MPU401 compatibles
MSADLIB.DRV	MIDI driver for Adlib compatibles
SNDBLST.DRV	SoundBlaster™ 1.5 DSP driver
SNDBLST2.DRV	SoundBlaster 2.0 DSP driver
TIMER.DRV	Multimedia timer driver

Font Files

Windows for Workgroups has several fonts for use with Windows for Workgroups, Windows- and MS-DOS-based applications, and data copied to the Clipboard from those applications. For details about Windows fonts, see Chapter 9, "Fonts," in the *Windows Resource Kit*.

Font files usually have a .TTF, .FON, or .FOT filename extension.

System Font Files

Three basic types of fonts are installed to support display and output devices:

- **System** is a proportional font used by default to draw menus, dialog box controls, and other text in Windows 3.x.
- **Fixed** is a fixed-width font used in Windows 2.x and earlier versions as the system font (for menus and dialog boxes).
- **OEM font**, or Terminal, is a fixed-width font used to display the OEM text in the Windows for Workgroups ClipBook Viewer. The OEM font also provides an OEM character set used by some Windows-based applications.

The system, fixed, and OEM fonts that are shipped with Windows for Workgroups are listed in the following tables.

System font file	Supported display resolution
8514SYS.FON	8514/a (1024x768) resolution system font
EGASYS.FON	EGA (640x350) resolution system font
EGASYS.FON	AT&T (640x400) resolution system font
VGASYS.FON	VGA (640x480) resolution system font

Fixed font file	Supported display resolution
8514FIX.FON	8514/a (1024x768) resolution fixed system font
EGAFIX.FON	EGA (640x350) resolution fixed system font
EGAFIX.FON	AT&T (640x400) resolution fixed system font
VGAFIX.FON	VGA (640x480) resolution fixed system font

OEM font file	Supported display resolution
8514OEM.FON	8514/a (1024x768) resolution Terminal font (U.S./Europe)
EGAOEM.FON	EGA (640x350) resolution Terminal font (U.S./Europe)
EGAOEM.FON	AT&T (640x400) resolution Terminal font (U.S./Europe)
VGAOEM.FON	VGA (640x480) resolution Terminal font (U.S./Europe)

Raster Font Files

Six resolutions of raster screen fonts are shipped with Windows for Workgroups. If used for printing, raster fonts print text and graphics as bitmaps or raster lines. The resolutions are identified by a letter appended to the filename of the font as described in the following table.

Letter	Output device	Resolution	x size*	y size*
A**	CGA display	2:1	96	48
B	EGA display	1.33:1	96	72
C**	Printer	1:1.2	60	72
D**	Printer	1.66:1	120	72
E	VGA display	1:1	96	96
F	8514 display	1:1	120	120

* x,y indicates the height/width aspect ratio, in pixels per inch.

** These fonts are not included on the Windows for Workgroups installation disks.

By appending the letter that identifies the resolution to the raster font filenames in the following table, you can see the files that Windows for Workgroups installs for a given display or printer. For example, the files for the 8514 raster fonts are COURF.FON, SSERIFF.FON, SERIFF.FON, SMALLF.FON, and SYMBOLF.FON.

Font	Filename	Character set	Font description
Courier	COURx.FON	ANSI	Fixed-width with serifs
MS Sans Serif	SSERIFx.FON	ANSI	Proportional-width sans serif
MS Serif	SERIFx.FON	ANSI	Proportional-width serif
Small	SMALLx.FON	ANSI	Proportional small size
Symbol	SYMBOLx.FON	Symbol	Math symbols

Vector Font Files

Windows for Workgroups provides these vector font files: ROMAN.FON, SCRIPT.FON, and MODERN.FON. For vector fonts, characters are stored as sets of relative coordinate pair points with connecting lines. Vector fonts are fully scalable fonts. The font can be created in any size desired, although applications or printing devices might have limits on the font sizes they support.

TrueType Font Files

The TrueType downloadable fonts shipped with Windows for Workgroups support the Arial, Courier, Symbol, and Times New Roman font families. Each family requires two files, a .TTF file and an .FOT file.

TrueType filenames	Font name
ARIAL.FOT, ARIAL.TTF	Arial
ARIALBD.FOT, ARIALBD.TTF	Arial Bold
ARIALBI.FOT, ARIALBI.TTF	Arial Bold Italic
ARIALI.FOT, ARIALI.TTF	Arial Italic
COUR.FOT, COUR.TTF	Courier
COURBD.FOT, COURBD.TTF	Courier Bold
COURBI.FOT, COURBI.TTF	Courier Bold Italic
COURI.FOT, COURI.TTF	Courier Italic
TIMES.FOT, TIMES.TTF	Times New Roman
TIMESBD.FOT, TIMESBD.TTF	Times New Roman Bold
TIMESBI.FOT, TIMESBI.TTF	Times New Roman Bold Italic
TIMESI.FOT, TIMESI.TTF	Times New Roman Italic
SYMBOL.FOT, SYMBOL.TTF	Symbol
WINGDING.FOT, WINGDING.TTF	Wingding

Font Files for MS-DOS-based Applications

Some fonts are installed for displaying MS-DOS-based applications in a window when Windows for Workgroups is running in 386 enhanced mode. By default, code page 437 (U.S.) fonts are installed. Other font files are included for international language support and are identified by the code page number appended to the filename.

The following font files are provided with the associated code page translation table files.

Font file	Translation table	Code page	Configuration
APP850.FON		850	U.S., 386 enhanced mode
DOSAPP.FON		437	U.S., 386 enhanced mode
CGA40850.FON	XLAT850.BIN	850	Multilingual
CGA40WOA.FON	-	437	U.S.
CGA80850.FON	XLAT850.BIN	850	Multilingual
CGA80WOA.FON	-	437	U.S.
EGA40850.FON	XLAT850.BIN	850	Multilingual
EGA40WOA.FON	-	437	U.S.
EGA80850.FON	XLAT850.BIN	850	Multilingual
EGA80WOA.FON	-	437	U.S.
HERC850.FON	XLAT850.BIN	850	Multilingual
HERCWOA.FON	-	437	U.S.
VGA850.FON	XLAT850.BIN	850	Multilingual
VGA860.FON	XLAT860.BIN	860	Portuguese
VGA861.FON	XLAT861.BIN	861	Icelandic
VGA863.FON	XLAT863.BIN	863	French Canadian
VGA865.FON	XLAT865.BIN	865	Norwegian/Danish

International Support Files

Windows for Workgroups provides language libraries to support a number of languages.

Filename	Supported languages
LANGDUT.DLL	Dutch language driver
LANGENG.DLL	General International language driver
LANGFRN.DLL	French language driver
LANGGER.DLL	German language driver
LANGSCA.DLL	Finnish/Icelandic/Norwegian/Swedish language driver
LANGSPA.DLL	Spanish language driver

MS-DOS Support Components of Windows for Workgroups

Two kinds of files provide MS-DOS support for Windows for Workgroups: MS-DOS drivers and grabber files that support data exchange between Windows and MS-DOS-based applications.

MS-DOS Driver Files

Several MS-DOS driver files are included with Windows for Workgroups. The drivers provided with Windows for Workgroups are the recommended versions to use.

Driver	Purpose
EGA.SYS	EGA MS-DOS driver
EMM386.EXE	Microsoft MS-DOS 386 EMS manager
HIMEM.SYS	Microsoft MS-DOS XMS manager
RAMDRIVE.SYS	Microsoft MS-DOS RAMDrive utility
SMARTDRV.EXE	Microsoft MS-DOS SMARTDrive 4.0 disk-caching utility
LMOUSE.COM	MS-DOS Level Logitech mouse driver
MOUSE.COM	MS-DOS mouse driver
MOUSE.SYS	MS-DOS mouse driver (installed at MS-DOS boot time)
MOUSEHP.COM	MS-DOS mouse driver for Hewlett-Packard systems
MOUSEHP.SYS	MS-DOS mouse driver for Hewlett-Packard systems
MSCDEX.EXE	Microsoft CD-ROM Extensions 2.21 driver
*.DOS	MS-DOS NDIS network interface card driver(s) - (see below)
NET.EXE	WFW MS-DOS network redirector
NET.MSG	WFW network redirector message file
NETH.MSG	WFW network redirector help message file
PROTMAN.EXE	WFW protocol manager TSR
WORKGRP.SYS	Real-mode stub for Windows for Workgroups network redirector



NDIS Network Adapter Card Driver Files

The following NDIS drivers are provided with Windows for Workgroups to support network adapter cards.

Driver	Related Network Adapter Card
AM2100.DOS	Advanced Micro Devices AM2100/PCnet
DEPCA.DOS	DEC EtherWorks
E20NDIS.DOS	Cabletron E2010-X
E21NDIS.DOS	Cabletron E2112
ELNK16.DOS	3Com EtherLink® 16
ELNK3.DOS	3Com EtherLink III
ELNKII.DOS	3Com EtherLink II
ELNKMC.DOS	3Com EtherLink/MC
ELNKPL.DOS	3Com EtherLink Plus
EVX16.DOS	Everex™ SpeedLink /PC16 (EV2027)
EXP16.DOS	Intel EtherExpress 16
HPLANB.DOS	HP LAN
I82593.DOS	Intel Motherboard Module
IBMTOK.DOS	IBM Token Ring
MAC586.SYS	DCA® 10 Mb



Driver	Related Network Adapter Card (continued)
NDIS39XR.DOS	Proteon Token Ring
NE1000.DOS	Novell/Anthem NE1000 (or compatible)
NE2000.DOS	Novell/Anthem NE2000 (or compatible)
NI6510.DOS	Racal-Interlan NI6510
OLITOK.DOS	Intel TokenExpress™ 16/4
PE2NDIS.DOS	Xircom Pocket Ethernet II
PRO4.DOS	Proteon ISA Token Ring
SMC3000.DOS	SMC 3000 Series
SMCMAC.DOS	SMC (WD) EtherCard PLUS
SMC_ARC.DOS	SMC ARCNET
STRN.DOS	NCR Token Ring
TCCARC.DOS	Thomas Conrad TC6x4x (Enhanced Mode)
TLNK.DOS	3Com TokenLink

Note Other NDIS drivers are available as part of the Windows Driver Library (WDL). See Appendix A for more information about where to obtain the WDL.

WinOldAp and the Grabber Files

Two primary parts of Windows support MS-DOS-based applications when Windows for Workgroups is running in standard mode: WinOldAp and the grabber. When Windows for Workgroups runs in 386 enhanced mode, the limited resources on the computer are virtualized to provide virtual memory, virtual displays, virtual communications, and a number of other services. The related files are discussed in “Files for 386 Enhanced Mode” later in this chapter.

WinOldAp and grabber files support data exchange between MS-DOS-based applications and Windows. Support for MS-DOS-based applications varies, depending on the capabilities of the system CPU and the mode in which Windows for Workgroups is running.

WinOldAp comes in two versions for the two Windows for Workgroups operating modes:

- WINOLDAP.MOD for standard mode
- WINOA386.MOD for 386 enhanced mode

The grabber for your system is specific to the display driver.

The 286 grabbers used for standard mode only support PrintScreen and copying and pasting text between Windows-based applications and MS-DOS-based applications. 386 grabbers that support Windows 386 enhanced mode provide the following capabilities:

- Copying text from MS-DOS-based applications
- Displaying data in a windowed virtual machine
- Selecting data in a windowed virtual machine
- Copying graphics to the Windows Clipboard
- PrintScreen

Files that provide font support for the grabbers are listed below, with descriptions of the kinds of display drivers that the grabbers support.

**286 grabber
support file**

Display device supported

CGA.2GR	CGA
EGACOLOR.2GR	EGA
EGAMONO.2GR	EGA monochrome
HERCULES.2GR	Hercules monochrome
OLIGRAB.2GR	Olivetti/AT&T monochrome or PVC
VGACOLOR.2GR	VGA
VGAMONO.2GR	VGA monochrome

**386 grabber
support file**

Display device supported

EGA.3GR	EGA 386
HERC.3GR	Hercules monochrome
PLASMA.3GR	Compaq Portable plasma
V7VGA.3GR	Video 7
VGA.3GR	VGA
VGA30.3GR	VGA (version 3.0)
VGADIB.3GR	DIB (8514/a monochrome)

Files for Standard Mode

When Windows for Workgroups is running in standard mode, the processor is switched to 80286 protected mode, allowing access to extended memory through XMS support. The DOSX.EXE file, required for standard mode, is the MS-DOS Extender for Windows. When Windows for Workgroups runs in standard mode, WIN.COM executes DOSX.EXE. The Kernel file is then loaded (KRNL286.EXE for 80286 machines, or KRNL386.EXE for 80386 machines), which in turn loads the other parts of Windows for Workgroups. Two more files support task swapping for standard mode:

- WSWAP.EXE supports Windows-based applications in standard mode.
- DSWAP.EXE supports MS-DOS-based applications in standard mode.

Files for 386 Enhanced Mode

In 386 enhanced mode, Windows for Workgroups can use virtual memory. Much of the virtual support is provided by WIN386.EXE, which is executed by WIN.COM. When WIN386.EXE begins to load, it searches for the files identified in the [386enh] section of SYSTEM.INI. Some standard files are built into WIN386.EXE (designated with the "*" symbol preceding the name of the driver in SYSTEM.INI entries). The other files WIN386.EXE loads to support virtual devices are listed in the following table.

Filename	Virtual device supported
LANMAN10.386	LAN Manager version 1.0 support
LVMD.386	Logitech virtual mouse device
HPEBIOS.386	EBIOS virtual device for Hewlett-Packard machines
MONOUMB2.386	UMB driver for Monochrome video space
MSCVMD.386	Mouse Systems virtual mouse device
V7VDD.386	Video Seven™ virtual display device
VADLIBD.386	Virtual DMA device for Adlib
VCD.386	Virtual communications device
VDD8514.386	8514/a virtual display device
VDDCGA.DRV	CGA virtual display device
VDDEGA.386	EGA virtual display device
VDDHERC.386	Hercules monochrome virtual display device
VDDTIGA.386	TIGA virtual display device
VDDTLI4.386	Tseng ET4000 Super VGA virtual display device
VDDVGA30.386	VGA virtual display device (version 3.0)
VDDXGA.386	XGA® virtual display device
VIPX.386	Novell NetWare virtual IPX support
VDMAD.386	Virtual DMA device
VNETWARE.386	NetWare virtual support
VPD.386	Virtual printer device

Filename	Virtual device supported	(continued)
VPCD.386	Virtual programmable interrupt controller device	
VPOWERD.386	Advanced Power Management virtual device	
VSBD.386	SoundBlaster virtual device	
VTDAPI.386	Multimedia virtual timer device	
WIN386.PS2	Support for PS/2 architecture	
VBROWSE.386	WFW virtual network browsing device	
VNB.386	WFW virtual NetBEUI device	
VNETBIOS.386	WFW virtual NetBIOS device	
VNETSUP.386	WFW virtual network support device	
VREDIR.386	WFW virtual network redirector device	
VSERVER.386	WFW virtual network server device	
VSHARE.386	WFW virtual file sharing device	
VWC.386	WFW virtual workgroup client device	



Windows for Workgroups Applications, Setup, and Other Files

Files for Windows for Workgroups Applications

Windows for Workgroups files also include applications, shells, utilities, accessories, and games. The following table lists the applications and associated files, with a brief description of each application.

Filename	Associated files	Application name and description
CALC.EXE	CALC.HLP	Calculator (general/scientific)
CARDFILE.EXE	CARDFILE.HLP	Cardfile (desktop Rolodex)
CHARMAP.EXE	CHARMAP.HLP	Character Map
CLIPBRD.EXE	CLIPBRD.HLP CLIPSRV.EXE	ClipBook Viewer ClipBook DDE server application
CLOCK.EXE		Clock (analog/digital)
CONTROL.EXE	CONTROL.HLP CONTROL.INI CPWIN386.CPL DRIVERS.CPL LZEXPAND.DLL MAIN.CPL MIDIMAP.CFG SND.CPL	Control Panel Initialization file 386 enhanced mode extension for Control Panel Installable drivers extension for Control Panel File expansion utility for Control Panel Main Control Panel extension MIDI Mapper extension file for Control Panel Sound extension for Control Panel
	WFWSETUP.DLL	WFW network setup extension for Control Panel
DRWATSON.EXE		Windows fault detection utility
MPLAYER.EXE	MPLAYER.HLP MMSYSTEM.DLL MMTASK.TSK	Media Player Multimedia system library Multimedia background task



<i>Filename</i>	<i>Associated files</i>	<i>Application name and description</i>	<i>(continued)</i>
 MSD.EXE	MSD.INI	Microsoft Diagnostics utility and initialization file	
 MSHEARTS.EXE	MSHEARTS.HLP CARDS.DLL	Hearts game	
 MSMAIL.EXE	MSMAIL.HLP SENDFILE.DLL AB.DLL DEMILAYR.DLL FRAMEWRK.DLL IMPEXP.DLL MAILMGR.DLL MAILSPL.EXE MAPI.DLL MSSFS.DLL STORE.DLL VFORMS.DLL WGPOMGR.DLL	Mail Application File Manager extension to send file as attachment Address Book user interface support functions MS WGA System Services layer Microsoft WGA Application Framework layer Mail message file import utility Mail Manager API support functions MS Mail for Windows—Mail spooler MS Messaging Applications Programming Interface Microsoft shared file system transport Message store support functions Mail viewed forms DLL Windows for Workgroups Post Office Manager functions	
 NETDDE.EXE	NDDEAPI.DLL NDDENB.DLL	Network DDE background application Network DDE—DDE shares API support Network DDE driver for NetBIOS	
 NETWATCH.EXE	NETWATCH.HLP	Net Watcher	
NOTEPAD.EXE	NOTEPAD.HLP	Notepad (desktop text editor)	
PACKAGER.EXE	PACKAGER.HLP	Object Packager	
PBRUSH.EXE	PBRUSH.DLL PBRUSH.HLP	Paintbrush	
PIFEDIT.EXE	PIFEDIT.HLP	PIF Editor	
POWER.HLP	SL.DLL, SL.HLP	Advanced Power Management supporting files	
PRINTMAN.EXE	PRINTMAN.HLP	Print Manager (Windows print spooler)	
PROGMAN.EXE	PROGMAN.INI PROGMAN.HLP	Program Manager (shell)	
RECORDER.EXE	RECORDER.HLP RECORDER.DLL	Recorder (desktop macro recorder)	
REGEDIT.EXE	REGEDIT.HLP REGEDITV.HLP DDEML.DLL OLECLI.DLL OLESVR.DLL	Registration Editor and supporting files DDE management library Client library and server for object linking and embedding	
 SCHDPLUS.EXE	SCHDPLUS.HLP SCHDPLUS.INI MSREMIND.EXE SCHEDMSG.DLL TRNSCHED.DLL	Schedule+ Application Schedule+ background reminder notification Schedule+ message forms for MS Mail Schedule+ shared file system transport	
SHELL.DLL		Shell library	
SOL.EXE	SOL.HLP	Solitaire game	
SMARTDRV.EXE		Disk-caching utility	
SOUNDREC.EXE	SOUNDREC.HLP	Sound Recorder	

Filename	Associated files	Application name and description	(continued)
SYSEDIT.EXE		Windows System Editor	
TASKMAN.EXE		Task Manager (application switcher)	
TERMINAL.EXE	TERMINAL.HLP	Terminal (desktop communications)	
TOOLHELP.DLL		Windows Tool Helper library	
WINCHAT.EXE	WINCHAT.HLP	Chat	
WINFILE.EXE	WINFILE.HLP	File Manager	
WINHELP.EXE	WINHELP.HLP GLOSSARY.HLP	Help (Windows help engine) Windows Help glossary	
WINMETER.EXE		System performance meter	
WINMINE.EXE	WINMINE.HLP	MineSweeper game	
WINTUTOR.EXE	WINTUTOR.DAT	Windows Tutorial	
WRITE.EXE	WRITE.HLP	Write (desktop word processor)	

Control Panel uses LZEXPAND.DLL to expand files from the Windows for Workgroups installation disks. Because most of the files on the Windows for Workgroups installation disks are compressed (except SETUP.INF, SETUP.EXE, and EXPAND.EXE), Control Panel must expand the files to install a new printer or to add fonts. LZEXPAND is a Windows library counterpart to EXPAND.EXE.

Setup-Related Files

Setup has a number of files for its exclusive use. For example, the *.LGO files contain the code for displaying the opening screen logo, and the *.RLE files contain the actual logo bitmap (in Run Length Encoded format). Setup combines the .LGO and .RLE files with the WIN.CNF file to create WIN.COM. Setup also uses the files listed in the following table.

Filename	Purpose
SETUP.EXE	Windows for Workgroups Setup application file
SETUP.HLP	Setup Help
SETUP.INF	Setup information file
SETUP.INI	Setup initialization file
SETUP.REG	Registration Database template
SETUP.SHH	Automated Setup template
SETUP.TXT	Windows for Workgroups Readme file
VER.DLL	Version Resource and File Installation library
WINVER	Windows-version utility
XMSMMGR.EXE	Setup XMS Manager
EXPAND.EXE	MS-DOS-based file expansion utility

Filename	Purpose	<i>(continued)</i>
Startup logo files:		
CGALOGO.LGO	CGA startup logo code	
CGALOGO.RLE	CGA display logo screen	
EGALOGO.LGO	EGA startup logo code	
EGALOGO.RLE	EGA logo screen	
EGAMONO.LGO	EGA monochrome startup logo code	
EGAMONO.RLE	EGA monochrome logo screen	
HERCLOGO.LOG	Hercules monochrome startup logo code	
HERCLOGO.RLE	Hercules logo screen	
VGALOGO.LGO	VGA startup logo code	
VGALOGO.RLE	VGA logo screen	
Initialization and information source files:		
APPS.INF	Information file for MS-DOS-based applications	
CONTROL.INF	Information file for Control Panel and printer installation	
CONTROL.SRC	CONTROL.INI template	
NETWORK.INF	Information file for network installation	
PRTUPD.INF	Information file for printer driver updates	
SYSTEM.SRC	SYSTEM.INI template	
WIN.CNF	Windows for Workgroups startup code	
WIN.SRC	WIN.INI template	



Other Files

The following files serve a wide range of functions, including support for PS/2 architectures and README files for general information.

Filename	Purpose
MORICONS.DLL	Icons for MS-DOS-based applications
Bitmaps files for wallpaper:	
256COLOR.BMP	256-color wallpaper
ARCADE.BMP	Arcade wallpaper
ARGYLE.BMP	Argyle wallpaper
CARS.BMP	Cars wallpaper
CASTLE.BMP	Castle wallpaper
EGYPT.BMP	Egypt wallpaper
HONEY.BMP	Honey wallpaper
LEAVES.BMP	Leaves wallpaper
REDBRICK.BMP	Red brick wallpaper
RIVETS.BMP	Rivets wallpaper
SQUARES.BMP	Squares wallpaper
THATCH.BMP	Thatch wallpaper
WINLOGO.BMP	Logo wallpaper
ZIGZAG.BMP	Zigzag wallpaper

Filename	Purpose	(continued)
-----------------	----------------	--------------------

Screensaver files:

SSFLYWIN.SCR	Flying Windows
SSMARQUE.SCR	Marquee screen saver
SSSTARS.SCR	Stars screen saver

MIDI sound file:

CANYON.MID	Canyon MIDI sound
------------	-------------------

Wave-form sound files:

CHIMES.WAV	Exit sound
DING.WAV	Default beep

RINGIN.WAV	Chat incoming ring sound
RINGOUT.WAV	Chat outgoing ring sound

README files:

NETWORKS.WRI	README file for networks
PRINTERS.WRI	README file for printers
README.WRI	README file
SYSINI.WRI	README file for SYSTEM.INI
WININI.WRI	README file for WIN.INI

Miscellaneous hardware support and other supporting files:

386MAX.VXD	Qualitas® 386MAX virtual device for standard mode
BLUEMAX.VXD	Qualitas BlueMAX™ virtual device

COMMCTRL.DLL	WFW internal custom-control user interface functions
COMMDLG.DLL	Windows Common Dialogs library
WIN87EM.DLL	80x87 math coprocessor emulation library
WINDOWS.LOD	Qualitas 386MAX/BlueMAX loadable module



Files You Can Delete

Because of the large number of files that come with Windows for Workgroups, you might want to delete some of the files to free disk space.

Note Do not delete any of these files while Windows for Workgroups is running. Instead, quit Windows for Workgroups, and then delete the files from the MS-DOS command prompt.

You can safely delete the following files when Windows for Workgroups is not running without degrading Windows for Workgroups performance:

- Any files in the TEMP directory (temporary files have a .TMP filename extension).
- Any files that start with the characters ~WOA or ~GRB.
- Any files named WIN386.SWP (a temporary Windows swap file). DO NOT delete files named 386SPART.PAR or SPART.PAR -- these files are used for a permanent swap file and should not be removed manually. (Use Control Panel to change the size or remove the permanent swapfile.)

From Windows Setup, choose Add/Remove Windows Components from the Options menu to remove any of these files from your system:

- Any accessories you do not use (such as Paintbrush, Write, and Cardfile) with their related .HLP and .DLL files (if any)
- Games
- Screen savers
- Wallpapers (.BMP files) and sound files (.WAV files)

Chapter
5

Windows for Workgroups Setup Information Files

This chapter contains information that can be used to maintain the Microsoft Windows for Workgroups 3.1 setup information files (SETUP.INF, APPS.INF, CONTROL.INF, NETWORK.INF, and OEMSETUP.INF) for a network installation. This chapter can also be used to create custom setup information for multiple installations.

Note You should only modify the setup information files to specify custom drivers, applications, and machine types for multiple systems. See “Modifying .INF Files for Custom Installations” later in this chapter for details.

Related Information

- *Windows for Workgroups Resource Kit: Chapter 4, “Windows for Workgroups Files;” Chapter 6, “Windows for Workgroups Initialization Files;” “Troubleshooting Setup,” in Chapter 14, “Troubleshooting Windows for Workgroups.”*

Contents of This Chapter

About the Setup Information Files	5-3
Format of the .INF Files	5-4
Changing Entries in .INF Files	5-5
SETUP.INF: Setup Information File	5-6
General Installation Sections	5-8
Display Driver Sections	5-12
Keyboard and Code Page Sections	5-15
Mouse Driver Sections	5-18
Network Installation Sections	5-19
System Fonts Sections	5-23
Copy-Files Sections	5-24
Program Manager Groups Sections	5-28
Fonts Sections	5-32
Incompatible Driver Sections	5-33
Miscellaneous Sections	5-34
System Configuration Sections	5-38
APPS.INF: Application Information File	5-42

CONTROL.INF: The Windows Control Panel Information File.....	5-48
NETWORK.INF: Network Information File.....	5-50
General Installation Sections	5-51
Network Adapter Driver Sections	5-52
Network Protocol Driver Sections	5-55
Secondary Network Driver Sections.....	5-57
Installation Sections Information	5-59
Protocol Sections Information.....	5-62
OEMSETUP.INF: OEM Network Setup Information File.....	5-64
Example OEMSETUP.INF	5-68
Modifying .INF Files for Custom Installations.....	5-69
Modifying .INF Files to Install Custom Windows	5-69
Modifying .INF Files to Install Custom Applications	5-71
Adding Custom PIF Information in APPS.INF	5-73

About the Setup Information Files

The following setup information (.INF) files are used to supply information for the Windows for Workgroups Setup program: SETUP.INF, APPS.INF, CONTROL.INF, NETWORK.INF, and OEMSETUP.INF. The entries for each .INF file are discussed in detail in this chapter.

SETUP.INF

SETUP.INF is the key file that Setup uses to install the Windows for Workgroups system and application files and to create program groups that you can use to run applications from Program Manager. This file is on Windows for Workgroups Installation Disk #1 and is not compressed.

Setup checks information in SETUP.INF for both MS-DOS mode setup and Windows mode setup. SETUP.INF entries specify which files are copied based on the system hardware configuration and the choices specified during installation. Setup places SETUP.INF in the SYSTEM subdirectory during installation. You can modify SETUP.INF entries to accommodate specific hardware and software configurations that are required to install Windows for Workgroups on a number of computers.

APPS.INF

APPS.INF contains information for installing applications on Windows for Workgroups. Setup places APPS.INF in the SYSTEM subdirectory during Windows for Workgroups installation. You can modify this file to install PIF information for custom applications during installation. Program information (.PIF) files enable MS-DOS-based applications to be run in Windows for Workgroups. See Chapter 8, "PIFs and PIF Editor," in the *Windows Resource Kit* for more information about program information files.

CONTROL.INF

CONTROL.INF contains information about specific printers and provides settings for international support. You can modify this file to install custom printer drivers. Setup places CONTROL.INF in the SYSTEM subdirectory during installation.



NETWORK.INF

NETWORK.INF contains specific information used to install network adapter cards and network transports that are supported in Windows for Workgroups 3.1. Setup places NETWORK.INF in the SYSTEM subdirectory during installation.



OEMSETUP.INF

OEMSETUP.INF contains setup information for original equipment manufacturer (OEM) network adapter cards and network transports that are not provided with Windows for Workgroups 3.1. Setup places any OEMSETUP.INF files in the SYSTEM subdirectory during installation. (Setup may rename OEMSETUP.INF if an OEMSETUP.INF file already exists in the SYSTEM subdirectory.)

For other ways to customize setup to meet the installation needs for your workgroup, see the custom installation decision tree in Chapter 3, “Windows for Workgroup Installation.”

Format of the .INF Files

Windows for Workgroups setup information (.INF) files are text files that are separated into logical categories that contain one or more sections. The format for each section is as follows:

```
[section]
keyname= value           ; comment
```

Value	Definition
[section]	The name (header) of a section. The enclosing brackets ([]) are required, and the left bracket must be in the leftmost column in the file.
keyname	The name of an entry, which can consist of any combination of letters and digits. For many entries, the <i>keyname</i> must be followed immediately by an equal sign (=).
value	The value of each entry, which can be an integer, a string, or a quoted string depending on the entry.
; comment	Some entries include brief comments below the header or on the same line as an entry. You can include comments anywhere in an .INF file by prefacing each line of the comment with a semicolon (;).

Note The order of the sections in an .INF file is not important, except for the placement of the **[blowaway]** section in SETUP.INF. Also, entries do not necessarily appear in the .INF files in the order they are listed in this chapter. To change an entry, you must search for the entry within the related section.

In this chapter, the sections that appear in bold contain the actual words that are used in an .INF file—for example, **[shell]**. When a section is referenced generically, the section name is shown in *italic*, in bold brackets. For example, NETWORK.INF contains the *[netcard_protocol]* section that actually includes separate sections for each network adapter card name, such as **[MSSEE16]** for the Intel EtherExpress 16 or 16TP adapter card.

The actual words of an entry are shown in bold, such as **startup=WIN.COM**. Some variable entries must be substituted for specific values in the entry. These entries are shown in *italic* in the following format:

keyname= profile, description, filesize

Uppercase or lowercase letters do not matter for values, unless specified for a particular item. Some items must be enclosed in quotation marks (" ")—for example, **caption="Windows Setup"**.

Changing Entries in .INF Files

Entries in .INF files can be changed to modify the behavior and appearance of Windows for Workgroups, if needed. Refer to “Modifying .INF Files for Custom Installations” later in this chapter for more information.

To change entries in an .INF file:

Important Always back up the .INF file before you open it, and use care when making changes with a text editor. Incorrect changes can lead to unexpected results when you run Setup. Also, some text editors can damage characters with ANSI values of greater than 127. Use a non-formatting text editor to change these files, or save the files in an ASCII (text-only) format.

1. Open the copy of the .INF file with a text editor.
2. Edit the specific entries and save the file in ASCII (text only) format.
3. Restart Windows for Workgroups.

SETUP.INF: Setup Information File

SETUP.INF is the key file that Setup uses to install the Windows for Workgroups system and application files and to create program groups that you can use to run applications from Program Manager. SETUP.INF is organized into the following types of sections:

- General installation
- Display driver
- Keyboard and code page
- Mouse driver
- Network installation
- System fonts
- Copy-files
- Program Manager
- Groups
- Fonts
- Incompatible driver
- Miscellaneous
- System configuration

The sections that are contained within SETUP.INF are described in the following table.

<i>Type of section</i>	<i>Section</i>	<i>Purpose</i>
General installation	[setup]	Defines the Windows setup Help file.
	[run]	Defines which application programs run at the end of setup.
	[dialog]	Defines the text for setup dialog boxes.
	[data]	Defines default settings for setup.
	[winexec]	Defines data to copy the correct kernel file to Windows mode setup.
	[disks]	Defines disks for setup installation.
	[user]	Defines the TEMP file containing user information.
	[windows]	Defines the files for the WINDOWS directory.
	[windows.system]	Defines the files for the SYSTEM subdirectory.
	[windows.system.386]	Defines the files for the SYSTEM subdirectory for an 80386.
	[386max] and [bluemax]	Defines subentries for 80386 expanded memory managers.
	[shell]	Defines the shell entry for setup, usually Program Manager.
Display driver	[display]	Defines a set of display drivers.
	[.3gr]	Defines the font files to copy for the 386 grabber.
Keyboard and code page	[keyboard.drivers]	Maps keyboard driver filenames to keyboard short names.
	[keyboard.types]	Defines the keyboard short names used in the [machine] section.
	[keyboard.tables]	Maps keyboard names for international support.
	[codepages]	Maps code pages for international support.
Mouse driver	[pointing.device]	Defines mouse device and driver information.
	[lmouse]	Defines data for the Logitech MS-DOS mouse driver.
	[dos.mouse.drivers]	Maps Windows for Workgroups mouse driver-names to MS-DOS drivers.

Type of section	Section	Purpose	(continued)
Network installation	[network] [network_versions] [network_specific]	Defines names for network files and .INI file changes for networking. Defines information for specific versions of network drivers. Defines entries for .INI files for specific networks.	
System fonts	[sysfonts] [fixedfonts] [oemfonts]	Defines system font files by display resolution. Defines fixed-pitch font files by display resolution. Defines OEM system font files by display resolution.	
Copy-files	[win.copy] [win.copy.net] [win.copy.net.386] [win.copy.386] [DelFiles] [RenFiles] [Win_CopyFiles]	Defines files to copy for 80286 personal computers. Defines files to copy for network workstations. Defines files to copy for 80386 and higher personal computers for network workstations. Defines files to copy for 80386 and higher personal computers. Defines files to delete when upgrading to Windows for Workgroups 3.1. Defines files to rename when upgrading to Windows for Workgroups 3.1. Defines Windows for Workgroups files that are to be copied.	
Program Manager	[new.groups]	Defines Program Manager groups to modify for the Windows for Workgroups upgrade	
Groups	[progman.groups] [group#]	Defines Program Manager groups that are created for a new installation. Defines program items that are installed in each Program Manager group.	
Fonts	[fonts] [ttfonts]	Defines raster and vector screen font files. Defines the TrueType font files to copy.	
Incompatible driver	[compatibility] [incompTSR1] [incompTSR2] [block_devices]	Defines the drivers to remove in CONFIG.SYS. Defines TSRs and drivers that cause problems during setup. Defines TSRs and drivers that cause problems if running with Windows for Workgroups. Defines incompatible block devices.	
Miscellaneous	[installable.drivers] [translate] [update.files] [update.dependents] [ini.upd.patches] [blowaway] [ini.upd.31]	Defines data for multimedia drivers. Defines filename translation data for version 3.0 of OEM disks. Defines installable drivers to update to version 3.1. Defines dependents for installable drivers. Temporarily maps .INI file updates. Defines the end of installation and the start of the configuration sections. Defines .INI file updates for Windows for Workgroups 3.1.	
System configuration	[system] [machine] [cookz] [special_adapter] [ebios] [language]	Maps system short names and descriptors. Defines names for specific computers. Defines the cookz for specific computer setups. Defines special data for EtherLink MC cards. Defines extended BIOS names. Defines language DLL names.	

General Installation Sections

The following sections in SETUP.INF define the installation options and specify the filenames and locations of related files for MS-DOS mode setup:

[setup]	[winexec]	[windows.system]
[run]	[disks]	[windows.system.386]
[dialog]	[user]	[386max]
[data]	[windows]	[bluemax]
		[shell]

[setup]

This section contains the location of the Help file for the Setup program. For online Help to be available during installation, there must be an entry in the following format:

```
[setup]
help= setup.hlp
```

[run]

This section specifies which application programs are run at the end of setup. The listed applications run in the order of appearance. No entries are present in this section by default.

[dialog]

This section contains the text strings used in the setup dialog boxes. Some of the keynames and default text strings are the following:

```
[dialog]
caption= "Windows Setup"
exit= "Exit Windows Setup"
title= "Installing Windows for Workgroups 3.1"
options= "In addition to installing Windows for Workgroups 3.1, you can:"
printwait= "Please wait while Setup configures your printer(s)..."
copywait= "Welcome to Microsoft Windows for Workgroups 3.1! . . ."
(other entries)
```

[data]

This section defines a series of default settings for setup. The first group of entries specifies the hard disk space required to run a type of configuration option of Windows for Workgroups. To install additional files during setup, you must increase the minimum disk space required for installation. The second group of entries refers to the default files and other data that is used by setup.

The following are the standard entries in the **[data]** section:

```
[data]
; Disk space required
; <type of setup>= <Full install space>, <Min install space>

upd2x386full = 15000000,10500000; 15.0 Mb, 10.5 Mb  WFW
upd2x286full = 13800000,9300000 ; 13.8 Mb, 9.3 Mb  WFW
upd3x386full = 10000000,7000000 ; 10.0 Mb, 7.0 Mb  WFW
upd3x286full = 6600000,6200000 ; 6.6 Mb, 6.2 Mb  WFW
updVVV386full = 500000,500000 ; .5 Mb, .5 Mb  WFW
updVVV286full = 500000,500000 ; .5 Mb, .5 Mb  WFW

new386full = 15000000,10500000; 15.0 Mb, 10.5 Mb  WFW
new286full = 13800000,9300000 ; 13.8 Mb, 9.3 Mb  WFW

netadmin = 21000000 ; 21.0 Mb  WFW
netadminupd = 21000000 ; 21.0 Mb  WFW
upd2x386net = 1200000 ; 1.2 Mb  WFW
upd3x386net = 1200000 ; 1.2 Mb  WFW
upd2x286net = 1200000 ; 1.2 Mb  WFW
upd3x286net = 1200000 ; 1.2 Mb  WFW
new386net = 1200000,1200000 ; 1.2 Mb, 1.2 Mb  WFW
new286net = 1200000,1200000 ; 1.2 Mb, 1.2 Mb  WFW

; Defaults used in setting up and names of a few files
startup = WIN.COM
defdir = C:\WINDOWS
shortname = Windows
welcome = "Windows 3.1"
deflang = enu
defxlat = 437
defkeydll = usadll
register = "regedit /s /u setup.reg"
tutor = "wintutor.exe"
NetSetup = FALSE
MouseDrv = TRUE
Version = "3.1.032W"
WFWDrv = "wfwnet.drv"
partial = FALSE
```

Note If you want to force users to set up using the network installation option (**setup /n**), make sure **netsetup= TRUE** in **[data]**. For more information, see "Setting Up Windows for Workgroups for a Network Installation," in Chapter 3, "Windows for Workgroups Installation."

[winexec]

This section provides data needed by MS-DOS mode setup to copy the correct kernel file for Windows mode setup. In most cases, you don't need to change these entries.

```
[winexec]
execstd= "dosx.exe "           ; MS-DOS extender for standard mode
execcmd= " krnl286.exe /b /q;" ; command and switches for kernel
exechimem= "xmsmmgr.exe"      ; extended memory manager for Setup
himemcmd= ""                  ; switches for HIMEM.SYS
Krnl386= 2:krnl386.exe        ; kernel for 386 enhanced mode
Krnl286= 2:krnl286.exe        ; kernel for standard mode
dosx= 3:dosx.exe              ; disk location for DOSX.EXE file
```

[disks]

This section identifies the disks that can be used for installation. The following is the format for entries in this section:

n= *path*, *disk name*, *disk tag*

Value	Definition
<i>n</i>	The disk <i>number</i> , including 1–9 and A–Z (single character only). The disk number 0 is reserved to represent the Windows setup directory.
<i>path</i>	The path of disk <i>n</i> . The symbol “.” tells Setup to look for the disk in the same drive as the Windows for Workgroups Setup disks.
<i>disk name</i>	The label of disk <i>n</i> , enclosed in quotation marks (“ ”). There must be a disk name for each disk number used.
<i>disk tag</i>	The volume name for disk <i>n</i> (used to determine if the correct disk is in the drive).

This is an example of the **[disks]** section:

```
[disks]
1=., "Microsoft Windows for Workgroups 3.1 Disk #1", disk1
2=., "Microsoft Windows for Workgroups 3.1 Disk #2", disk2
3=., "Microsoft Windows for Workgroups 3.1 Disk #3", disk3
4=., "Microsoft Windows for Workgroups 3.1 Disk #4", disk4
5=., "Microsoft Windows for Workgroups 3.1 Disk #5", disk5
6=., "Microsoft Windows for Workgroups 3.1 Disk #6", disk6
7=., "Microsoft Windows for Workgroups 3.1 Disk #7", disk7
8=., "Microsoft Windows for Workgroups 3.1 Disk #8", disk8
9=., "Microsoft Windows for Workgroups 3.1 Disk #9", disk9
A=., "Microsoft Windows for Workgroups 3.1 Disk #10", diska
```

To install other applications from disk during the Windows for Workgroups installation, you can add an entry in the **[disks]** section. For details, see “Modifying .INF Files for Custom Installations” later in this chapter.

Note For SETUP.INF entries, a number followed by a colon, such as **1:**, identifies the Windows for Workgroups installation disk that contains the file based on the entries in the **[disks]** section. If installed from a network server where the SETUP.INF files all exist in a single directory, Setup ignores the disk numbers.

[user]

This is a temporary file that identifies which installation disk contains the user and company identification—for example:

```
[user]
4:setup.ini, nouupdate
```

[windows]

This section identifies the files MS-DOS mode setup copies to the WINDOWS directory. When the **win.src** and **system.src** entries are assigned a value of **Net**, Setup copies these files only for the Windows administrative setup option (**setup /a**). This option prepares a shared copy of Windows for Workgroups, which permits setup to run over a network. The following is an example of the **[windows]** section:

```
[windows]
2:setup.hlp
1:setup.txt
3:win.src,      Net
3:system.src,  Net
2:winhelp.exe
2:control.hlp
```

[windows.system]

This section defines the files MS-DOS mode setup copies to the Windows for Workgroups SYSTEM subdirectory. For details about the files that are copied during Windows mode setup, refer to “Copy-Files Sections” later in this chapter. This is an example of the **[windows.system]** section:

```
[windows.system]
1:gdi.exe
1:user.exe
3:win.cnf
3:lzexpand.dll
3:ver.dll
```

[windows.system.386], [386max], [bluemax]

This section describes the files that are copied to the SYSTEM subdirectory for 80386 machines. The following shows an example:

```
[windows.system.386]
2:cpwin386.cpl
```

```
[386max]                ; files copied if system is using 386Max®
3:386max.vxd
3:windows lod
```

```
[bluemax]               ; files copied if system is using BlueMax
3:bluemax.vxd
3:windows lod
```

[shell]

This section defines the Windows for Workgroups shell for setup. The default shell is Program Manager. The following is an example:

```
[shell]
progman.exe, "Program Manager"
```

Display Driver Sections

Windows for Workgroups display drivers require three types of files, which are specified in the [display] section.

File type	Definition
Display driver	A Windows for Workgroups display driver (.DRV) file, such as VGA.DRV. The file contains information for communication between the display adapter board and Windows for Workgroups.
Grabber file	This file (.2GR for standard mode; .3GR for 386 enhanced mode) contains code to support data exchange between Windows for Workgroups and MS-DOS-based applications. The fonts for the 386 grabbers are specified in [.3gr] sections. In Windows 3.0, these files had .GR2 and .GR3 filename extensions.
Virtual display driver	This file (VDDx.386) provides virtual display support in 386 enhanced mode.

To install a display driver other than those supplied with Windows for Workgroups, you can add the custom display driver data in the [display] entry. For setup to correctly install third-party (OEM) display drivers, you must copy the driver and the appropriate grabber files. You must also update the **286grabber=** or **386grabber=** entries in the [boot] section of SYSTEM.INI.

During custom setup, you can specify the display type. Setup copies the matching system fonts for that display type. For information about the related system font files, see “System Fonts” later in this chapter.

[display]

This section specifies information for the various display devices. The format for entries in this section is the following:

keyname= driver, description, resolution, 286grabber, logo code, VDD, 386grabber, ega.sys, logo data, optional section

Value	Definition
<i>keyname</i>	The profile of the driver, used to refer to the driver in other sections of SETUP.INF
<i>driver</i>	The filename of the display device driver
<i>description</i>	A description of the driver, enclosed in quotation marks (" "), which appears in the Change System Settings dialog box in setup
<i>resolution</i>	The aspect ratio for the display, provided as three comma-separated numbers enclosed in quotation marks (" ")
<i>286grabber</i>	The filename of the appropriate 286 grabber that supports data exchange between Windows and MS-DOS-based applications in standard mode
<i>logo code</i>	The filename for the Microsoft Windows logo code
<i>VDD</i>	The filename for the virtual display driver (VDD)
<i>386grabber</i>	The filename for the appropriate 386 grabber that supports data exchange in a virtual machine in 386 enhanced mode
<i>ega.sys</i>	The filename for the data that determines when an EGA.SYS driver needs to be installed
<i>logo data</i>	The filename for the appropriate Microsoft logo bitmap
<i>optional section</i>	The name of a section that contains additional information about the display driver

Important Don't change description strings for drivers. They're used during installation to decide to upgrade files, depending on which driver is already installed. Changing description strings can cause errors in updating to new drivers.

The following example shows the default entry for the 8514 display, which is a high-resolution IBM display:

```
[display]
8514= 2:8514.drv, "8514/a", "100, 120, 120", 3:vgacolor.2gr, 3:vgalogo.lgo, 2:vdd8514.386,
      3:vgadib.3gr,, 3:vgalogo.rle, 8514
```

(Note that this is a single command line in SETUP.INF. Don't add any carriage returns even if line-wrapping occurs.)

These optional sections in SETUP.INF also define entries for the display drivers:

```
[v7vga]      [8514s]      [v7b]      [v7e]      [xgalg]
[vgamono]    [tiga1]      [v7c]      [v7f]      [xgalo]
[8514]       [tiga2]      [v7d]      [xgasm]    [s3lg]
```

The optional sections have the following format:

file, destination, ini file, section, OldEntry, NewEntry

Value	Definition
<i>file</i>	An optional file to be copied. This item can be left null.
<i>destination</i>	The directory where the file is to be copied. This is null if <i>file</i> is null. The value is 0: if the file is to be copied to the WINDOWS directory or 0:system if the file is copied to the SYSTEM subdirectory.
<i>ini file</i>	Any .INI file, such as WIN.INI or SYSTEM.INI, that is to be modified or created. This item is optional and must be specified if <i>section</i> , <i>OldEntry</i> , or <i>NewEntry</i> are included.
<i>section</i>	The section of the .INI file that is to be modified.
<i>OldEntry</i>	Any entry, enclosed in quotation marks (" "), that is to be removed from the .INI file. This item is optional.
<i>NewEntry</i>	Any new entry, enclosed in quotation marks (" "), that is to be added in the .INI file.

For example, the following is the optional section for the 8514 driver that modifies the **dpi=** entry in the **[8514.drv]** section in SYSTEM.INI:

```
[8514]
,, system.ini, 8514.DRV, "dpi= ", "dpi= 120"
```

[.3gr] 386 Grabber Fonts

The 386 grabber supports copying text and graphics in an MS-DOS-based application for translation to Windows for Workgroups. (The 386 grabber is specified in **[display]**.) The **[.3gr]** sections identify the font files to copy for the 386 grabber:

[vga.3gr]	[v7vga.3gr]	[vgadib.3gr]	[herc.3gr]
[vga30.3gr]	[ega.3gr]	[plasma.3gr]	[tiga.3gr]

Each entry in a **[.3gr]** section identifies a filename for each font that is appropriate for the 386 grabber and display. Each entry also provides the installation disk number that contains the font. For example, the following entry specifies fonts for the 386 grabber that supports Video 7 VGA display:

```
[V7VGA.3gr]
3:CGA40WOA.FON, 3:CGA40850.FON
3:CGA80WOA.FON, 3:CGA80850.FON
3:EGA40WOA.FON, 3:EGA40850.FON
3:EGA80WOA.FON, 3:EGA80850.FON
```

If code page 437 (the default for character support in the U.S.) is used, then Setup copies the *WOA.FON files. If the code page is not 437, Setup copies the *850.FON files.

Keyboard and Code Page Sections

Specifies files for the keyboard drivers and code page support of Windows for Workgroups:

[keyboard.drivers]	[keyboard.tables]
[keyboard.types]	[codepages]

You can change these sections to use a keyboard driver or foreign language support other than that specified in SETUP.INF. If you change the keyboard support in these sections, you can also modify the **deflang=**, **defxlat=**, and **defkeydll=** entries in the **[data]** section in SETUP.INF.

[keyboard.drivers]

This section maps the actual keyboard driver filenames to the keyboard short names defined in **[keyboard.types]**.

```
[keyboard.drivers]
kbd= 3:keyboard.driv
kbdhp= 3:kbdhp.driv
```

[keyboard.types]

This section creates the keyboard short names used in the **[machine]** section, as described in the "System Configuration Sections" later in this chapter. Each entry also specifies whether a keyboard-specific DLL must be installed. The format for entries in this section is the following:

short name= description, DLL filename

Value	Definition
<i>short name</i>	The abbreviated descriptor to be used in the [machine] section.
<i>description</i>	A description, enclosed in quotation marks (" "), that appears in the Change System Settings dialog box in setup.
<i>DLL filename</i>	The filename for any supporting DLL that is required. This value can be null (nodll).

The standard entries for this section are as follows:

```
[keyboard.types]
t3s0alat= "All AT type keyboards (84 - 86 keys)", nodll
t1s2at&t= "AT&T '301' keyboard", nodll
t1s4at&t= "AT&T '302' keyboard", nodll
t4s0enha= "Enhanced 101 or 102 key US and Non US keyboards", nodll
t3s0hp1= "Hewlett-Packard Vectra keyboard (DIN)", nodll
t4s40oliv= "Olivetti 101/102 A keyboard", nodll
t1s0oliv= "Olivetti 83 key keyboard", nodll
t3s10oliv= "Olivetti 86 Key keyboard", nodll
t2s1oliv= "Olivetti M24 102 key keyboard", usadll
t1s42oliv= "PC-XT 83 key keyboard", nodll
t1s0pcxt= "PC/XT - Type keyboard (84 keys)", nodll
```

[keyboard.tables]

This section maps short names for DLLs to disk locations and filenames for the DLLs which are required by specific keyboard types. This section is used only to provide foreign language support in Windows for Workgroups.

Note Changes to this section can affect **defkeydll=** in **[data]**.

The following is an example of the **[keyboard.tables]** section:

```
[keyboard.tables]
beldll= 3:kbdbe.dll, "Belgian"
bridll= 3:kbduk.dll, "British"
cafdll= 3:kbdfc.dll, "Canadian Multilingual"
dandll= 3:kbdda.dll, "Danish"
dutdll= 3:kbdne.dll, "Dutch"
findll= 3:kbdfi.dll, "Finnish"
fredll= 3:kbdfr.dll, "French"
candll= 3:kbdca.dll, "French Canadian"
gerdll= 3:kbdgr.dll, "German"
icedll= 3:kbdic.dll, "Icelandic"
itadll= 3:kbdit.dll, "Italian"
latdll= 3:kbdla.dll, "Latin American"
nordll= 3:kbdno.dll, "Norwegian"
pordll= 3:kbdpo.dll, "Portuguese"
spadll= 3:kbdsp.dll, "Spanish"
swedll= 3:kbdsw.dll, "Swedish"
swfdll= 3:kbsdf.dll, "Swiss French"
swgdll= 3:kbdsg.dll, "Swiss German"
nodll=, "US"
usadll= 3:kbdus.dll, "US"
usddll= 2:kbdsv.dll, "US-Dvorak"
usxdll= 3:kbdusx.dll, "US-International"
```

[codepages]

Setup refers to this section when any code page other than code page 437 is installed in the system. In this case, Setup installs the correct files for the translation table and the correct OEM font (for VGA displays).

Note Changes to this section can affect **defxlat=** in **[data]**.

The format for entries in this section is the following:

codepage= xlat table, OEM font, WOA font, description

Value	Definition
<i>codepage</i>	The table number used to associate the keys on the keyboard with the appearance of characters on the display.
<i>xlat table</i>	The filename for the translation table used to associate the OEM font file with the code page.
<i>OEM font</i>	The filename and installation disk number containing the correct OEM font.
<i>WOA font</i>	The filename and installation disk number containing the correct font to be used by the Windows for Workgroups grabber.
<i>description</i>	A string, enclosed in quotation marks (" "), identifying the country or language that the code page represents.

The following are the entries in the **[codepages]** section:

```
[codepages]
863= 3:xlat863.bin, 3:vga863.fon, 3:dosapp.fon, "Canadian-French (863)"
861= 3:xlat861.bin, 3:vga861.fon, 3:app850.fon, "Icelandic (861)"
865= 3:xlat865.bin, 3:vga865.fon, 3:app850.fon, "Nordic (865)"
850= 3:xlat850.bin, 3:vga850.fon, 3:app850.fon, "Multi-Lingual (850)"
860= 3:xlat860.bin, 3:vga860.fon, 3:dosapp.fon, "Portuguese (860)"
437=, 3:dosapp.fon, "English (437)"
```

Mouse Driver Sections

The **[pointing.device]**, **[lmouse]**, and **[dos.mouse.driver]** sections define information about the pointing device for Windows for Workgroups. If you want to use a mouse driver that is not included with Windows for Workgroups, add custom mouse-driver references in these sections.

[pointing.device]

This section provides a table of information for the mouse devices that are supported by Windows for Workgroups.

The format for each entry in this section is the following:

keyname= *mouse driver*, *mouse descriptor*, *VMD*, *optional section*

Value	Definition
<i>keyname</i>	The profile name.
<i>mouse driver</i>	The filename for the mouse driver and the installation disk number.
<i>mouse descriptor</i>	A description of the driver, enclosed in quotation marks (" "), that appears in the Change System Settings dialog box in setup.
<i>VMD</i>	The filename with installation disk number for the virtual mouse device support file (for using the mouse with MS-DOS-based applications in 386 enhanced mode). If multiple VMDs are specified, they must be separated by commas and enclosed in quotation marks (" ")—for example, " x:*vmd, 2:1vmd.386 ".
<i>optional section</i>	The name of a section that contains additional information about a mouse driver.

The assignment of ***vmd** as the *VMD* value indicates that this file is built into WIN386.EXE. (this means the file is not a separate file on the installation disks.) **x**: refers to the disk where WIN386.EXE is located.

The following is an example of the **[pointing.device]** section:

```
[pointing.device]
hpmouse= 3:hpmouse.driv, "HP Mouse (HP-HIL)", x:*vmd
lmouse= 3:lmouse.driv, "Logitech", 3:lvmd.386, lmouse
ps2mouse= 3:mouse.driv, "Microsoft, or IBM PS/2", x:*vmd
genius1= 3:msscmouse.driv, "Genius serial mouse on COM1", 3:msscvm.386
genius2= 3:mssc3bc2.driv, "Genius serial mouse on COM2", 3:msscvm.386
msmouse2= 3:msscmouse.driv, "Mouse Systems serial or bus mouse", 3:msscvm.386
msmouse1= 3:mssc3bc2.driv, "Mouse Systems serial mouse on COM2", 3:msscvm.386
nomouse= 2:nomouse.driv, "No mouse or other pointing device", x:*vmd
```

[lmouse]

This section provides data to support the Logitech MS-DOS mouse driver—for example:

```
[lmouse]
3:lmouse.com, 0:,,,
```

[dos.mouse.drivers]

This section maps the Windows for Workgroups mouse driver to the appropriate MS-DOS mouse driver. Setup updates the MS-DOS mouse driver for compatibility with Windows for Workgroups—for example:

```
[dos.mouse.drivers]
mouse.sys= 7:mouse.sys, "MS DOS Mouse driver .SYS ver 7.XX"
mouse.com= 7:mouse.com, "MS DOS Mouse driver .COM ver 7.XX"
mousehp.sys= 8:mousehp.sys, "HP DOS Mouse driver .SYS ver 7.XX"
mousehp.com= 8:mousehp.com, "HP DOS Mouse driver .COM ver 7.XX"
```

Network Installation Sections

The network installation sections in SETUP.INF include the **[network]**, **[network_version]**, and **[network_specific]** sections. In particular, the **[network_version]** sections provide information for specific versions of network drivers. **[network_specific]** sections describe SYSTEM.INI entries and other information for specific networks. You can include customized entries for updated network drivers in the **[network_version]** sections, or entries in the **[network_specific]** sections to specify information for SYSTEM.INI, such as for **EMMExclude=** entries.

[network]

This section associates the network *keyname* with the files that must be installed to support the network driver from within Windows for Workgroups and contains .INI file changes for networking.

The format for entries in this section is the following:

```
network keyname= driver, description, HelpFile, optional file,
WininiSectName, SysiniSectName, VDD, VDD, . . .
```

(Note that this is a single command line in SETUP.INF. Don't add any carriage returns even if line-wrapping occurs.)

Value	Definition
<i>network keyname</i>	The profile for the specific network driver.
<i>driver</i>	The network driver filename and installation disk number.
<i>description</i>	A description of the driver, enclosed in quotation marks (" "), that appears in the Change System Settings dialog box in setup.
<i>HelpFile</i>	The filename for any associated help file.
<i>optional file</i>	The filename for any optional file to be installed.
<i>WininiSectName</i>	The section name to add to or modify in WIN.INI.
<i>SysiniSectName</i>	The section name to add to or modify in SYSTEM.INI.
<i>VDD</i>	The filenames for the virtual display device support files for MS-DOS-based applications in 386 enhanced mode. Any number of filenames can be specified.

The following are the standard entries for the **[network]** section:

```
[network]
;Prof = Driver, Description, HelpFile,OptFile,Optional section
nonet = ,"No Network Installed"           ,,,nonet_stuff
3open = ,"3Com 3+Open"
3share = ,"3Com 3+Share"                   ,,,msnet_stuff
lantastic = ,"Artisoft LANtastic"
banyan = ,"Banyan Vines"
dlr = ,"IBM OS/2 LAN Server"
pclp = ,"IBM PC LAN Program"
lanman = ,"Microsoft LAN Manager"
msnet = ,"Microsoft Network (or 100% compatible)", ,,,msnet_stuff
novell = ,"Novell NetWare"
pathworks = ,"DEC PATHWORKS"
10net = ,"TCS@ 10Net"
wfwnet = ,"Microsoft Windows for Workgroups"
wfwclient = ,"Microsoft Workgroup Client"
```

[network_version]

This section provides information about specific versions of the network drivers.

Section	Network type
[3open.versions]	3Com 3+Open
[lantastic.versions]	Artisoft LANtastic
[banyan.versions]	Banyan Vines
[pathworks.versions]	DEC PathWorks
[dlr.versions]	IBM OS/2 LAN Server
[pclp.versions]	IBM PC LAN Program
[lanman.versions]	Microsoft LAN Manager
[novell.versions]	Novell NetWare
[10net.versions]	TCS 10Net
[fwclient.versions]	Microsoft Windows for Workgroups client
[fwfnet.versions]	Microsoft Windows for Workgroups

The format for these sections is as follows:

version# = description, optional files, optional sections

Value	Definition
<i>version#</i>	The version number for the network driver
<i>description</i>	A descriptive string, enclosed in quotation marks (" ")
<i>optional files</i>	Any optional files to be installed
<i>optional sections</i>	One or more [network_specific] sections in SETUP.INF.

For example, this is the section that describes the Microsoft LAN Manager versions:

```
[lanman.versions]
xx000000="versions 1.X",,lmbasic
00020000="version 2.0 Basic",,lmbasic
01020000="version 2.0 Enhanced",,lmenh20
00021000="version 2.1 Basic",,lmbasic21
01021000="version 2.1 Enhanced",lanman21.drv,lmenh21.nodrv,lmenh21
```

These entries describe five versions of LAN Manager, including references to the [lmbasic], [lmenh20], and [lmbasic21] network-specific sections.

[network_specific]

These sections specify the data that Setup must add to WIN.INI or SYSTEM.INI, as indicated by the *WininiSectName* and *SysiniSectName* entries in the **[network]** section described earlier in this chapter.

These are the *[network_specific]* sections:

[10net41]	[dec41]	[lmenh21]
[10net41a]	[dlr12]	[lmenh21nodrv]
[10net42]	[dlr13csd]	[novell301]
[10net42def]	[dlr20]	[novell321]
[10net50]	[lant3]	[novell326]
[3open1]	[lant4]	[pclp]
[ban4]	[lant4def]	[unsupported_net]
[ban41]	[lmbasic]	[nonet_stuff]
[ban41def]	[lmbasic21]	[msnet_stuff]
[dec4]	[lmenh20]	

This is the format for these sections:

```
[network keyname]
ini-keyname= section, entry-keyname, value
```

Value	Definition
<i>network keyname</i>	The keyname defined in the [network] section
<i>ini-keyname</i>	Initialization filename (such as SYSTEM.INI)
<i>section</i>	The section name in the .INI file
<i>entry-keyname</i>	The keyname for the entry in the .INI file
<i>value</i>	The value to be assigned to the .INI entry

In addition, each section contains the **WBUppgradable=** entry. The value of this entry is defined as follows:

Value	Definition
1	Windows for Workgroups provides network components to upgrade files provided with the given network operating system.
0	Windows for Workgroups does not provide upgraded network components and displays a message to the user that the given network operating system is not supported.

For example, this is the section for Windows for Workgroups version 3.1:

```
[wfwnet31]
WBUpgradable= 1
driver= 1:wfwnet.drv
vxd= 3:vnetbios.386, 3:vnetsup.386, 2:vredir.386
vxd= 2:vserver.386, 3:vbrowse.386, 3:vwv.386
sysdir= 1:netapi.dll, 3:pmspl.dll
windir= 2:netdde.exe, 3:nddenb.dll
sysdir= 2:wfwsetup.cpl, 2:network.inf,3:lm21drv.upd
windir= 3:wfwnet.hlp
```

System Fonts Sections

Setup refers to the **[sysfonts]**, **[fixedfonts]**, and **[oemfonts]** sections to install a set of fonts to use as the system font. The **[sysfonts]** section defines the system font files, **[fixedfonts]** defines the fixed-pitch font files, and **[oemfonts]** defines OEM system font files. In each case, Setup determines the appropriate fonts to install by finding the nearest match for the display resolution defined in the **[display]** section. This is the format for entries in these sections:

font filename, description, resolution

These are the standard system fonts sections:

```
[sysfonts]
3:egasys.fon, "EGA (640x350) resolution System Font", "133, 96, 72"
3:vgasys.fon, "VGA (640x480) resolution System Font", "100, 96, 96"
3:8514sys.fon, "8514/a (1024x768) resolution System Font", "100, 120, 120"
3:egasys.fon, "AT&T (640x400) resolution System Font", "120, 96, 80"

[fixedfonts]
3:egafix.fon, "EGA (640x350) resolution Fixed System Font", "133, 96, 72"
3:vgafix.fon, "VGA (640x480) resolution Fixed System Font", "100, 96, 96"
3:8514fix.fon, "8514/a (1024x768) resolution Fixed System Font", "100, 120, 120"
3:egafix.fon, "AT&T (640x400) resolution Fixed System Font", "120, 96, 80"

[oemfonts]
3:egaoem.fon, "EGA (640x350) resolution Terminal Font (USA/Europe)", "133, 96, 72", 1
3:vgaoem.fon, "VGA (640x480) resolution Terminal Font (USA/Europe)", "100, 96, 96", 1
3:8514oem.fon, "8514/a (1024x768) resolution Terminal Font (USA/Europe)", "100, 120, 120", 1
3:egaoem.fon, "AT&T (640x400) resolution Terminal Font (USA/Europe)", "120, 96, 80", 1
```

For more information about fonts, see "Fonts Sections" later in this chapter. In addition, refer to Chapter 9, "Fonts" in the *Windows Resource Kit*, for more information.

Copy-Files Sections

Windows mode setup uses the copy-files section names to determine which groups of files to copy to the WINDOWS directory or SYSTEM subdirectory for the appropriate configuration of Windows for Workgroups. The copy-files sections are **[win.copy]** for 80286 computers, **[win.copy.net]** and **[win.copy.net.win386]** for 80386 and higher network workstations, and **[win.copy.win386]** for 80386 and higher computers.

This is the format for the entries in each copy-files section:

from, to

Value	Definition
<i>from</i>	A file or a reference to a group of files.
<i>to</i>	The destination for the files. This is either 0: to indicate the WINDOWS directory, or 0:system to indicate the SYSTEM subdirectory.

For example, the **1:myfile.txt, 0:** entry copies the file MYFILE.TXT from Disk #1 to the WINDOWS directory. If the **[win.copy.win386]** section contains a “#pwin386, 0:system” entry, Setup copies all files in the **[pwin386]** section to the SYSTEM subdirectory. If the name of an entry in any copy-files section is preceded by a pound sign (#), the entry refers to a group of files in a separate section in SETUP.INF. For example, the **[win.copy]** section is as follows:

```
[win.copy]
#net, 0:
#win.shell, 0:
#win.other, 0:system
```

The “#net, 0:” entry indicates that Setup is to copy the files in the **[net]** section to the user’s WINDOWS directory. For all items in the copy-files sections, a text string, enclosed in quotation marks (“ ”), shows the text that Setup displays to indicate which files are currently being installed.

These are the main copy-files sections:

Section	Definition
[win.copy]	Specifies that the files listed in [net] , [win.shell] , and [win.other] are to be copied for 80286 computers
[win.copy.net]	Specifies that the files listed in [net] are to be copied for network setup
[win.copy.net.win386]	Specifies that the files listed in [net] are to be copied for network setup for 80386 and higher computers
[win.copy.win386]	Specifies that the files listed in [net] , [win.shell] , [pwin386] , and [win.other] are to be copied for 80386 and higher computers

The entries the **[net]** section specify the Windows Control Panel source files and memory driver files to copy for network setup:

```
[net]
8:CONTROL.SRC, "Windows User Files"
8:WINVER
```

The entries in **[win.devices]** and **[win.devices.386]** specify the memory drivers to install as follows:

```
[win.devices]
; These devices will be copied on all machines
8:HIMEM.SYS, "XMS Memory Manager"
8:SMARTDRV.EXE, "Disk Caching Program"
8:RAMDRIVE.SYS, "RAM Drive Program"
```

```
[win.devices.win386]
; These devices will be copied on 386 machines only
8:HIMEM.SYS, "XMS Memory Manager"
6:EMM386.EXE, "LIM Expanded Memory Manager"
8:SMARTDRV.EXE, "Disk Caching Program"
8:RAMDRIVE.SYS, "RAM Drive Program"
```

The **[win.other]** entries specify various drivers, DLLs, and font files to copy, in addition to the Windows for Workgroups core files—for example:

```
[win.other]
8:WINOLDAP.MOD, "Windows Drivers"
8:WIN87EM.DLL
8:SYSEDIT.EXE
6:COMMDLG.DLL
(other entries)
```

The **[win.shell]** entries specify files for applications that are installed with the Windows for Workgroups shell—for example:

```
[win.shell]
6:PROGMAN.EXE, "Program Manager"
8:TASKMAN.EXE, "Task Manager"
5:WINFILE.EXE, "File Manager"
7:CLIPBRD.EXE, "ClipBook Viewer"
8:CLIPSRV.EXE, "ClipBook Server"
(other entries)
```

The **[pwin386]** entries specify the files that provide alternate executable files for 386 enhanced mode—for example:

```
[pwin386]
7:WINOA386.MOD, "386 Enhanced Mode files"
8:WIN386.PS2
4:WIN386.EXE
8:VTDAPI.386
(other entries)
```

[DelFiles], [RenFiles]

These sections list files that are to be deleted or renamed when Setup upgrades from Windows version 3.0 to Windows for Workgroups version 3.1. You can add or remove items from these lists, using wildcards in filenames when appropriate. In **[DelFiles]**, the first files in the list are deleted from the Windows SYSTEM subdirectory (for local setup). In **[RenFiles]**, the indicated files are renamed to enable a backup copy of the file to be created without losing the original file.

Note The **[DelFiles]** section is always processed before **[RenFiles]**.

For example, some of the standard entries in these sections are as follows:

```
[DelFiles]
tmsr?.fon
helv?.fon
swapfile.exe
kernel.exe
(other entries)
```

```
[RenFiles]
MIDIMAP.CFG, MIDIMAP.OLD
ADLIB.DRV, MSADLIB.DRV
MSCDEX.EXE, MSCDEX.OLD
```

[Win_CopyFiles]

These sections include **[win.apps]**, **[win.dependents]**, **[msmail]**, **[sched]**, **[win.games]**, **[win.scrs]**, **[win.bmps]**, and **[win.readme]**. They describe the Windows for Workgroups applications and dependents, Mail application, Schedule+ application, DLL utilities, games, screensaver, wallpaper, and README files, respectively. Each file is copied if the user requests these files during custom setup. All of the files listed are copied in Express Setup. For details about adding information for custom applications, see “Modifying .INF Files for Custom Installations” later in this chapter.

These sections have this format:

#:filename, description, filesize, profile string

Value	Definition
<i>#:filename</i>	The disk number and filename of the application or other element to be copied.
<i>description</i>	The string, enclosed in quotation marks (" "), that is to be displayed in the Windows Setup dialog box for selecting applications to copy during custom setup.
<i>filesize</i>	The expanded size of the file in bytes.
<i>profile string</i>	The string used by Setup to locate the item in the appropriate [group#] section. This string determines if and how the item is added to a Program Manager group with DDE. The <i>profile string</i> is also used to locate any needed DLL files in the [win.dependents] section.

Note The last entry in most sections specifies the total disk space required for the files in that section. If you add an entry to any section, you must also modify the disk space number—for example:
 diskspace=1506946.

The [win.apps] entries specify the Windows for Workgroups applications, utilities, and Help files—for example:

```
[win.apps]
7:CALC.EXE, "Calculator", 43072, calc
6:CARDFILE.EXE, "Cardfile", 93184, cardfile
8:CLOCK.EXE, "Clock", 16416, clock
(other entries)
```

The [win.dependents] entries specify the DLL files for utilities—for example:

```
[win.dependents]
pbrush= 8:PBRUSH.DLL
recorder= 8:RECORDER.DLL
wintutor= 6:WINTUTOR.DAT
hearts= 6:CARDS.DLL
msmail= #msmail
schdplus= #sched
```

The [msmail] entries specify the files for the Mail application:

```
[msmail]
7:SCHDMSG.DLL, "Mail Drivers"
6:WGPOMGR.DLL
7:IMPEXP.DLL
```

The **[sched]** entries specify the files for the Schedule+ application:

```
[sched]
8:MSREMIND.EXE, "Schedule+ Drivers"
5:MSSCHED.DLL
8:TRNSCHED.DLL
```

The **[win.games]** entries specify game files and their Help files—for example:

```
[win.games]
5:SOL.EXE, "Solitaire", 180688, sol
8:WINMINE.EXE, "Minesweeper", 27776, winmine
8:SOL.HLP, "Solitaire Help", 13753
8:WINMINE.HLP, "Minesweeper Help", 12754
5:MSHEARTS.EXE, "Hearts", 260144, hearts
8:MSHEARTS.HLP, "Hearts Help", 13048
(other entries)
```

The **[win.scrs]** entries specify the screensaver files—for example:

```
[win.scrs]
8:SCRNSAVE.SCR, "Default Screen Saver", 5328
8:SSMARQUE.SCR, "Marquee Screen Saver", 16896
(other entries)
```

The **[win.bmps]** entries specify the bitmap files for wallpaper, wave-form audio files, and MIDI sample files—for example:

```
[win.bmps]
8:256COLOR.BMP, "256-Color Wallpaper", 5078
8:ARCADE.BMP, "Arcade Wallpaper", 630
(other entries)
8:canyon.mid, "Canyon MIDI Song", 33883
(other entries)
```

The **[win.readme]** entries specify the README text files:

```
[win.readme]
7:readme.wri, "General Readme", 97280, readme
7:networks.wri, "Networks Readme", 22528
7:winini.wri, "Win.ini Readme", 23168
8:sysini.wri, "System.ini Readme", 58496
8:printers.wri, "Printers Readme", 37760
```

Program Manager Groups Sections

After Setup copies files to the appropriate Windows for Workgroups directories, the group sections are mapped to a group name that is used in Program Manager. Program Manager creates the groups listed in the **[new.groups]** or **[progman.groups]** sections. Related **[group#]** sections contain

entries that describe the contents of the groups that Program Manager is to create.

This is the format for entries in the **[new.groups.]** and **[progman.groups]** sections:

section= Progman group, Min/Max

Value	Definition
<i>section</i>	The name of the <i>[group#]</i> section that contains the group definition.
<i>Progman group</i>	The profile name that identifies the group in Program Manager.
<i>Min/Max</i>	A switch defining whether to minimize or maximize the group when loading Windows for Workgroups, where 1 = maximize, no value= minimize.

For details about installing a custom group in Program Manager, see “Modifying .INF Files for Custom Installations” later in this chapter.

[new.groups]

If you are updating from Windows 3.0 to Windows for Workgroups 3.1, Setup uses the **[new.groups]** section instead of the **[progman.groups]** section.

These are the standard entries for this section:

```
[new.groups]
group7= Main, 1
group1= StartUp
group2= Accessories
group6= Games
```

[progman.groups]

This section contains a list of Program Manager groups that are built for a new installation of Windows for Workgroups 3.1. These are the standard entries in this section:

```
[progman.groups]
group3= Main, 1
group4= Accessories
group5= Games
group1= StartUp
```

[group#]

After Program Manager groups are created, program items are installed in the groups based on the [group#] section entries. This is the format for entries in these sections:

description, filename.exe, icon filename, icon n, profile

Value	Definition
<i>description</i>	A string, enclosed in quotation marks (" "), that defines the text that appears below the icon in Program Manager.
<i>filename.exe</i>	The executable filename, including the extension, that is represented in the group. If <i>filename.exe</i> is null, the item is deleted from the group if present.
<i>icon filename</i>	The filename of the icon extraction file that contains the icon for the program item. If this is blank, the first icon in <i>filename.exe</i> is used.
<i>icon n</i>	The number from the icon extraction index that defines the offset of the icon specified in <i>icon filename</i> . If this item is null, the first icon is used. If you want to use the <i>n</i> th icon, use the number <i>n</i> -1. That is, for the third icon, specify 2.
<i>profile</i>	A string that defines when the item is added to the group. If this value is null, the item is always added to the group. Otherwise, the item is only added if it is installed with the Add/Remove Files dialog box in custom setup.

These are the standard entries for the [group#] sections:

```
[group3]
"File Manager", WINFILE.EXE
"Control Panel", CONTROL.EXE
"Print Manager", PRINTMAN.EXE
"ClipBook Viewer", CLIPBRD.EXE
"MS-DOS Prompt", DOSPRMPT.PIF, PROGMAN.EXE, 9
"Windows Setup", SETUP.EXE
"PIF Editor", PIFEDIT.EXE
"Read Me", README.WRI,,, readme
"Mail", MSMAIL.EXE,,,msmail
"Schedule+", SCHDPLUS.EXE,,,schdplus
```

```
[group4]
"Write", WRITE.EXE
"Paintbrush", PBRUSH.EXE,,, pbrush
"Terminal", TERMINAL.EXE,,, terminal
"Notepad", NOTEPAD.EXE,,, notepad
"Recorder", RECORDER.EXE,,, recorder
"Cardfile", CARDFILE.EXE,,, cardfile
"Calculator", CALC.EXE,,, calc
"Clock", CLOCK.EXE,,, clock
"Object Packager", PACKAGER.EXE,,, packager
"Character Map", CHARMAP.EXE,,, charmap
"Media Player", MPLAYER.EXE,,, mplayer
"Sound Recorder", SOUNDREC.EXE,,, soundrec
"Chat", WINCHAT.EXE,,, winchat
"WinMeter", WINMETER.EXE,,, winmeter
"Net Watcher", NETWATCH.EXE,,, netwatch

[group5]
"Solitaire", SOL.EXE,,, sol
"Minesweeper", WINMINE.EXE,,, winmine
"Hearts", MSHEARTS.EXE,,, hearts

[group1]
"Dr. Watson"

[group2]
"Object Packager", PACKAGER.EXE,,, packager
"Character Map", CHARMAP.EXE,,, charmap
"Media Player", MPLAYER.EXE,,, mplayer
"Sound Recorder", SOUNDREC.EXE,,, soundrec
"PIF Editor"
"Write", WRITE.EXE
"Chat", WINCHAT.EXE,,, winchat
"WinMeter", WINMETER.EXE,,, winmeter
"Net Watcher", NETWATCH.EXE,,, netwatch

[group6]
"Solitaire", SOL.EXE,,, sol
"Minesweeper", WINMINE.EXE,,, winmine
"Hearts", MSHEARTS.EXE,,, hearts

[group7]
"Clipboard Viewer"
"ClipBook Viewer", CLIPBRD.EXE
"PIF Editor", PIFEDIT.EXE
"Read Me"
"DOS Prompt",
"MS-DOS Prompt", DOS PRMPT.PIF, PROGMAN.EXE, 9
"Read Me", README.WRI,,, readme
"Tutorial"
"Mail", MSMAIL.EXE,,,msmail
"Schedule+", SCHDPLUS.EXE,,,schdplus
```

Fonts Sections

The last files that Setup copies are the raster, vector, and TrueType font files, based on the resolution of the display as defined in the **[display]** section. Three vector fonts (ROMAN.FON, MODERN.FON, and SCRIPT.FON) are resolution-independent and are installed by default.

This is the format of entries in the **[fonts]** section:

filename, description, resolution

Value	Definition
<i>filename</i>	The filename of the file containing the font
<i>description</i>	The font description, enclosed in quotation marks (" "), that appears in the Fonts dialog box in the Windows Control Panel, plus the point sizes that are installed
<i>resolution</i>	Three comma-separated numbers, enclosed in quotation marks (" "), that represent the font aspect ratio

For example, these are the font entries for VGA displays:

```
[fonts]
(other entries)
8:SSERIFE.FON, "MS Sans Serif 8, 10, 12, 14, 18, 24 (VGA res)", "100, 96, 96"
(other entries)
8:COURE.FON, "Courier 10, 12, 15 (VGA res)", "100, 96, 96"
(other entries)
8:SERIFE.FON, "MS Serif 8, 10, 12, 14, 18, 24 (VGA res)", "100, 96, 96"
(other entries)
8:SYMBOLE.FON, "Symbol 8, 10, 12, 14, 18, 24 (VGA res)", "100, 96, 96"
(other entries)
8:SMALLE.FON, "Small Fonts (VGA res)", "100, 96, 96"
(other entries)
```

This is the format for entries in the **[ttfonts]** section:

font header file, description, font file, fontfamily/flags

Value	Definition
<i>font header file</i>	The filename for the TrueType header source file.
<i>description</i>	The string, enclosed in quotation marks (" "), that appears in the Fonts dialog box. The first letter of each word in the string must be capitalized.
<i>font file</i>	The filename for the TrueType font.
<i>fontfamily/flags</i>	The name of the font family plus a flag, where 0000 is a normal-weight font, 0100 is a bold font, and 1000 is an italic font. The <i>fontfamily/flags</i> value must be enclosed in quotation marks (" ").

These are the entries for the **[tfonts]** section:

```
[tfonts]
8:ARIAL.FOT, "Arial (TrueType)", 6:arial.ttf, ""
8:ARIALBD.FOT, "Arial Bold (TrueType)", 7:arialbd.ttf, "Arial0100"
8:ARIALBI.FOT, "Arial Bold Italic (TrueType)", 6:arialbi.ttf, "Arial1100"
8:ARIALI.FOT, "Arial Italic (TrueType)", 7:ariali.ttf, "Arial1000"
8:COUR.FOT, "Courier New (TrueType)", 6:cour.ttf, ""
8:COURBD.FOT, "Courier New Bold (TrueType)", 6:courbd.ttf, "Courier0100"
8:COURBI.FOT, "Courier New Bold Italic (TrueType)", 6:courbi.ttf, "Courier1100"
8:COURI.FOT, "Courier New Italic (TrueType)", 6:couri.ttf, "Courier1000"
8:TIMES.FOT, "Times New Roman (TrueType)", 6:times.ttf, ""
9:TIMESBD.FOT, "Times New Roman Bold (TrueType)", 6:timesbd.ttf,
  "Times New Roman0100"
8:TIMESBI.FOT, "Times New Roman Bold Italic (TrueType)", 6:timesbi.ttf,
  "Times New Roman1100"
8:TIMESI.FOT, "Times New Roman Italic (TrueType)", 6:timesi.ttf, "Times New Roman1000"
8:SYMBOL.FOT, "Symbol (TrueType)", 7:symbol.ttf, ""
8:WINGDING.FOT, "Wingdings (TrueType)", 6:wingding.ttf, ""
```

Incompatible Driver Sections

Some MS-DOS drivers and TSRs are not compatible with Windows for Workgroups 3.1. These drivers potentially can cause problems if running while Setup or Windows is running. Setup checks the following sections in SETUP.INF for the names of known incompatible drivers and TSRs:

```
[compatibility]      [incompTSR1]
[incompTSR2]        [block_devices]
```

You can add to the lists of drivers and TSRs in any of these sections.

Note Some manufacturers are updating their drivers for compatibility with Windows for Workgroups. To determine whether you have the latest version of a driver, check with the driver manufacturer.

If Setup finds a command in the CONFIG.SYS file that loads any of the files listed in **[compatibility]**, Setup removes the command and leaves a blank line. You can reinstall these files at a later time.

The TSRs and drivers listed in **[incompTSR1]** can prevent setup from running and should be removed from memory and from AUTOEXEC.BAT and CONFIG.SYS before running setup. The TSRs and device drivers listed in **[incompTSR2]** can cause problems if they are running during setup or when you start Windows for Workgroups. The block devices listed in the **[block_devices]** section are not compatible with Windows for Workgroups 3.1.

Note To determine whether you have a driver that is incompatible, you can check the current entries for incompatible drivers in SETUP.INF on your disk, and the information in the related readme file in your Windows for Workgroups package. Microsoft updates the entries in the incompatible driver sections when new information is known about these and other programs. For more information about these TSRs and Setup, refer to “Troubleshooting TSRs During Setup” in Chapter 14, “Troubleshooting Windows for Workgroups.”

Miscellaneous Sections

Several sections provide information about files to update and translate filenames for OEM installation disks.

[installable.drivers]

This section specifies drivers that can be installed for multimedia support in Windows for Workgroups, including drivers for MIDI, sequencers, waveform audio, SoundBlaster boards, and device timers. This is the format for entries in this section:

keyname= filename, types, description, VxDs, default parameters

Value	Definition
<i>keyname</i>	The profile name of the driver
<i>filename</i>	The filename of the device driver with the disk number
<i>types</i>	The types of multimedia devices, enclosed in quotation marks (" ")
<i>description</i>	A descriptive string, enclosed in quotation marks (" ")
<i>VxDs</i>	Filenames for any supporting files for virtual devices
<i>default parameters</i>	The default parameters for the device if any

The standard entries for this section are as follows:

```
[installable.drivers]
msadlib= 8:msadlib.driv, "MIDI", "Ad Lib", 8:vadlibd.386,
lapc1= 8:mpu401.driv, "MIDI", "Roland LAPC1",,
midimapper= 7:midimap.driv, "MidiMapper", "MIDI Mapper",,
mpu401= 8:mpu401.driv, "MIDI", "Roland MPU401",,
sequencer= 8:mciseq.driv, "Sequencer", "[MCI] MIDI Sequencer",,
soundblaster= 8:sndblst.driv, "Wave, MIDI", "Creative Labs Sound Blaster 1.0",
8:vsbd.386,, msadlib
soundblaster2= 8:sndblst2.driv, "Wave, MIDI", "Creative Labs Sound Blaster 1.5",
8:vsbd.386,, msadlib
timer= 8:timer.driv, "Timer", "Timer", 8:vtdapi.386,
thunder= 8:sndblst2.driv, "Wave", "Media Vision Thunder Board", 8:vsbd.386,, msadlib
wave= 8:mcwave.driv, "WaveAudio", "[MCI] Sound",, "4"
cdaudio= 8:mcicda.driv, "CDAudio", "[MCI] CD Audio",,
```

[translate]

This section translates the Windows 3.0 OEMSETUP.INF entries to their Windows for Workgroups 3.1 equivalents. The disk names should appear first in this list. Some examples of entries in this section include the following:

```
[translate]
WinDiskName= 26, "Microsoft Windows 3.0 Disk"
WinDiskName= 14, "Windows Disk #"
*vddvga= 3:vddvga30.386
cti.lgo= 3:herclogo.lgo
herclogo.rle= 3:herclogo.rle
vddega= 3:vddega.386
vgacolor.gr2= 3:vgacolor.2gr
vgalogo.lgo= 3:vgalogo.lgo
vgalogo.rle= 3:vgalogo.rle
rgdi.lgo= 3:vgalogo.lgo
rgdi.rle= 3:vgalogo.rle
vga.gr3= 3:vga30.3gr
(other entries for font, keyboard, DLL, and code page filename translations)
```

[update.files]

Identifies the installable drivers that are to be updated if earlier versions of the files reside on the user's hard disk. The format for the entries in this section is as follows:

location, filename

Value Definition

location The location of the file to be updated, as either **0:** to indicate the WINDOWS directory, or **0:system** to indicate the SYSTEM subdirectory.

filename The filename of the file to be updated with the disk number.

These are the standard entries for this section:

```
[update.files]
O:system, 8:sndblst.drv
O:system, 8:sndblst2.drv
O:system, 8:mpu401.drv
O:system, 8:msadlib.drv
O:system, 8:vtdapi.386
O:system, 8:vsbd.386
O:system, 8:vadlibd.386
O:system, 8:mcicda.drv
```

[update.dependents]

This section updates dependents of files listed in **[update.files]**. These are the standard entries for this section:

```
[update.dependents]
msadlib.drv= 8:vadlibd.386
sndblst.drv= 8:vsbd.386
sndblst2.drv= 8:vsbd.386
```

[ini.upd.patches]

This section is used by MS-DOS mode setup and Windows mode setup to temporarily rename profile strings for .INI entries during setup, if the original profile string exists in the .INI file and has a value defined. The format entries in this section is as follows:

ini file, section, temporary profile, original profile

Value	Definition
<i>ini file</i>	The initialization file
<i>section</i>	The section name in the .INI file
<i>temporary profile</i>	The profile to be replaced, enclosed in quotation marks (" ")
<i>original profile</i>	The value to be added in the .INI file, enclosed in quotation marks (" ")

These are the standard entries for this section:

```
[ini.upd.patches]
system.ini, Boot, "olddrivers", "drivers"
win.ini, Desktop, "oldwallpaper", "wallpaper"
system.ini, Boot, "oldtaskman", "taskman.exe"
system.ini, Boot, "oldshell", "shell"
```

[blowaway]

This section marks the end of the installation information and the beginning of the configuration sections. This section tells Windows mode setup where to stop reading the SETUP.INF file, because Windows mode setup does not need this information.

[ini.upd.31]

This section tells Setup which lines in SYSTEM.INI and WIN.INI are to be replaced with new values in the upgrade from Windows version 3.0 to Windows for Workgroups 3.1. This is the format for entries in this section:

ini file, section, OldEntry, NewEntry, rename operator

Value	Definition
<i>ini file</i>	The .INI file that is to be changed.
<i>section</i>	A section name in the .INI file.
<i>OldEntry</i>	The old entry to be replaced, enclosed in quotation marks (" "). If this value is null, Setup assumes that the <i>NewEntry</i> is to be added.
<i>NewEntry</i>	The new entry to be added, enclosed in quotation marks (" ").
<i>rename operator</i>	Indicator for setup to retain the rename history of the profile. This value is used as a renaming mechanism during upgrades.

Some examples of the entries in this section are as follows:

```
[ini.upd.31]
(other entries)
system.ini, drivers, "midi4= adlib.drv", "midi4= msadlib.drv", rename
system.ini, 386Enh, "keyboard= ?Pbpvkd.386",
system.ini, 386Enh, "keyboard= ", "keyboard= *vkd", noclobber
system.ini, 386Enh, "device= vpicda.386", "device= vpicd.386"
system.ini, 386Enh, "device= *vpicd", "device= vpicd.386"
system.ini, 386Enh, "device= *vhd", "device= *blockdev"
system.ini, 386Enh, "device= *vpd", "device= vpd.386"
(other entries)
win.ini, FontSubstitutes,, "Helv= MS Sans Serif"
win.ini, FontSubstitutes,, "Tms Rmn= MS Serif"
win.ini, FontSubstitutes, "Courier= Courier New",
win.ini, FontSubstitutes,, "Times= Times New Roman"
win.ini, FontSubstitutes,, "Helvetica= Arial"
(other entries)
```

System Configuration Sections

These sections provide information to install the appropriate system files for the various computers. The system configuration sections include **[system]**, **[machine]**, *[cookz]*, **[special-adapter]**, **[ebios]**, and **[language]**.

[system]

This section maps the system short names used in the **[machine]** section to the appropriate system files. This section lists drivers that are selected only by the **[machine]** settings. These drivers can vary from system to system and do not have special selections in the setup menus.

The standard entries for this section are as follows:

```
[system]
system= 3:system.driv
sound= 3:mmsound.driv
comm= 3:comm.driv
hpsystem= 3:hpsystem.driv
```

[machine]

This section is used by Setup to install the appropriate system files for the various computers. Most computers use the MS-DOS System option, but some OEMs need special drivers or for specific switches to be set in the SYSTEM.INI file for compatibility.

A system description for each entry in **[machine]** appears in the System Settings list in setup. An asterisk (*) in the menu means that Setup uses the device specified in **[machine]** and overrides the description of the detected device for keyboards, mouse drivers, display adapters, and other elements.

The format for the entries in this section is:

```
keyname= description, system driv, kbd driv, kbd type, mouse driv, disp driv,
        sound driv, comm driv, himem switch, ebios, cookz
```

(Note that this is a single command line in SETUP.INF. Don't add any carriage returns even if line-wrapping occurs.)

Value	Definition
<i>keyname</i>	The profile string for the system.
<i>description</i>	A string describing the system, enclosed in quotation marks (" "), that appears in the system-description dialog box in setup.
<i>system driv</i>	The short name for the system driver identified in [system] .

Value	Definition	<i>(continued)</i>
<i>kbd drv</i>	The short name for the keyboard driver as defined in [keyboard.drivers] .	
<i>kbd type</i>	The short name for the keyboard type as defined in [keyboard.types] .	
<i>mouse drv</i>	The name of the mouse driver that is the <i>keyname</i> defined in [pointing.device] .	
<i>disp drv</i>	The name of the display driver that is the <i>keyname</i> defined in [display] .	
<i>sound drv</i>	The name of the sound driver as defined in [system] .	
<i>comm drv</i>	The name of the communications driver as defined in [system] .	
<i>himem switch</i>	A number (1–8) that identifies the A20 handler, which allows HIMEM.SYS to access extended memory. For a list of these numbers, see Chapter 14, "Troubleshooting Windows for Workgroups 3.1."	
<i>ebios</i>	The EBIOS support, defined in [ebios] .	
<i>cookz</i>	A [cookz] section (called <i>cookies</i> in the SETUP.INF comments) that contains information for modifying entries in the .INI files, usually to define [386enh] entries in SYSTEM.INI.	

Important Do not change the order of entries in the **[machine]** section.

The list of drivers in the **[machine]** section is a default list. If Setup detects a different display adapter than the type listed for the computer, the detected display driver is installed.

For example, the following shows the first three entries in the **[machine]** section:

```
[machine]
ibm_compatible= "MS-DOS System", system, kbd, t4s0enha, nomouse, egahires,
  sound, comm,, ebios,
ast_386_486= "AST@: Premium 386/25 and 386/33 (CUPID)", system, kbd,
  t4s0enha, nomouse, egahires, sound, comm,, ebios, ast_cookz
at_and_t= "AT&T PC", system, kbd, t4s0enha, nomouse, !olibw, sound, comm,, ebios,
(other entries)
```

[cookz]

This section (called *cookies* in the SETUP.INF comments) contains information for modifying entries in the .INI files, usually to define **[386enh]** entries in SYSTEM.INI as specified in the **[machine]** section. The following are the default **[cookz]** specified in the **[machine]** entries:

[apm_cookz]	[everex_cookz]	[nec_pm_cookz]
[apm_sl_cookz]	[ibml40_cookz]	[t5200_cookz]
[ast_cookz]	[ncr386sx_cookz]	[zen386_cookz]

The entries in the `[cookz]` sections have the following format:

ini file, section, cookie, needed file

Value	Definition
--------------	-------------------

<i>ini file</i>	The .INI file that is to be modified.
<i>section</i>	The section to be changed in the .INI file.
<i>cookie</i>	The specific entry to be added to the section, enclosed in quotation marks (" ")
<i>needed file</i>	Any related file that might be required

For example, the following `[cookz]` section for the AST system specifies that the entry **EMMExclude=E000-EFFF** is to be added to the **[386enh]** section in SYSTEM.INI:

```
[ast_cookz]
system.ini, 386enh, "EMMExclude= E000-EFFF",
```

The `[cookz]` section for MS-DOS systems with Advanced Power Management (APM) defines the special drivers to be copied, plus SYSTEM.INI additions:

```
[apm_cookz]
specialdriver,,, 3:power.driv
system.ini, 386enh, "device= vpowerd.386", 3:vpowerd.386
,,, 3:power.hlp
```

[special_adapter]

This section is used by MS-DOS mode setup to provide support for special adapters that require additional files for the standard Windows for Workgroups installation or other changes. For example, for the EtherLink MC adapter, the entry **DMABufferSize= 32** is placed in the **[386enh]** section of SYSTEM.INI.

The default entries in **[special_adapter]** are the following:

```
[special_adapter]
eitherlink= DMAbuffersize, 32 ; needed if EtherLink MC is detected
cdrom= 3:lanman10.386 ; needed for CD ROM extensions
```

[ebios]

This section is used by the **[machine]** section to indicate which files are copied for Extended Basic Input/Output System (EBIOS) support. The following is an example:

```
[ebios]
ebios= x:*ebios                ; required for most 386 computers
hpebios= 3:hpebios.386, x:*ebios ; required for HP 386 computers
```

[language]

This section installs the Windows for Workgroups language libraries. A language DLL must be installed for systems needing language support other than for U.S. English.

Each entry in this section has the following format:

keyname= language DLL, DLL description, language ID

Value	Definition
<i>keyname</i>	The profile string for the short name for the language
<i>language DLL</i>	The filename of the language DLL including the installation disk number
<i>DLL description</i>	A string, enclosed in quotation marks (" "), that describes the language in the Windows Control Panel
<i>language ID</i>	A special identifying code

Note Changes to this section can affect **deflang=** in **[data]**.

The following is an example of the **[language]** section:

```
[language]
dan= 3:langscs.dll, "Danish", 1030
nld= 3:langdut.dll, "Dutch", 1043
enu= "English (American)", 1033
eng= 3:langeng.dll, "English (International)", 2057
fin= 3:langscs.dll, "Finnish", 1035
fra= 3:langfrn.dll, "French", 1036
frc= 3:langeng.dll, "French Canadian", 3084
deu= 3:langger.dll, "German", 1031
isl= 3:langscs.dll, "Icelandic", 1039
ita= 3:langeng.dll, "Italian", 1040
nor= 3:langscs.dll, "Norwegian", 1044
ptg= 3:langeng.dll, "Portuguese", 2070
esp= 3:langspa.dll, "Spanish (Classic)", 1034
esn= 3:langeng.dll, "Spanish (Modern)", 3082
sve= 3:langscs.dll, "Swedish", 1053
```

APPS.INF: Application Information File

Setup places the APPS.INF file in the SYSTEM subdirectory during Windows for Workgroups installation. The APPS.INF file contains information in the following sections:

Section	Definition
[dialog]	Contains the title text used when Setup displays the Setup Applications dialog box.
[base_PIFs]	Defines a batch file for creating the _DEFAULT.PIF file and specifies settings for COMMAND.COM.
[enha_dosprompt]	Defines memory requirements for the MS-DOS command prompt under 386 enhanced mode.
[dontfind]	Lists Windows for Workgroups applications to be ignored during the Setup Existing Applications stage of installation as defined in this chapter.
[pif]	Contains PIF parameters for MS-DOS-based applications. Later sections of this chapter provide information for optimizing PIFs.

If you want to install a custom PIF for multiple systems or tailor the application setup in other ways, you can change the entries in APPS.INF.

Note Installing MS-DOS-based applications with Setup helps avoid conflicts setting up PIFs for applications that use the same executable filename.

[dontfind]

This section is used when setup is searching the hard disk for Windows-based applications. The entries in this section are overlooked by setup and are not placed in the Program Manager Applications group. This prevents setup from creating duplicate copies of standard Windows-based applications.

Some of the files in the **[dontfind]** section are listed alphabetically in the following table. Check the APPS.INF file in your SYSTEM subdirectory for the most up-to-date list.

[dontfind] filenames

APM.EXE	IMPLIBW.EXE	PAINT.EXE	STRESS.EXE
CALC.EXE	JFPRINT.EXE	PBRUSH.EXE	SWAPFILE.EXE
CALENDAR.EXE	LIBW.EXE	PIFEDIT.EXE	SYSEDIT.EXE
CARDFILE.EXE	LINKW.EXE	PLAY.EXE	TASKMAN.EXE
CHARMAP.EXE	LWINHELP.EXE	PPTGRAPH.EXE	TBOOK.EXE
CLIPBRD.EXE	MACRODE.EXE	PRINTMAN.EXE	TERMINAL.EXE
CLIPSRV.EXE	MAILSPL.EXE	PROGMAN.EXE	TRANS.EXE
CLOCK.EXE	MAPSYM32.EXE	QT.EXE	WHELP.EXE
CONTROL.EXE	MCWIN.EXE	RCPW.EXE	WIN2WRS.EXE
CONTROL3.EXE	MMSETUP.EXE	RCW.EXE	WINBBS.EXE
CVPACKW.EXE	MPLAYER.EXE	RECORDER.EXE	WINCBT.EXE
DDESPY.EXE	MSDOS.EXE	REGEDIT.EXE	WINCHAT.EXE
DEWPROJ.EXE	MSDRAW.EXE	REGLOAD.EXE	WINFILE.EXE
DIALOG.EXE	MSHEARTS.EXE	REVERSI.EXE	WINHELP.EXE
DLGEDIT.EXE	MSMAIL.EXE	SCHDPLUS.EXE	WINMETER.EXE
DRWATSON.EXE	MSREMIND.EXE	SDKPAINT.EXE	WINMINE.EXE
EQNEDIT.EXE	MXWIN.EXE	SETUP.EXE	WINTUTOR.EXE
FONTEdit.EXE	NET.EXE	SHAKER.EXE	WINVER.EXE
FTP.EXE	NETDDE.EXE	SHED.EXE	WORDART.EXE
GRAPH.EXE	NETWATCH.EXE	SOL.EXE	WPCDLL.EXE
GRAFLINK.EXE	NOTE-IT.EXE	SOUNDREC.EXE	WPWINFIL.EXE
HEAPWALK.EXE	NOTEPAD.EXE	SPOOLER.EXE	WRITE.EXE
IMAGEDIT.EXE	PACKAGER.EXE	SPY.EXE	ZOOMIN.EXE

[pif]

Complete setup information for PIFs in Windows for Workgroups 3.1 is stored in APPS.INF. You can modify this file to install PIF information for custom applications during Windows for Workgroups installation. Program information (.PIF) files that are stored in the WINDOWS directory can be created to enable MS-DOS-based applications to run in Windows for Workgroups.

A PIF is always created for any MS-DOS-based application when you install it to run under Windows for Workgroups, and a program item is added to the Applications group in Program Manager. You can add PIFs during setup by specifying the disk location of a custom PIF in the *[Win_CopyFiles]* section in SETUP.INF, or by letting Setup create a PIF from custom entries in APPS.INF.

For a list of applications for which PIF information is included in APPS.INF, see “Applications with PIF Data in APPS.INF,” in Chapter 7, “Setting Up Non-Windows Applications,” in the *Windows Resource Kit*.

The following is the format for each entry in this section:

exe file= *pif name*, *window title*, *startup directory*, *close-window flag*
icon filename, *icon n*, *standard pif*, *386 pif*, *ambiguous exe*, *optimized pifs*

(Note that this is a single command line in APPS.INF. Don't add any carriage returns even if line-wrapping occurs.)

Value	Definition
<i>exe file</i>	The filename with extension for the application's executable file.
<i>pif name</i>	The filename for the PIF without the extension.
<i>window title</i>	The string, enclosed in quotation marks (" "), that appears in the title bar.
<i>startup directory</i>	The directory that is the current directory when the application starts.
<i>close-window flag</i>	The flag that indicates whether the application window is automatically closed when the user quits the application. The value is cwe if the application window should be closed on exit. The value is null if the window should not be closed on exit.
<i>icon filename</i>	The filename from which to extract the program icon. The default is PROGMAN.EXE. In Windows for Workgroups 3.1, many PIFs list icons from MORICONS.DLL.
<i>icon n</i>	The number from the icon extraction table; the default is 0.
<i>standard pif</i>	The APPS.INF section that contains the PIF settings for standard mode; the default is [std_dflt] .
<i>386 pif</i>	The APPS.INF section that contains the PIF settings for 386 enhanced mode; the default is [enha_dflt] .
<i>ambiguous exe</i>	The APPS.INF section that lists applications with the same executable filename specified in <i>exe file</i> .
<i>optimized pifs</i>	Other APPS.INF sections that contain optimized PIF settings.

Important The **[pif]** entries must be listed alphabetically, sorted by executable filename.

The following are the entries for **[std_dflt]**, and **[enha_dflt]** which specify the values for default standard mode and 386 enhanced mode PIFs:

```
[std_dflt]
; Default is a text mode application that does not directly modify COM ports
; Other possible options are given for reference
; (or) means the entry corresponds to a radio button group
minconvmem= 128
videomode= txt      ; (or) gra
xmsmem= 0, 0        ; ##,, ## (min, max)
checkboxes=           ; c1, c2, c3, c4, kbd, nse, pps, ata, aes, ces, psc, aps, nss
```

```
[enha_dflt]
; Other possible options are given for reference
; (or) means entry corresponds to radio button group
convmem= 128, 640   ; ##, ## (Required, Limit)
emsmem= 0, 1024    ; ##, ## (Required, Limit)
xmsmem= 0, 1024    ; ##, ## (Required, Limit)
dispusage= fs      ; (or) win
execflags= ; bgd, exc
multaskopt= 50, 100 ; ##, ## (Bgd Pri, Fgd Pri)
procmemflags= dit, hma ; eml, xml, lam
dispoptvideo= txt  ; (or) lgr, hgr
dispoptports= hgr ; txt, lgr
dispflags= emt     ; rvm
otheroptions= afp  ; cwa, ata, aes, ces, psc, aps, asp, aen
```

The options for *[std_*]* and *[enha_*]* entries are the same kinds of settings you make in the PIF Editor dialog boxes for standard and 386 enhanced mode.

The following example shows the basic **[pif]** entry for an application that uses LEARN.EXE as the executable filename:

```
LEARN.EXE= LEARN, "Learning Microsoft Works",, cwe, moricons.dll, 5,
std_LEARN, enha_LEARN, amb_learn
```

(Note that this is a single command line in APPS.INF. Don't add any carriage returns even if line-wrapping occurs.)

For example, the following are the *optimized PIF* sections for Learning Microsoft Works:

```
;Learning Works (LEARN.EXE)
[std_LEARN]
minconvmem= 185
videomode= gra
xmsmem= 0, 1024
checkboxes= kbd, pps
```

```
[enha_LEARN]
dispoptvideo= hgr
convmem= 185, 520
emsmem= 0, 4096
xmsmem= 0, 4096
dispusage= win
execflags= exc
procmemflags= hma
dispoptports=
dispflags= emt, rvm
```

When you choose to have Setup install other applications to run under Windows for Workgroups, Setup displays a list of applications that use the same .EXE (execution) file. This enables you to choose which application you want to install. For example, the following are the default “ambiguous exe” entries in APPS.INF for other applications besides Microsoft Works that use LEARN.EXE as the .EXE file:

```
[amb_learn]
;MS quick pascal, learning works, learning word (LEARN.EXE)
LEARN.EXE= LEARN, "MS Quick Pascal Express 1.0",, cwe, moricons.dll, 5,
  std_PASEXPES, enha_PASEXPES
LEARN.EXE= LEARN, "Learning Microsoft Word 5.0",, cwe, moricons.dll, 5,
  std_LEARN5, enha_LEARN5
LEARN.EXE= LEARN, "Learning Microsoft Word 5.5",, cwe, moricons.dll, 5,
  std_LEARN55, enha_LEARN55
```

And, as an example, the following are the *optimized PIF* sections for Learning Microsoft Word version 5.5:

```
[std_LEARN55]
minconvmem= 180
videomode= gra

[enha_LEARN55]
dispoptvideo= hgr
convmem= 180, 195
emsmem= 0, 0
xmsmem= 0, 0
dispoptports=
```

You can include custom entries in APPS.INF if, for example, you want to use different PIFs for the same application. During setup, users are prompted to specify which PIF to install.

The following example adds PIF information for a custom application and uses a custom *optimized PIF* section:

```
XYZAPP.EXE= xyzapp, "XYZ - Example APP", C:\XYZAPP, cwe, moricons.dll, 25,  
  std_dfit, enha_dfit, opt_xyz  
  
[opt_XYZ]  
XYZAPP.EXE= xyzapp2, "XYZ - Enhanced", C:\XYZAPP, cwe, moricons.dll, 27,, enha_XYZ2,  
  
[enha_XYZ2]  
minconvmem= 204  
videomode= gra  
xmsmem= 320, 1024  
dispoptports= txt
```

Note Be sure you thoroughly understand the settings in PIF Editor before editing APPS.INF to incorporate new standard settings. For details, see Chapter 8, "PIFs and PIF Editor," in the *Windows Resource Kit*.

CONTROL.INF: The Windows Control Panel Information File

CONTROL.INF contains information that the Windows Control Panel uses during setup to install printer drivers and to provide international support. The three sections include **[io.device]**, **[io.dependent]**, and **[country]**.

In particular, the **[io.device]** and **[io.dependent]** sections provide information about printer drivers. The **[io.device]** section defines the printers supported in Windows for Workgroups. The **[io.dependent]** section defines supplementary files required for some printer drivers. If you want to install a printer driver during setup that is not supplied with Windows for Workgroups, add references in these two sections.

A custom printer driver can use the fonts already installed in Windows for Workgroups or may require additional files to be installed and specified in the **[sysfonts]**, **[fixedfonts]**, and **[oemfonts]** sections of SETUP.INF.

[io.device]

This section lists all of the supported printers for Windows for Workgroups. This section is used by the Windows Control Panel to install printer support. The format for each entry in this section is:

filename, description, scaling string, scaling string

Value	Definition
<i>filename</i>	The filename and extension with the installation disk number.
<i>description</i>	The string, enclosed in quotation marks (" "), that appears in the Windows Control Panel and is used in WIN.INI.
<i>scaling string</i>	One or two strings, enclosed in quotation marks (" "), that indicate the scaling for the device.

A driver can have more than one entry in **[io.device]** to specify settings that correspond to different printers.

The following are some examples of the entries in this section:

```
[io.device]
9:TTY.DRV,"Generic / Text Only","DEVICESPECIFIC"
9:pscript.DRV,"Agfa 9000 Series PS","DEVICESPECIFIC"
9:pscript.DRV,"Agfa Compugraphic 400PS","DEVICESPECIFIC"
9:HPPCL.DRV,9:unidrv.dll,"Agfa Compugraphic Genics","DEVICESPECIFIC"
9:pscript.DRV,"Apple LaserWriter","DEVICESPECIFIC"
9:pscript.DRV,"Apple LaserWriter II NT","DEVICESPECIFIC"
9:pscript.DRV,"Apple LaserWriter II NTX","DEVICESPECIFIC"
9:pscript.DRV,"Apple LaserWriter Plus","DEVICESPECIFIC"
9:HPPCL.DRV,9:unidrv.dll,"Apricot Laser","DEVICESPECIFIC"
9:pscript.DRV,"AST TurboLaser/PS","DEVICESPECIFIC"(other entries)
```

[io.dependent]

The Windows Control Panel checks this section in case the printer driver requires supplementary files to be copied to the SYSTEM subdirectory. The files to be copied can be soft font installers, font files, help files, or other support for the printer. The format for each entry in this section is:

driver filename= files to be copied with the driver

The following are examples of the default entries in this section:

```
[io.dependent]
pscript.driv = 9:pscript.hlp, 9:testps.txt
tty.driv     = 9:tty.hlp
ibm4019.driv = 9:sf4019.exe
lbp111.driv  = 9:can_adf.exe
lbp11.driv   = 9:can_adf.exe
hppcl.driv   = 9:fininstall.dll, 9:fininstall.hlp, 9:unidrv.hlp, 9:BP1CP2.PCM,, 9:DD1CP1.PCM
hppcl5ms.driv = 9:fininstall.dll, 9:fininstall.hlp, 9:unidrv.hlp, 9:BP1CP2.PCM,, 9:DD1CP1.PCM
(other entries)
```

[country]

This section defines the international formats that are supported by the Windows Control Panel. Make modifications to the international formats only by choosing the International icon in the Windows Control Panel.

For example, the following are the two entries for Canada (the exclamation point [!] is a field delimiter):

```
"Canada (English)", "2!1!0!2!1!1!0!2!1!1!!!$!, !./!/:!, !dd/MM/yy!MMMM d, yyyy!ENG"
"Canada (French)", "2!2!3!2!1!1!0!2!8!1!!!$! !, !/!/:!;!yy-MM-dd!d MMMM, yyyy!FRC"
```

The [country] section contains entries for the following countries:

Australia	Germany	Spain
Austria	Iceland	Sweden
Belgium (Dutch)	Ireland	Switzerland (French)
Belgium (French)	Italy	Switzerland (German)
Brazil	Mexico	Switzerland (Italian)
Canada (English)	Netherlands	Taiwan
Canada (French)	New Zealand	United Kingdom
Denmark	Norway	United States
Finland	Portugal	Other Country
France	South Korea	

NETWORK.INF: Network Information File



NETWORK.INF contains specific information used to install network adapter cards and network transport protocols supported in Windows for Workgroups 3.1. Setup places NETWORK.INF in the SYSTEM subdirectory during Windows for Workgroups installation. This section describes how and when to modify entries in NETWORK.INF.

NETWORK.INF is organized into the following types of sections:

- General installation
- Network adapter drivers
- Network protocol drivers
- Secondary network drivers

The sections contained within NETWORK.INF are described in the following table. NETWORK.INF information contains additional entry definitions in “Installation Sections Information” and “Protocol Sections Information.”

<i>Type of section</i>	<i>Section</i>	<i>Purpose</i>
General installation	[nwsdata]	Defines default parameters used for the network setup.
	[workgroup]	Defines common Windows for Workgroups network driver files.
	[protman_install]	Specifies the Protocol Manager driver files.
	[protman]	Defines the Protocol Manager PROTOCOL.INI section.
Network adapter drivers	[netcard]	Defines network adapter drivers supported by Windows for Workgroups.
	[netcard_install]	Defines network driver files for a specific network adapter.
	[netcard_protocol]	Defines the PROTOCOL.INI section for a specific network adapter.
Network protocol drivers	[transport]	Defines the set of network protocols supported by Windows for Workgroups.
	[transport_install]	Defines the driver files for a specific transport protocol.
	[transport_protocol]	Defines the PROTOCOL.INI section for a specific transport protocol.
Secondary network drivers	[multinet]	Defines the set of secondary network drivers supported by Windows for Workgroups.
	[multinet_install]	Defines the driver files for a specific secondary network.

Note You can include customized entries for updated network drivers in the **[netcard]** sections, or entries in the *[netcard_install]* and *[netcard_protocol]* sections to specify information for SYSTEM.INI and PROTOCOL.INI. All sections and entries that are prefixed with the characters “MS\$” are reserved for Microsoft use only.

General Installation Sections

In NETWORK.INF, the **[nwsdata]**, **[workgroup]**, **[protman_install]**, and **[protman]** sections define installation options and specify filenames and locations of related files for the network setup in Windows for Workgroups setup.

[nwsdata]

This section provides the default settings and information for setup and the Windows Control Panel regarding the network setup. The following are the standard entries for this section:

```
[nwsdata]
version= "3.1.032"           ; current Windows for Workgroups version
autoexec_anchor= "net start" ; line in AUTOEXEC.BAT where the network drivers
                             are to be inserted
config_anchor= "workgrp.sys" ; line in CONFIG.SYS where the network drivers are
                             to be installed
default_transport= ms$netbeui ; keyname of the default NDIS protocol
```

Note The only entry in this section that might need to be modified is **default_transport=**. If the standard protocol for a workgroup or company is not NetBEUI, add the new protocol to the **[transport]** section and reference the entry as **default_transport=**.

[workgroup]

This section describes the general installation options and files for all Windows for Workgroups configurations, regardless of which network adapter or secondary networks are installed. Refer to “Installation Sections Information” later in this chapter for specific definitions of the keynames for these entries.

The following is an example of the **[workgroups]** section:

```
[workgroup]
descr= "Workgroup Drivers"
install= delete_drivers
devdir= 9:workgrp.sys
netdir= A:net.exe, 9:net.msg, 9:neth.msg
device= workgrp.sys, @devdir\workgrp.sys, protman.dos
sysparm= 386enh, netheapsize, 20, minval
sysparm= standard, netheapsize, 8, minval
lastdrive= Z
```

[protman_install]

This section provides setup with the installation options and files for the Windows for Workgroups 3.1 Protocol Manager driver. The Protocol Manager links NDIS-compliant network adapter drivers and protocol drivers. Refer to "Installation Sections Information" later in this chapter for specific definitions of the keynames for these entries.

The following is an example of the **[protoman_install]** section:

```
[protman_install]
descr= "Microsoft Protocol Manager"
devdir= A:protman.dos, A:protman.exe
device= protman.dos, "@devdir\protman.dos /i:@devdir;
tsr= netbind
tsr= "net start", "@netdir\net start"
```

[protman]

This section defines the Protocol Manager section for the PROTOCOL.INI file. Refer to "Protocol Sections Information" later in this chapter for specific definitions of the keynames for these entries. The format for the entries in this section is:

```
[protman]
drivename= PROTMAN$
param= PRIORITY,,static, MS$NETBEUI
```

Network Adapter Driver Sections

The **[netcard]**, **[netcard_install]**, and **[netcard_protocol]** sections define network adapter hardware that can be installed with Windows for Workgroups.

[netcard]

This section defines the set of network adapter drivers that can be selected from Setup or from the Windows Control Panel. The network adapter card entries are organized alphabetically by *keyname*. If you add a new entry, be sure the network card *keynames* remain in alphabetical order.

The format for entries in this section is:

keyname= *description*, *MSID*, *type*, *media*, *mode*, *install section*,
protocol section, *style*

(Note that this is a single command line in NETWORK.INF. Don't add any carriage returns even if line-wrapping occurs.)

Value	Definition
<i>keyname</i>	The profile of the network driver that is used to refer to the driver in the [network.setup] section of PROTOCOL.INI (maximum 11 characters).
<i>description</i>	A description of the driver, enclosed in quotation marks (" "), that appears in the Install New Network Adapter dialog box in setup and in the Windows Control Panel.
<i>MSID</i>	The Microsoft ID for this adapter. Zero is valid for an unknown ID. Values from 1-999 are reserved for assignment by Microsoft. Values from 1,000 to 65,535 can be used by OEM vendors.
<i>type</i>	The type of network adapter driver:
ndis	NDIS 2.x compatible
ndis_sngl	NDIS 2.x compatible, single device driver loaded in CONFIG.SYS for multiple cards
mono	Monolithic device driver, not NDIS-compatible
<i>media</i>	The network media type for the network adapter: arcnet , baseband , broadband , ethernet , FDDI , tokenring , or serial . This field may be left empty for other media types.
<i>mode</i>	The operating system mode for the device driver: real Real mode (DOS) device driver vxd Windows 386 enhanced mode device driver
<i>install section</i>	The section name that contains file installation information for this driver (maximum 19 characters).
<i>protocol section</i>	The section name that contains file information for this driver in PROTOCOL.INI (maximum 19 characters).
<i>style</i>	This value specifies compatibility of the adapter driver with other network components. The remaining bit positions are reserved for future use by Microsoft. 0x0001 Not compatible with protected-mode protocols (mode= vxd) 0x0002 Not compatible with IPX protocols (transport= IPX)

Here is an example of a standard entry for the **[netcard]** section:

```
[netcard]
(other entries)
ms$ee16= "Intel EtherExpress 16 or 16TP",301,ndis,ethernet,real,ms$ee16,ms$ee16_nif
(other entries)
```

[netcard_install]

This section provides Setup with an installation section for each entry in the **[netcard]** section. Each section name can be unique to the driver or shared by a number of network adapter cards that use the same driver files. This section defines the files that must be installed and the configuration options for SYSTEM.INI, WIN.INI, and the DOS files CONFIG.SYS and AUTOEXEC.BAT.

For example, the entries in the *[netcard_install]* section for the Intel EtherExpress 16 or 16TP are:

```
[ms$ee16]
devdir= A:exp16.dos
device= exp16.dos,@devdir\exp16.dos
```

Refer to “Installation Sections Information” later in this chapter for specific definitions of the keynames for these entries.

[netcard_protocol]

This section specifies information that Setup must add to PROTOCOL.INI for each of the entries in the **[netcard]** section. Each section name can be unique to the driver or shared by a number of network adapters that use common driver settings. Following is an example of the format for this section using the Intel EtherExpress 16 or 16TP:

```
[ms$ee16_nif]
drivename= EXP16$
io=IOADDRESS,,text,"0x200,0x210,0x220,0x230,0x240,0x250,0x260,0x270,0x300,0x310,
  0x320,0x330,0x340,0x350,0x360,0x370","0x300",0x04
irq= IRQ,,text,"3,4,5,9,10,11",3,0x04
; NOTE: do not change the order of Early, Late, Never.
; ncd_card_specific relies on this ordering.
param= IOCHRDY,"I/O Channel Ready",text,"Early,Late,Never",Late,0x04
param= TRANSCEIVER,"Transceiver Type",text,"Thick Net (AUI/DIX)",
  'Thin Net (BNC/COAX)',Twisted-Pair (TPE)',"Thin Net (BNC/COAX)",0x04
```

Refer to “Protocol Sections Information” later in this chapter for specific definitions of the keynames for these entries.

Network Protocol Driver Sections

The **[transport]**, **[transport_install]**, and **[transport_protocol]** sections define the NDIS network transport protocols that can be installed with Windows for Workgroups.

[transport]

This section defines the set of network protocols that can be selected from setup or from the Windows Control Panel. The format for entries in this section is:

*keyname= description, MSID, type, transport, mode, exclusive,
lanabase method, install section, protocol section, winnet type, style*

(Note that this is a single command line in NETWORK.INF. Don't add any carriage returns even if line-wrapping occurs.)

Value	Definition
<i>keyname</i>	The profile of the transport driver. Used to refer to the driver in the [network.setup] section of PROTOCOL.INI (maximum 11 characters).
<i>description</i>	A description of the driver, enclosed in quotation marks (" "), that appears in the Protocol dialog box in setup and in the Windows Control Panel.
<i>MSID</i>	The Microsoft ID for this adapter. Zero is valid for an unknown ID. Values from 1-999 are reserved for assignment by Microsoft. Values from 1,000 to 65,535 can be used by OEM vendors.
<i>type</i>	The type of network protocol driver: ndis NDIS 2.x compatible non-netbios NDIS 2.x compatible, but not NETBIOS compatible
<i>transport</i>	The transport protocol type used by the protocol driver: ipx , netbeui , tcp , xns . This field is left empty for other transport types.
<i>mode</i>	The operating system mode for the device driver: real Real mode (DOS) device driver vxd Windows 386 enhanced mode device driver
<i>exclusive</i>	The exclusive Identifier specifies compatibility with other transport protocols: 0 Protocol is compatible with other NDIS 2.x protocols 1 Protocol is not compatible with other NDIS 2.x protocols and must be installed at NETBIOS lana zero. Only one exclusive protocol can be installed in the system
<i>lanabase method</i>	The method used to specify the lanabase in PROTOCOL.INI: implicit No lanabase specified lanabase lanabase= is specified in PROTOCOL.INI drivername Lanabase is specified in the driver name after the dollar sign (\$)

Value	Definition	(continued)
<i>install section</i>	The section name that contains file installation information for this driver (maximum 19 characters).	
<i>protocol section</i>	The section name that contains file information for this driver for PROTOCOL.INI (maximum 19 characters).	
<i>winnet type</i>	The WNNC Net Type for the secondary network associated with a protocol that must be used with a particular secondary network, such as IPX with NetWare. This entry is left blank for standard NDIS 2.x, NETBIOS-compatible protocols.	
<i>style</i>	This value specifies compatibility with other network components. The remaining bit positions are reserved for future use by Microsoft. 0x0001 Not compatible with protect-mode network adapter cards (mode= vxd)	

The following are standard entries for this section:

```
[transport]
ms$netbeui= "Microsoft NetBEUI",1,ndis,netbeui,,0,lanabase,ms$netbeui,
  ms$netbeui_xif
ms$ipx= "Novell IPX",101,nonetbios,ipx,real,1,implicit,ms$ipx,ms$ipx_xif,0x300
ms$ipx_tr= "Novell IPX (Token Ring)",102,nonetbios,ipx,real,1,implicit,ms$ipx_tr,
  ms$ipx_tr_xif, 0x300
```

[transport_install]

This section provides Setup with an installation section for each entry in the **[transport]** section. Each section name can be unique to the driver or shared in common by a number of transports that use the same driver files. This section defines the files that must be installed and the configuration options for SYSTEM.INI, WIN.INI, and MS-DOS files CONFIG.SYS and AUTOEXEC.BAT.

The following is the format for entries in this section. Refer to "Installation Sections Information" later in this chapter for specific definitions of the keynames for these entries:

```
[ms$netbeui]
sysdir= A:\vnb.386
transport= vnb.386,vnb.386

[ms$ipx]
;Standard IPX
devdir= A:\msipx.sys
netdir= A:\msipx.com
sysdir= A:\vipx.386
secondnet= vipx.386,vipx.386
device= msipx.sys,@devdir\msipx.sys
load= msipx,@netdir\msipx,MSCDEX
```

[transport_protocol]

This section specifies information that Setup must add to PROTOCOL.INI for each of the entries in the **[transport]** section. Each section name can be unique to the driver or shared by a number of network adapters that use common driver settings. Refer to "Protocol Sections Information" later in this chapter for specific definitions of the keynames for these entries.

Here are examples of the **[transport_protocol]** sections:

```
[ms$netbeui_xif]
drivename= netbeui$
param= SESSIONS,"Maximum Sessions",int,"3,117,1",10
param= NCBS,"NCBS",int,"7,255,1",32
```

```
[ms$ipx_xif]
DriverName= IPX$
; NOTE: do not change the order of Ethernet_II, Novell/Ethernet.
; ncd_trans_specific relies on this ordering.
param= MediaType,"Adapter Media Type",text,"Novell/Ethernet,'Ethernet_II (DIX)',",
      "Novell/Ethernet"
```

Secondary Network Driver Sections

The **[multinet]** and **[multinet_install]** sections define compatible secondary networks that can be installed with Windows for Workgroups.

[multinet]

This section defines the set of secondary networks that can be selected from setup or from the Windows Control Panel. The format for entries in this section is:

keyname= description, net type, exclusive, protocol ID, install section, style

Value	Definition
<i>keyname</i>	The profile for the secondary network. Used to refer to the network in the multinet= entry of the [network] section in SYSTEM.INI.
<i>description</i>	A description of the secondary network, enclosed in quotation marks (" "), that appears in the Other Networks dialog box in setup and in the Windows Control Panel.
<i>net type</i>	The WNNC Net Type that is associated with a protocol that must be used with a particular secondary network, such as IPX with NetWare.

<i>Value</i>	<i>Definition</i>	<i>(continued)</i>
<i>exclusive</i>	The exclusive identifier specifies compatibility with other secondary network drivers: 0 Secondary network is multi-net aware and can be used with other secondary network drivers 1 Secondary network is not multi-net aware and cannot be used with other secondary network drivers	
<i>protocol ID</i>	The MSID of the required protocol for this secondary network. The protocol referenced by this ID is added to the first available network adapter when the secondary network is added (maximum 19 characters).	
<i>install section</i>	The section name that contains file installation information for this driver (maximum 19 characters).	
<i>style</i>	This value specifies compatibility with other network components. Reserved for future use.	

Here are the standard entries for this section:

```
[MultiNet]
;key = description, net_type, exclusive, protocol id, install_section, style
lanman="Microsoft LAN Manager",0x200,0,0,mn$lanman
netware="Novell NetWare",0x0300,1,101,mn$netware
```

[multinet_install]

This section provides Setup with an installation section for each entry in the **[multinet]** section. Each section name can be unique to the driver or shared by a number of transports that use the same driver files. This section defines the files that must be installed and the configuration options for SYSTEM.INI, WIN.INI, and the DOS files CONFIG.SYS and AUTOEXEC.BAT.

The following is the format for entries in this section. The material that directly follows, "Installation Sections Information," contains specific definitions of the keynames for these entries.

```
[mn$lanman]
sysdir= A:\mscript.exe
sysdir= A:\mscript.pif

[mn$netware]
install=delete_netware
netdir=A:\netx.com
windir=A:\nwpopup.exe
sysdir=A:\vnetware.386,A:\netware.driv,A:\netware.hlp
secondnet.driv=netware.driv,netware.driv
secondnet=vnetware.386,vnetware.386
sysparm=NetWare,NWShareHandles,FALSE,noclobber
sysparm=NetWare,RestoreDrives,TRUE
sysparm=386Enh,OverlappedIO,off
winparm=windows,load,nwpopup.exe,append
load=netx,@netdir\netx,route,msipx
lastdrive=p
```

Installation Sections Information

The install sections, including **[netcard]**, **[transport]**, and **[multinet]**, define a set of commands required to install a network driver. These sections contain the following entries:

devdir= *filelist*

Specifies the files that are to be copied to the device directory. Windows for Workgroups defines the device directory as the WINDOWS directory. If Setup detects that Windows is being installed on a removable drive or on a drive that uses Stacker, the device is defined as the root of the drive where CONFIG.SYS and AUTOEXEC.BAT are located.

device= *lookupname, commandline, after1, after2*

Specifies a device driver to add to the **device=** entry in CONFIG.SYS. See the following table for more details.

install= *section_name*

The section name of another installation section that is to be run.

lastdrive= *lastdrive*

Specifies the maximum allowed **lastdrive=** value in CONFIG.SYS. If the **lastdrive=** value specified in CONFIG.SYS is less than the value specified in this entry, the CONFIG.SYS value remains the same. If the value specified in CONFIG.SYS is greater than the value of this entry, CONFIG.SYS is modified to the value of the **lastdrive=** entry.

load= *lookupname, commandline, after1, after2*

Specifies a device driver to add to AUTOEXEC.BAT *after* the **net start** command (binding). See the following table for more details.

netcard= *lookupname, commandline*

Specifies a network adapter device driver to add to the **netcard=** entry of the **[386enh]** section of SYSTEM.INI. See the following table for more details.

netdir= *filelist*

Specifies the files that are to be copied to the network directory. Windows for Workgroups defines the network directory as the WINDOWS directory.

secondnet.drv= *lookupname, commandline*

Specifies a secondary network driver to add to the **secondnet=** entry of the **[boot]** section of SYSTEM.INI. See the following table for more details.

secondnet= *lookupname, commandline*

Specifies a secondary network to add to the **[boot]** section of SYSTEM.INI. See the following table for more details.

sysdir= *filelist*

Specifies the files that are to be copied to the Windows SYSTEM subdirectory.

sysparm= *section, keyname, value, edit option*

Provides additional system parameters that are to be added to the specified *section* and *keyname* in SYSTEM.INI. See the following table for more details.

transport= *lookupname, commandline*

Specifies a protocol device driver to add to the **transport**= entry of the **[386enh]** section of SYSTEM.INI. See the following table for more details.

tsr= *lookupname, commandline, after1, after2*

Specifies a device driver to add to AUTOEXEC.BAT *before* the **net start** command (binding). See the following table for more details.

vxd= *lookupname, commandline*

Specifies a network device driver to add to the **[386enh]** section of SYSTEM.INI. See the following table for more details.

windir= *filelist*

Specifies the files that are to be copied to the WINDOWS directory.

winparm= *section, keyname, value, edit option*

Provides additional system parameters that are to be added to the specified *section* and *keyname* in WIN.INI. See the following table for more details.

These entries can use all or some of the values that are defined in the following table:

device= **load=** **netcard=** **secondnet=**
secondnet.driv= **transport=** **tsr=** **vxd=**

Value	Definition
<i>lookupname</i>	The minimum test used to look up the line that contains the specified driver in CONFIG.SYS, AUTOEXEC.BAT, or SYSTEM.INI.
<i>commandline</i>	The command line that is to be added to CONFIG.SYS, AUTOEXEC.BAT, or SYSTEM.INI. For device= , the <i>commandline</i> specifies only the right side of the DEVICE= line in CONFIG.SYS. If prefixed by an at sign (@), the <i>commandline</i> can specify any number of copy keywords, including netdir= , windir= , sysdir= , or devdir= , that are to be replaced with the path corresponding to the keyword—for example: device= protman.dos, @devdir\protman.dos This entry is expanded in CONFIG.SYS as the following: device= c:\windows\protman.dos
<i>after1</i>	The name of the driver that this driver must follow.
<i>after2</i>	The name of the driver that this driver must follow only if <i>after1</i> is not located.

The **sysparm=** and **winparm=** entries use the same format and can use all or some of the values that are defined in the following table:

Value	Definition
<i>section</i>	The section for the entry to be added to SYSTEM.INI
<i>keyname</i>	The keyname for the entry to be added to SYSTEM.INI
<i>value</i>	The value for the entry
<i>edit option</i>	The edit option for the value: noclobber If a value already exists for the entry, don't modify the value minval If a greater value exist for the entry, don't modify the value maxval If a lesser value exists for the entry, don't modify the value append If the specified value does not exist, append the value preceded by a space (used for the load= or run= entries in WIN.INI.)

Protocol Sections Information

The protocol sections list the set of entries that can define an NDIS network adapter or protocol driver for the PROTOCOL.INI file. Here is an example of the `[transport_protocol]` section:

```
[ms$netbeui_xif]
drivename= netbeui$
param= SESSIONS,"Maximum Sessions",int,"3,117,1",10
param= NCBS,"NCBS",int,"7,255,1",32
```

These sections contain the following entries:

drivename= drivename

Specifies the DOS device name for this protocol driver.

dma= <fields>

Specifies the DMA channel used by the network adapter. See the following tables for field definitions and details.

io= <fields>

Specifies the base I/O port used by the network adapter. See the following tables for field definitions and details.

irq= <fields>

Specifies the interrupt request line settings used by the network adapter. See the following tables for field definitions and details.

mem= <fields>

Specifies the base RAM address used by the network adapter. See the following tables for field definitions and details.

param= <fields>

Specifies any other parameters that are used by the driver. The **param=** entry can be used more than once in a section. The **param=** entries are also used to define the special features of some network adapter cards, such as the information required for use with multiple configurations—for example, the Primary and Alternate entries for IBM Token Ring cards. Setup handles the **param=** entries automatically. See the following tables for field definitions and details.

rom= <fields>

Specifies the base ROM address used by the network adapter. See the following tables for field definitions and details.

slot= <fields>

Specifies the card slot number where the network adapter is installed. See the following tables for field definitions and details.

This table describes the fields used by all [*transport_protocol*] entries except **drivername=**:

Value	Definition
<i>keyname</i>	The left-side keyname in the PROTOCOL.INI section for this driver.
<i>description</i>	The text description of the keyname, enclosed in quotation marks (" "), which appears in the Advanced Network-Adapter Settings and the Advanced Protocol Settings dialog boxes.
<i>type</i>	The parameter type (shown in the following table).
<i>range</i>	The range of values for the parameter (shown in the following table).
<i>default</i>	The default value for the parameter (shown in the following table).
<i>flags</i>	The modifier for the parameter: 0x01 Range and default value fields specify absolute memory addresses rather than paragraph addresses 0x02 Optional parameter that is not added by default to PROTOCOL.INI 0x04 Parameter value is set by the device driver on the network adapter card 0x08 Reserved

The following table describes the *type*, *range*, and *default* values for the protocol section entries.

Type	Range	Default	Definition
static	Default	N/A	Non-modifiable
text	"value, value, ..."	Yes	List of text values
int	"first, last, step"	Yes	Integer range
hex	"first, last, step"	Yes	Integer range (hexadecimal format)
chars	Number of characters	Yes	Any characters
keyonly	N/A	N/A	No right-side value or equal sign

OEMSETUP.INF: OEM Network Setup Information File



The OEMSETUP.INF file is used to define third party, original equipment manufacturer (OEM) network card drivers and additional transport protocol information for network cards that are not provided with Windows for Workgroups 3.1. Setup uses this information to configure and install the additional OEM network components. Setup places OEMSETUP.INF in the SYSTEM subdirectory during Windows for Workgroups installation, or when additional network components are installed through the Network icon in the Windows Control Panel.

OEMSETUP.INF contains the **[disk]**, **[netcard]**, **[netcard_install]**, **[netcard_protocol]**, **[transport]**, **[transport_install]**, **[transport_protocol]**, **[multinet]**, and **[multinet_install]** sections. Because the information in OEMSETUP.INF is similar to the NETWORK.INF file, refer to “NETWORK.INF: Network Information File” earlier in this chapter for detailed information about the formats used for entries.

[disk]

This section identifies the disks that contain the OEM network information that can be requested during setup installation. The **[disks]** section uses the same syntax as the **[disks]** section described in SETUP.INF—refer to “SETUP.INF: Setup Information File” earlier in this chapter for detailed information—for example:

```
[disks]
1=, "OEM Drivers Disk #1", disk1
```

Tip OEMSETUP.INF should be placed in the root directory of the OEM driver disk. However, drivers referenced in OEMSETUP.INF may be located in a subdirectory on the same floppy disk and referenced by using the *path* parameter. For example, if you already have a LAN Manager 2.1 compatible drivers disk, you can specify two different subdirectories as two disk numbers but let each refer to the same physical disk. Setup will copy the drivers from the specified subdirectory. For example:

```
[disks]
1 = \mslanman.dos\drivers\ethernet\oem, "OEM Drivers Disk #1", disk 1
2 = \mslanman.dos\drivers\protocol\tcp, "OEM Drivers Disk #1", disk 1
(other entries)

[oem$card]                ; refers to an NDIS adapter card driver
devdir= 1:oemdrv.dos

[oem$tcp]                  ; refers to an NDIS transport protocol
devdir= 2:oemtcp.dos
```

[netcard]

This section defines the OEM network adapter drivers that can be selected from Windows for Workgroups setup or from the Windows Control Panel. The network adapter card entries should be listed alphabetically, sorted by *netcard keyname*. The following is the format for the entries in this section:

keyname= description, MSID, type, media, mode, install section, protocol section, style

(Note that this is a single command line in OEMSETUP.INF. Don't add any carriage returns even if text wraps to the next line.)

The following is an example of an entry for the **[netcard]** section. This example defines a network adapter card called "ACME Ethernet Adapter 1," and references a protocol driver section (acme_001) and a network installation section (acme_001_nif):

```
[netcard]
acme_001=, "ACME Ethernet Adapter 1", 0, ndis, ethernet, real, acme_001, acme_001_nif
```

[netcard_install]

This section provides Setup with an installation section for each OEM entry in the **[netcard]** section. Each section name can be unique to the driver or shared by a number of network adapter cards that use the same driver files. This section defines the files that must be installed and the configuration options for SYSTEM.INI, WIN.INI, and the DOS files CONFIG.SYS and AUTOEXEC.BAT.

Following are sample entries for the **[netcard_install]** section that define the network adapter card driver that needs to be installed to support the "ACME Ethernet Adapter 1" network adapter card:

```
[acme_001]
devdir= 1:acme01.dos
device= acme01.dos, @devdir\acme01\acme01.dos
```

[netcard_protocol]

This section specifies the information that Setup must add to PROTOCOL.INI for each of the entries in the **[netcard]** section. Each section name can be unique to the driver or shared by a number of network adapters that use common driver settings.

Following are sample entries for the `[netcard_protocol]` section that define the network adapter card settings that need to be specified to configure the “ACME Ethernet Adapter 1” network adapter card:

```
[acme_001_nif]
drivername= ACME$           ; driver name defined by NDIS driver
irq= INTERRUPT,, int, "2,7,1", 3
io= IOADDRESS,, hex, "0x200, 0x3f0, 0x10", 0x300
param= ACMEPARAM, "Special ACME Driver Parameter", text, "A,B,C,D", A
```

[transport]

This section defines the set of OEM network protocols that can be selected from setup or from the Windows Control Panel. The format for entries in this section is:

```
keyname= description, MSID, type, transport, mode, exclusive,
         lanabase method, install section, protocol section, winnet type, style
```

(Note that this is a single command line in OEMSETUP.INF. Don't add any carriage returns even if line-wrapping occurs.)

Following is an example of the `[transport]` section that defines an NDIS transport protocol to install which is called “ACME TCP Protocol” and references a protocol driver section (`acme_tcp`) and a network installation section (`acme_tcp_xif`):

```
[transport]
acme_tcp= , "ACME TCP Protocol", 0, ndis, tcp, vxd, 0, lanabase, acme_tcp, acme_tcp_xif
```

[transport_install]

This section provides Setup with an installation section for each OEM entry in the `[transport]` section. Each section name can be unique to the driver or shared by a number of transports that use the same driver files. This section defines the files that must be installed and the configuration options for SYSTEM.INI, WIN.INI, and the DOS files CONFIG.SYS and AUTOEXEC.BAT.

The following example shows sample entries for the `[transport_install]` section. The entries define the transport protocol driver files that need to be installed to support the “ACME TCP Protocol” transport protocol:

```
[acme_tcp]
devdir= 1:acmetcp.dos
sysdir= 1:acmetcp.386
device= acmetcp.dos, @devdir\acmetcp.dos
transport= acmetcp.386, acmetcp.386
```

[transport_protocol]

This section specifies the information that Setup must add to PROTOCOL.INI for each of the OEM entries in the **[transport]** section. Each section name can be unique to the driver or shared by a number of network adapters that use common driver settings.

The following example shows entries for the *[transport_protocol]* section that define the transport protocol settings that need to be specified to configure the “ACME TCP Protocol” transport protocol:

```
[acme_tcp_xif]
drivername= ACMETCP$           ; driver name defined by transport protocol
param= IPADDRESS, "TCP/IP Address", chars, 12, "1 1 1 1"
param= SUBNETMASK, "TCP/IP Subnet Mask", chars, 12, "255 0 0 0"
param= SESSIONS, "NETBIOS Sessions", int, "1, 12, 1", 6, 0x02
```

[multinet]

This section defines the set of secondary networks that can be selected from setup or from the Windows Control Panel.

This example entry for the **[multinet]** section identifies the “ACME Network” as a secondary network for use with Windows for Workgroups:

```
[multinet]
acmenet="ACME Network", 0, 1, 0, mn_acme_net
```

[multinet_install]

This section provides Setup an installation section for each entry in the **[multinet]** section. Each section name can be unique to the driver or shared in common by a number of transports that use the same driver files. This section defines the files that must be installed and the configuration options for SYSTEM.INI, WIN.INI, and the DOS files CONFIG.SYS and AUTOEXEC.BAT.

This example entry for the *[multinet_install]* section defines the driver files and SYSTEM.INI entries needed to support the “ACME Network” as a secondary network for use with Windows for Workgroups:

```
[mn_acme_net]
sysdir=1:acmenet.drv, 1:acmenet.386
secondnet.drv=acmenet.drv, acmenet.drv
secondnet=acmenet.386, acmenet.386
```

Example OEMSETUP.INF

The following is an example OEMSETUP.INF file that will install the network adapter card drivers to support the "ACME Ethernet Adapter 1" network adapter card, the transport protocol drivers to support the "ACME TCP Protocol" transport protocol, and the Windows drivers to support the "ACME Network" secondary network.

```
[disks]
1=., "OEM Drivers Disk #1", disk1

[netcard]
acme=, "ACME Ethernet Adapter 1", 0, ndis, ethernet, real, acme_001, acme_001_nif

[acme_001]
devdir= 1:acme01.dos
device= acme01.dos, @devdir\acme01\acme01.dos

[acme_001_nif]
drivername= ACME$ ; driver name defined by NDIS driver
irq= INTERRUPT,, int, "2,7,1", 3
io= IOADDRESS,, hex, "0x200, 0x3f0, 0x10", 0x300
param= ACMEPARAM, "Special ACME Driver Parameter", text, "A,B,C,D", A

[transport]
acme_tcp= , "ACME TCP Protocol", 0, ndis, tcp, vxd, 0, lanabase, acme_tcp, acme_tcp_xif

[acme_tcp]
devdir= 1:acmetcp.dos
sysdir= 1:acmetcp.386
device= acmetcp.dos, @devdir\acmetcp.dos
transport= acmetcp.386, acmetcp.386

[acme_tcp_xif]
drivername= ACMETCP$ ; driver name defined by transport protocol
param= IPADDRESS, "TCP/IP Address", chars, 12, "1 1 1 1"
param= SUBNETMASK, "TCP/IP Subnet Mask", chars, 12, "255 0 0 0"
param= SESSIONS, "NetBIOS Sessions", int, "1, 12, 1", 6, 0x02

[multinet]
acmenet="ACME Network", 0, 1, 0, mn_acme_net

[mn_acme_net]
sysdir=1:acmenet.driv, 1:acmenet.386
secondnet.driv=acmenet.driv, acmenet.driv
secondnet=acmenet.386, acmenet.386
```

Modifying .INF Files for Custom Installations

You can create custom Windows for Workgroups installations for multiple systems by changing entries in the setup information files to modify the following:

- Hardware options available for installation on individual workstations.
- Setup options available for a network installation.
- Files copied to the user's Windows for Workgroups directories.
- Applications that are installed automatically during setup.

Modifying .INF Files to Install Custom Windows

If you want users at your site to run a customized version of Windows for Workgroups, you can modify SETUP.INF before setting up Windows for the workstations.

You can change default Program Manager groups, create other groups, and add applications to these groups. You should only modify SETUP.INF if you want to set up a customized Windows for Workgroups desktop. You do not need to modify this file if you want the default Windows for Workgroups configuration.

Removing Hardware Devices from Lists in Setup

During setup, users might be required to specify hardware devices from a list of supported devices. For example, when setting up a printer, the user is presented with a list of printers. The user must know what kind of printer the system is connected to and select that printer from the list. Users might also have to select their particular display adapter, pointing device, network, or other hardware devices.

You can remove from the list any devices that your users do not have by modifying SETUP.INF, CONTROL.INF, and NETWORK.INF. Be sure you do not remove a device that a user might have, because doing so might cause Setup to run incorrectly.

To remove a printer from the list of printers

1. Make a backup copy of the original .INF files and edit the copies.
2. In the [io.device] section of CONTROL.INF, delete the entry for any printers you do not want displayed in the setup lists.

To remove other hardware devices from the lists in setup

- In SETUP.INF, delete the entry for the hardware device you don't want displayed in the setup lists.

You can delete entries from the **[display]**, **[system]**, **[keyboard.types]**, **[network]**, **[machine]**, and **[pointing.device]** sections.

If you remove an entry from SETUP.INF, CONTROL.INF, or NETWORK.INF, the entry won't appear when you install a device from the Windows Control Panel or change system settings by running setup from within Windows for Workgroups.

To remove network adapter cards from the lists in setup

- In NETWORK.INF, delete the entry for the network adapter card you do not want displayed in the setup lists.

In NETWORK.INF, you can remove unwanted network adapter drivers from the **[netcard]** section. This keeps the network adapters from appearing when you attempt to install a device from the Windows Control Panel, or change system settings by running setup from within Windows for Workgroups. You can also remove the related **[netcard_install]** and **[netcard_protocol]** sections in NETWORK.INF, but this action is not necessary.

Specifying Windows Applications to Install

You can create a custom list of Windows for Workgroups applications, bitmaps, and screen savers to install with Windows for Workgroups to restrict or customize the installation for multiple systems. To do this, add or remove entries in the **[win.apps]**, **[win.dependents]**, **[win.games]**, **[win.scrs]**, and **[win.bmps]** sections of SETUP.INF.

Forcing the Network Setup Option

To be sure that users set up a shared copy of Windows for Workgroups for a network installation (the **setup /n** option), you can modify SETUP.INF. Setup copies these files to the user's WINDOWS directory before completing installation. In this case, the Setup program installs a shared copy of Windows for Workgroups on the workstation, even if the user types **setup** without the **/n** option.

To do this, make sure **netsetup= true** in the **[data]** section of SETUP.INF. The default entry is **netsetup= false**, which enables users to run any of the setup options.



Modifying .INF Files to Install Custom Applications

You can make changes in .INF files to install a custom application for multiple systems at the time Windows for Workgroups is installed. The following examples specify the steps for installing a custom application, QB.EXE, with its help file and PIF.

Copying Additional Files to a User's Directory

To copy application files to the user's WINDOWS directory during setup, you can specify additional files to copy during setup in the **[net]** section of SETUP.INF, for network setup, or specify files to copy **[win.copy]** for stand-alone systems.

The files you want to copy must be on a floppy disk or in the same directory as the Windows for Workgroups files on the network server. If the files are on a floppy disk, you must first assign a disk number in the **[disks]** section of SETUP.INF and then specify that disk number in the **[net]** or **[win.copy]** sections.

To specify an additional disk that setup can prompt for

Add an entry to the **[disks]** section of SETUP.INF and identify the disk number, path of the disk, name of the disk, and disk tag (used to determine if the disk is in the drive)—for example:

```
[disks]
(other entries)
9= ., "Quick Blibbet Application", disk7
```

The disk number refers to the Windows for Workgroups installation disk where the file is located. Make sure the values for the *disk name* and *disk tag* items in the **[disks]** entry match the actual disk label and volume name.

To copy additional files to a user's system:

1. Add one or more entries to the **[win.copy]** or **[win.net]** sections in SETUP.INF for the files that Setup is to copy, or add a reference in one of these sections for an optional **[Win_CopyFiles]** section.

You can also specify that the files be copied to a WINDOWS subdirectory. For example, to place the file in the user's C:\WINDOWS\APPDIR subdirectory, change the **[win.net]** section as follows:

```
[win.net]
(other entries)
#custom.apps, 0:appdir ; files are specified in [custom.apps]
```

2. If you specify a `[Win_CopyFiles]` section as the value in `[win.net]` or `[win.copy]`, add that section to SETUP.INF with a list of the files to copy. For each file you want to copy, include a disk number a colon, the filename, and optionally, a description.

If the files are on the network, you must specify a disk number, even though the number is ignored. If the files are on a floppy disk, use the number specified in the `[disks]` section.

For example, to install the custom application QB.EXE and its related PIF and help file, the `[custom.apps]` section should contain these entries:

```
[custom.apps]
9:QB.EXE, "Quick Blibbet", 165000, qblibbet
9:QB.PIF, "QBlibbet", 545
9:QB.HLP, "QBlibbet Help", 15000
diskspace= 180545
```

3. If you have a custom Windows-based application that uses a DLL, add an entry to the `[win.dependents]` section to install the DLL file—for example:

```
[win.dependents]
qblibbet= 9:QBLIBBET.DLL
(other entries)
```

Adding Custom Groups to Program Manager

You can edit SETUP.INF to change the content of the default Program Manager groups or to add your own custom groups. You should only modify SETUP.INF to create a custom Windows for Workgroups desktop for multiple systems. Otherwise, individual users can use the menu commands in Program Manager to customize their personal desktops.

If the modified SETUP.INF file is used both for updating and for new installations, add the group to both the `[program.groups]` and the `[new.groups]` sections.

To add an application to a Program Manager group

1. Add entries to the `[progman.groups]` and `[new.groups]` sections of SETUP.INF to install a custom Program Manager group—for example:

<code>[progman.groups]</code>	<code>[new.groups]</code>
(other entries)	(other entries)
group9= QBlibbet	group9= QBlibbet

2. Add one or more `[group#]` sections to SETUP.INF with an entry that describes the program item.

Enclose the program title in quotation marks (" ") followed by a comma (,) and the application filename and icon information. If the application does not reside in the WINDOWS directory, include its full path—for example:

```
[group9]
"Quick Blibbet", k:\qb\qblibbet.pif, moricons.dll, 24, qblibbet
```

This example specifies the icon extraction file as MORICONS.DLL, a file that comes with Windows for Workgroups 3.1.

A custom group for your application is created in Program Manager, with a program item in that group, when the other Program Manager groups are installed during setup.

If the application is on a network, make sure each user has access to the appropriate network drive. After setup, you can specify a working directory for user files to move personal files from the shared application directory. You or a user can specify a working directory by modifying the Program Item Properties in Program Manager.

Adding Custom PIF Information in APPS.INF

If your custom application uses the same executable filename as other applications listed in the `[pif]` section in APPS.INF, add entries in that file to define specific information about your application.

To add custom PIF information in APPS.INF

1. In the basic entry in `[pif]`, add an "ambiguous exe" item—for example:

```
QB.EXE= QB, "Microsoft QuickBASIC",, cwe, moricons.dll, 14,
std_QB, enha_QB, amb_QB
```

2. Add a new `[amb_*]` section to define the PIF information for your application—for example:

```
[amb_QB]
;"Microsoft QuickBASIC" and "Quick Blibbet" both use QB.EXE
QB.EXE= QBlibbet, "Quick Blibbet",, cwe, moricons.dll, 24, std_QB1, enha_QB1
```

3. Add `[std_*]` and `[enha_*]` sections to define custom PIF information—for example:

```
[std_QB1]
minconvmem= 192
videomode= gra
```

```
[enha_QB1]
convmem= 380, 640
execflags= bgd
```

Windows for Workgroups Initialization Files

This chapter contains information about the settings that are present in the Windows for Workgroups initialization files. This information is provided for reference and describes the entries that may be present in some of the initialization files used by components of Windows for Workgroups: WIN.INI, SYSTEM.INI, PROTOCOL.INI, PROGMAN.INI, CONTROL.INI, WINFILE.INI, MSMAIL.INI, SHARED.INI, and SCHDPLUS.INI. This chapter describes the sections of each initialization file and each of the related entries. The chapter also specifies the default value used for an entry (the default value is used if the entry is not present in the initialization file), and the means for changing an entry present in an initialization file.

Related Information

- *Windows for Workgroups Resource Kit: Chapter 5, “Windows for Workgroups Setup Information Files”; Chapter 6, “Windows for Workgroups Files”; Chapter 14, “Troubleshooting Windows for Workgroups.”*

Contents of This Chapter

About the Initialization Files	6-2
Format of the .INI Files.....	6-2
Changing Entries in .INI Files	6-3
Editing .INI Source Files	6-5
WIN.INI: Windows for Workgroups Initialization File	6-6
SYSTEM.INI: System Initialization File.....	6-29
PROTOCOL.INI: Network Protocol Initialization File.....	6-73
PROGMAN.INI: Program Manager Initialization File.....	6-76
CONTROL.INI: Control Panel Initialization File.....	6-79
WINFILE.INI: File Manager Initialization File	6-80
MSMAIL.INI: Microsoft Mail Initialization File	6-82
SHARED.INI: Shared Mail Settings Initialization File	6-94
SCHDPLUS.INI: Schedule+ Initialization File	6-95

About the Initialization Files

The Windows for Workgroups initialization files contain information that defines your Windows for Workgroups environment. Microsoft Windows for Workgroups and Windows-based applications use the information in these files to configure themselves according to your needs and preferences.

The standard Windows for Workgroups initialization files are the WIN.INI, SYSTEM.INI, PROTOCOL.INI, PROGMAN.INI, CONTROL.INI, WINFILE.INI, MSMAIL.INI, SHARED.INI, and SCHDPLUS.INI files. Other Windows-based applications may also add .INI files in the WINDOWS directory.

.INI File	File Contents
WIN.INI	Entries that you can use to set up the Windows for Workgroups environment according to your preferences.
SYSTEM.INI	Entries that you can use to set up Windows for Workgroups to meet your system's hardware needs.
PROTOCOL.INI	Entries that define the network protocol and media-access control drivers.
PROGMAN.INI	Entries that define the content of program groups.
CONTROL.INI	Entries that describe the color schemes and patterns used in Windows for Workgroups, and the settings for printers and installable drivers.
WINFILE.INI	Entries that define the appearance and behavior of items in File Manager.
MSMAIL.INI	Entries that define the appearance and behavior of Mail.
SHARED.INI	Entries that you can use to share Mail custom commands and custom messages with other members of your workgroup.
SCHDPLUS.INI	Entries that define the appearance and behavior of Schedule+.

Important Errors made by editing initialization files can lead to undesirable results when you run Windows for Workgroups. Before changing any entry, make a backup copy of the file. Read "Changing Entries in .INI Files" later in this chapter for guidelines.

Format of the .INI Files

The Windows for Workgroups setup information (.INI) files are text files that contain one or more sections. The format for each section is:

```
[section]
keyname= value           ; comment
```

Value	Definition
<i>[section]</i>	The name (header) of a section. The enclosing brackets ([]) are required, and the left bracket must be in the leftmost column on the screen.
<i>keyname</i>	The name of an entry, which can consist of any combination of letters and digits. For many entries, the <i>keyname</i> must be followed immediately by an equal sign.
<i>value</i>	The value of each entry which can be an integer, a string, or a quoted string, depending on the entry.
<i>; comment</i>	Some entries include brief comments below the header or on the same line as an entry. You can include comments anywhere in an .INI file by prefacing the comment with a semicolon (;).

In this chapter, the sections that appear in bold contain the actual words that are used in the INI file. For example: **[windows]** section. When a section is referenced generically, the section name is shown in italics, in bold brackets. For example, PROTOCOL.INI contains the *[netcard_protocol]* section that actually includes separate sections for each network adapter card name, such as **[MSSEE16]** for the Intel EtherExpress 16 or 16TP adapter card.

Some variable entries must be substituted for specific values in the entry. These entries are shown in italics in this format:

keyname= profile, description, filesize

Case (capital or lowercase letters) does not matter for values, unless specified for a particular item. Some items must be enclosed in double quotation marks (" "). For example: **caption= "Windows Setup"**.

The actual words of an entry are shown in bold, such as "**Beep= Yes | No**". The values that can be substituted in the entry are shown in italics. When the value can be one of several choices, the choices are separated with a pipe character (|). For example:

Beep= Yes | No

For many entries, the *value* is shown as *Boolean*. To enable an entry that requires a Boolean value, you can enter **True**, **Yes**, **On**, or **1**. To disable such an entry, you can enter **False**, **No**, **Off**, or **0**. Case doesn't matter.

Any entry listed here that does not appear in your .INI file has the default value.

Changing Entries in .INI Files

Windows for Workgroups creates the initialization files during installation and assigns default values. Some entries are added or changed when you install or

configure a Windows application. You can edit these entries to change the appearance or performance of Windows for Workgroups.

You can use the following options to change the entries in an .INI file:

- Use Control Panel, Program Manager, File Manager, Mail or Schedule+ to change entries with the menu commands and dialog box options.
- Run Windows Setup again to change system settings, the keyboard or mouse configuration, or network options, and to add or remove printers and fonts.
- Choose a command such as Printer Setup from the File menu in Print Manager and specify new options.
- Use a text editor such as Notepad to edit the file directly.

Many of the entries in this section recommend the most appropriate method for changing the entry.

To change .INI file entries with a text editor:

Important Always create a back up copy of the .INI file before you open it, and use care when making changes. Incorrect changes can lead to unexpected results when you run Windows for Workgroups. Edit the entries for this file only when necessary. Do not use a formatting editor, such as a word processor in document mode. Some editors can damage characters with ANSI values of greater than 127. We recommend that you use Control Panel or Setup when possible to make changes. If you must edit the file, use a text editor such as Notepad, System Editor, or Edit (provided with MS-DOS 5.0 and greater).

1. **Important** Create a backup copy of the .INI file and save the original version in unedited form.
2. Open the copy of the .INI file with a text editor such as Notepad.
3. Edit the specific entries and save the file in ASCII (text only) format.
4. Restart Windows for Workgroups.

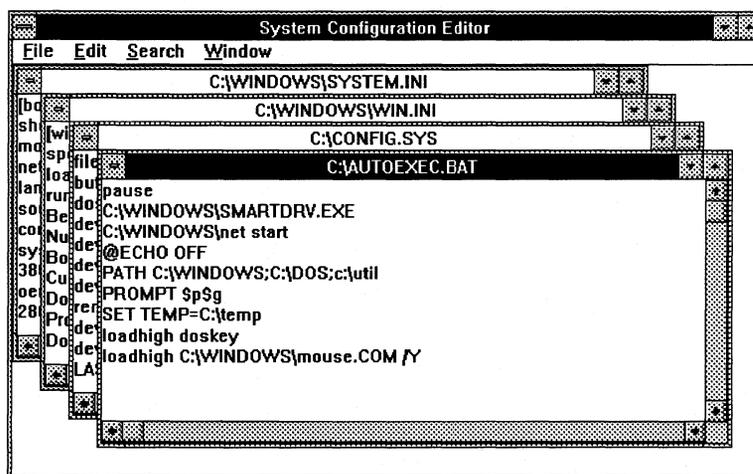
To edit system files with System Editor:

You can use the System Editor utility in Windows for Workgroups to edit SYSTEM.INI, WIN.INI, CONFIG.SYS, and AUTOEXEC.BAT at the same time if needed. The System Editor is installed in your WINDOWS directory by Windows Setup.

1. **Important** Create a backup copy of the .INI file you want to edit.
2. Choose Run from the File Menu in Program Manager. Type `sysedit` and press ENTER.
3. Click the window in System Editor that contains the file you want to edit.

Figure 6.1

The System Editor in
Windows for
Workgroups 3.1



4. Edit the file, using the same text editing techniques as in Notepad.
5. Save the file, and choose Exit from the File menu to close System Editor.
6. Restart Windows for Workgroups to enable the changes you made to take effect. If you edit CONFIG.SYS or AUTOEXEC.BAT, you must reboot your system for the changes to take effect.

Editing .INI Source Files

If you copy the Windows for Workgroups files to a network server using the `setup /a` option (to install a shared copy of Windows for Workgroups), Setup uses WIN.SRC, SYSTEM.SRC, and CONTROL.SRC to build WIN.INI, SYSTEM.INI, and CONTROL.INI. To create custom initialization files for multiple installations, you can change the .SRC files.

To edit the .SRC files, first save a backup copy of the original file, then make editing changes in the file, following the guidelines in this chapter. Save the file with an .SRC filename extension in the WINDOWS directory on the network server. And, of course, test the new .INI file on a single system before installing Windows for Workgroups on multiple systems.

WIN.INI: Windows for Workgroups Initialization File

The WIN.INI file contains settings that you can use to customize the Windows for Workgroups environment to meet your needs and preferences. Each WIN.INI file is configured with a number of sections and entries, depending on your system's hardware and software requirements. For example, many Windows-based applications add entries in WIN.INI to define user preferences and other items.

These basic sections can appear in WIN.INI after you first install Windows for Workgroups or use applications provided with Windows for Workgroups. The WIN.INI sections discussed in this chapter appear here in alphabetical order for easy reference, the sections may not appear in this sequence in the initialization file.

Section	Purpose
[colors]	Defines the colors for the Windows for Workgroups display.
[desktop]	Controls the appearance of the desktop and the position of windows and icons.
[devices]	Lists active output devices that provide compatibility with earlier versions of Windows for Workgroups applications.
[embedding]	Lists the server objects used in Object Linking and Embedding (OLE).
[extensions]	Associates specified types of files with corresponding applications.
[fonts]	Describes the screen font files that are loaded by Windows for Workgroups.
[FontSubstitutes]	Lists pairs of fonts that are recognized by Windows for Workgroups as interchangeable.
[Hearts]	Used to define the appearance and behavior of the Hearts application.
[intl]	Describes how to display items for countries other than the United States.
[mci extensions]	Associates specified types of files with Media Control Interface devices.
[network]	Describes network settings and previous network connections.
[ports]	Lists all available output ports.
[programs]	Lists additional paths that Windows for Workgroups searches to find a program file when you try to open an associated data file.
[PrinterPorts]	Lists active and inactive output devices to be accessed by Windows for Workgroups.
[WindowsHelp]	Lists settings used to specify the default size, placement, and text colors of the help window and dialog boxes.
[sounds]	Lists the sound files that are assigned to each system event.
[spooler]	This section is used by Print Manager.
[Mail]	Contains global information for the Mail program provided with Windows for Workgroups.
[MRU_Files]	Maintains the 12 most recently used (MRU) file share connections for this workstation. Utilized by the Connect Network Drive dialog box.





[MRU_Printers]	Maintains the 12 most recently used (MRU) printer share connections for this workstation. Utilized by the Connect Network Printer dialog box.
[TrueType]	Describes options for using and displaying TrueType fonts.
[Windows]	Affects several elements of the Windows for Workgroups environment.

More sections than listed in the previous table can be inserted in WIN.INI by other Windows-based applications, although some applications set up their own private .INI files. For information about those sections, refer to the application's product or user documentation for specific information.

[colors]

The **[colors]** section defines the colors for components of the Windows for Workgroups display. To change these entries, choose the Color icon from Control Panel. This section can contain entries in this format:

component= *red-value green-value blue-value*

Value	Definition
<i>component</i>	Any one of these keynames: ActiveBorder (the border of the active window) ActiveTitle (the title bar of the active window) AppWorkspace (the application workspace for Windows-based applications) Background (refers to the desktop) ButtonFace ButtonShadow ButtonText GrayText (text that is dimmed, as in an unavailable command name) Highlight (the background of highlighted text) HighlightText (the highlighted text itself) InactiveBorder (the border of an inactive window) InactiveTitle (the title bar of an inactive window) InactiveTitleText (text in an inactive title bar) Menu (the background of menus) MenuText Scrollbar TitleText (the text in the title bar) Window (the workspace in a window) WindowFrame WindowText
<i>red-value</i>	Integers that specify the relative intensities of the respective colors. These values can range from 0 (minimum intensity) to 255 (maximum intensity).
<i>blue-value</i>	
<i>green-value</i>	

[desktop]

The **[desktop]** section contains entries that control the appearance of the screen background (desktop) and the position of windows and icons on the screen. This section can contain the following entries:

GridGranularity= *number*

Specifies the size of the grid that Windows for Workgroups uses to position windows on the screen. The possible range is 0 through 49, in units of 8 pixels. The default is 0. To make changes to this entry, choose the Desktop icon from Control Panel.

IconSpacing= *pixels*

Specifies the number of pixels that appear horizontally between icons. A larger number increases the space between icons. The default is 77. To make changes to this entry, choose the Desktop icon from Control Panel.

IconTitleFaceName= *fontname*

Specifies the font used to display icon titles. Change this value if the icon title is difficult to read. The default is **MS Sans Serif**. To make changes to this entry, edit WIN.INI.

IconTitleSize= *number*

Specifies the size of the font used to display icon titles. Change this value if the icon title is difficult to read. The default is 8. To make changes to this entry, edit WIN.INI.

IconTitleWrap= 0 | 1

Specifies whether to wrap icon titles. A value of 1 allows icon title wrapping and increases icon vertical spacing by three lines; 0 turns off icon title wrapping. The default is 1. To make changes to this entry, choose the Desktop icon from Control Panel.

IconVerticalSpacing= *pixels*

Specifies the number of pixels that appears vertically between icons. Increasing this number increases the amount of vertical space between icons. Windows for Workgroups determines the default based on the icon title font and the display you are using. To make changes to this entry, edit WIN.INI.

Pattern= *b1 b2 b3 b4 b5 b6 b7 b8*

Specifies a pattern for the screen background. The 8 numeric values define a bitmap 8 pixels wide and 8 pixels high. Each decimal value represents a byte, and each byte represents a row of 8 pixels, where 0 sets the corresponding pixel to the background color and 1 sets the corresponding pixel to the foreground color (specified by the **Background=** and **WindowText=** entries in the **[colors]** section, respectively). For example, if you set the *b1* value to the decimal value 175, the top row of pixels in the bitmap appears as the binary equivalent (10101111). The default is "(None)" (this string appears when no pattern is specified). To make changes to this entry, choose the Desktop icon from Control Panel.

TileWallpaper= *0 | 1*

If this entry is 1, the desktop wallpaper is tiled across the screen. If this entry is 0, the wallpaper is centered. The default is 0. To make changes to this entry, choose the Desktop icon from Control Panel.

Wallpaper= *bitmap-filename*

Supplies the filename for the bitmap on the screen background. Include the path if the file is not in the WINDOWS directory or SYSTEM subdirectory. The default is "(None)" (this string appears when no pattern is specified). To make changes to this entry, choose the Desktop icon from Control Panel.

[devices]

The **[devices]** section lists the active printers and is necessary only for compatibility with Windows version 2.x applications. The entries in this section are identical to those in the **[PrinterPorts]** section, without the timeout values. The **[devices]** section can contain one or more entries in this format:

device= driver, port [, other ports . . .]

The values specified are identical to the first three items in the **[PrinterPorts]** section. To make changes to this entry, choose the Printers icon from Control Panel.

Important When changing an entry in the **[devices]** section, be sure the entry does not conflict with a corresponding entry in the **[PrinterPorts]** section. Problems can occur if a connected device does not appear in the **[PrinterPorts]** section.

[embedding]

The **[embedding]** section lists the Object Linking and Embedding (OLE) objects, their description, the program that is used to create them, and their file format. This information also appears in the Registration Info Editor. We recommend that you use the Registration Info Editor to register applications or edit application information for OLE. The **[embedding]** section is included in WIN.INI only to maintain compatibility with Windows 3.0 and Windows 3.0 applications.

The **[embedding]** section can contain one or more entries in the following format to specify the objects used as servers for OLE:

object= description, description, program file, format

Value	Definition
<i>object</i>	Specifies the name of the server object.
<i>description, description</i>	Provides a description of the server object. The second description appears in the list of registered file types in the Registration Info Editor.
<i>program file</i>	Identifies the path and name of the executable file for the application that is used to create the object.
<i>format</i>	Identifies the format of the file. The value for this entry is most often picture , which is the Windows Metafile format. To make changes to this entry, use the Registration Info Editor.

[extensions]

The **[extensions]** section contains entries that identify document files with corresponding command lines, so that opening a document file automatically starts the application. The format for each entry is:

extension= command-line

Value	Definition
<i>extension</i>	A filename extension of one to three characters.
<i>command-line</i>	An ordinary MS-DOS command line. The command line always begins with an application filename (including the .EXE extension). This can be followed by any command parameters the application needs, usually including the document filename. The caret (^) character can be used when the original document filename is needed in the command line. The caret is replaced by the document filename without any extension.

The most typical **[extensions]** entries simply list the application name and the document name. For example:

TXT= NOTEPAD.EXE ^.TXT

If you run a file called README.TXT after this association is made, the Notepad program starts and automatically opens the file, README.TXT.

You can enter only one line per extension (.TXT, .INI, and others), but more than one line for any program (for example, separate lines to indicate that Write is to be used for .TXT, .INI, and .WRI files). To change an entry in **[extensions]**, choose Associate from the File menu in File Manager. Windows for Workgroups also updates this entry from any changes made in the Registration Info Editor. If an entry in **[extensions]** duplicates information in the registration database, File Manager uses the information from the database.



[Hearts]

The **[Hearts]** section contains entries used by the MSHEARTS.EXE application to identify the names the computer uses when playing additional hands in Hearts. To change the predefined names for the hands the computer plays, add or change the following entries in the WIN.INI file directly:

p1name= *playername*

p2name= *playername*

p3name= *playername*

The *playername* value can be up to 14 characters in length.

[intl]

The **[intl]** section describes how to display dates, times, currency amounts, and other items for countries other than the United States. The defaults reflect U.S. options. To change any entry in this section, choose the International icon from Control Panel. For details about the specific values for individual entries, see the WININI.WRI file in your WINDOWS directory.

If the keyword in an entry in the **[intl]** section begins with **i**, the value is an integer. If the keyword begins with **s**, the value is a string.

iCountry= *country*

Specifies the country code. This number matches the country's international telephone code, with the exception of Canada, which is 2. The default is 1.

iCurrDigits= *number*

Specifies the number of digits to put after the decimal separator in currency. The default is 2.

iCurrency= *number*

Specifies a currency format, where 0 = \$2, 1 = 2\$, 2 = \$ 2, and 3 = 2. The default is 0. The actual currency symbol is specified by the **sCurrency=** value.

iDate= number

Specifies a numerical date format for compatibility with Windows 2.x, where **0** = 12/31/90, **1** = 31/12/90, and **2** = 90/12/31. The default is **0**. The actual date divider is specified by the **sShortDate=** value.

iDigits= number

Specifies the number of digits to display after the decimal separator in numbers. The default is **2**.

iLZero= 0 | 1

Specifies whether to put leading zeros in decimal numbers, where **0** = .7 and **1** = 0.7. The default is **0**. The actual decimal separator is specified by the **sDecimal=** value.

iMeasure= 0 | 1

Specifies the measurement system as metric or English, where **0** = metric and **1** = English. The default is **1**.

iNegCurr= number

Specifies a format to use for negative values for currency, where:

0 = (\$1)	3 = \$1-	6 = 1-\$	9 = -\$ 1
1 = -\$1	4 = (1\$)	7 = 1\$-	10 = \$ 1-
2 = \$-1	5 = -1\$	8 = -1 \$	

The default is **0**. The actual currency symbol is specified by the **sCurrency=** value.

iTime= number

Specifies whether to format time using a 12-hour or 24-hour clock, where **0** = 1:00 (12-hour clock) and **1** = 13:00 (24-hour clock). The default is **0**. The actual time separator is specified by the **sTime=** value.

iTLZero= number

Specifies whether to put leading zeros in time, where **0** = 9:15 and **1** = 09:15. The default is **0**. The actual time separator is specified by the **sTime=** value.

s1159= string

Specifies the string that follows times before noon in the 12-hour time format. The default is **AM**.

s2359= *string*

Specifies the string that follows times after noon in the 12-hour format, or that follows all times in the 24-hour format. The default is **PM**.

sCountry= *string*

Specifies the name of the country whose standard value you want to use. The default is **United States**.

sCurrency= *string*

Specifies the currency symbol you want to use. The default is **\$**.

sDecimal= *string*

Specifies the punctuation used to separate the fractional part of a decimal number from the whole number part. The default is **.** (a period).

sLanguage= *string*

Specifies the language you want to work in. Windows-based applications that provide language specific tasks, such as sorting or spell checking, use this entry. The default is **enu**. Values are:

dan = Danish	frc = French Canadian
deu = German	isl = Icelandic
eng = International English	ita = Italian
enu = U.S. English	nld = Dutch
esn = Modern Spanish	nor = Norwegian
esp = Castilian Spanish	ptg = Portuguese
fin = Finnish	sve = Swedish
fra = French	

sLis= *string*

Specifies the character used to separate items in a list. In U.S. English, the most common separator is a comma. The default is a comma (**,**).

sLongDate= *format*

Specifies your choices for the long date formats including abbreviations for the words and separators. Control Panel accepts only certain *format* combinations. You can use Control Panel to change these entries. The default is **dddd, MMMM dd, yyyy** (Friday, June 1, 1993). The values are:

d = Day (1–31)	MM = Month (01–12)
dd = Day (01–31)	MMM = Month (Jan–Dec)
ddd = Day (Mon–Sun)	MMMM = Month (January–December)
dddd = Day (Monday–Sunday)	yy = Year (00–99)
M = Month (1–12)	yyyy = Year (1900–2040)

sShortDate= *format*

Specifies a choice for the short date format including abbreviations for the words and separators, according to the list described for **sLongDate=**. Control Panel accepts only certain *format* combinations. Use Control Panel to change this setting. The default is **M/d/yy** (that is, 6/1/90).

sThousand= *string*

Specifies the symbol used to separate thousands. For example, if the value is a comma (,), the number appears as 3,000. The default is a comma (,).

sTime= *string*

Specifies the character used to separate the hours, minutes, and seconds in time. For example, if the value is a colon (:), the time appears as 15:29:31. The default is : (a colon).

[fonts]

The **[fonts]** section describes the screen font files that Windows for Workgroups loads at startup. To make changes to this entry, choose the Fonts icon from Control Panel. This section can contain one or more entries in this format:

font-name= font-filename

Value	Definition
<i>font-name</i>	The descriptive name of a font.
<i>font-filename</i>	The filename of a file that contains font resources.

Note Adding entries to the **[fonts]** section does not install a font. You must choose the Fonts icon in Control Panel to install new fonts.

[FontSubstitutes]

The **[FontSubstitutes]** section describes fonts recognized by Windows as identical to another typeface. This is useful if you want to work on documents that include screen fonts that are not supported by Windows for Workgroups 3.1, but were supported by Windows 3.0.

The **[FontSubstitutes]** section contains one or more occurrences of the following entry:

font-name= *font-name*

Specifies the font that Windows for Workgroups uses in place of another font, only if that font is not installed on your system. For example, if you are viewing a document that is formatted in Helvetica, but Helvetica is not installed on your system, Windows for Workgroups uses Arial to display the document. To make changes to this entry, edit WIN.INI.

The default entries for this section are:

```
[FontSubstitutes]
Helv= MS Sans Serif
Tms Rmn= MS Serif
Times= Times New Roman
Helvetica= Arial
```



[Mail]

Contains information global for the Mail program provided with Windows for Workgroups. The contents of this section is read by some mail-enabled applications to detect Mail functionality required for their operation.

MAPI= *0* | *1*

This entry is used to inform other mail-enabled applications that the MAPI support functions are present on this workstation, and is written by the Mail program provided with Windows for Workgroups when Mail is run for the first time on this workstation. An entry of **1** identifies that the MAPI functions are present on this workstation. An entry of **0** is used to inform other applications that the MAPI support functions are not present. The Mail program will create this entry with a default value of **1** the first time it is run. (You should not need to change this entry.)

[mci extensions]

The **[mci extensions]** section contains entries that associate different types of media files with the Media Control Interface (MCI) drivers. When a media file is selected, Windows for Workgroups plays the file using the associated driver. This section can contain one or more occurrences of the following entry:

extension= mcidevice-type

To make changes to this entry, edit WIN.INI.



[MRU_Files]

Maintains the 12 most recently used (MRU) file share connections for this workstation. The values for these entries are displayed in the Path combo-box of the Connect Network Drive dialog box in File Manager. Refer to the “File Manager” section in Chapter 10, “New and Updated Accessories,” of this manual for more information.



[MRU_Printers]

Maintains the 12 most recently used (MRU) printer share connections for this workstation. The values for these entries are displayed in the Path combo-box of the Connect Network Drive dialog box in Print Manager. Refer to the “Print Manager” section in Chapter 10, “New and Updated Accessories,” of this manual for more information.

[network]

The **[network]** section describes network settings used by secondary Windows network drivers and are not used by the Windows for Workgroups WFWNET.DRV driver. The **[network]** section can contain the following entries:

drive= network-server-and-share

Shows the network connections that are restored by a secondary network driver each time Windows for Workgroups is started. To make changes to this entry, choose the Network Connections command from the Disk menu in File Manager.

port= *network-printer-path*

Defines the path for a network printer and the port the printer is assigned to. This connection is restored by a secondary network driver each time Windows for Workgroups is started. To make changes to this entry, choose the Printers option from Control Panel, or the Network Connections command from the Options menu in Print Manager.

[ports]

This section lists the available communications and printer ports, defines defaults, and lists files to which printer output can be sent. This is similar to the **mode** command for MS-DOS.

This section can contain one or more entries in this format:

portname:= baud-rate, parity, word-length, stop-bits [, p]

Value	Definition
<i>portname</i>	The name of an output port as it is recognized by MS-DOS. This can be: COMx:= for a communications port, where <i>x</i> represents the port number. LPTx:= for a parallel port, where <i>x</i> represents the port number (1-9) EPT:= for a specific IBM printer FILE:= for a filename LPTx.DOS= for a parallel port, where <i>x</i> represents the port number (1-9). Connecting to LPTx.DOS can bypass Windows handling or print output. Choose this option if you have trouble printing.
<i>baud-rate</i>	Specifies a COM port's baud rate.
<i>parity</i>	Specifies the parity setting for a COM port.
<i>word-length</i>	Specifies the length of a word (in bits) for a COM port.
<i>stop-bits</i>	Specifies the number of stop bits to be used for a COM port.
[, p]	Specifies that hardware handshaking is in effect.

Values for LPT, EPT, FILE, and LPTx.DOS ports do not take any parameters.

Specify a filename for the *portname* value to direct output to that file. A filename, unlike the other port names, must not be followed by a colon (:). If you specify a value for **FILE:=**, Windows for Workgroups prompts you to specify an output file each time you print.

To change COM port settings, choose the Ports icon from Control Panel. To change or add other ports, edit WIN.INI.

Up to nine LPT port entries can be defined as LPT1 - LPT9. This allows up to nine connections to be specified for network printers.

The “**LPTx.DOS= path**” settings are used (instead of the corresponding LPT ports) to specify the path of the printer you are using, and use DOS file I/O calls to send output to the printer device. Use these settings to bypass Windows for Workgroups special handling for parallel port output. The default is none (blank). To make changes to this entry, edit WIN.INI.

The default entries for this section are:

```
[ports]
LPT1:=
LPT2:=
LPT3:=
COM1:=9600,n,8,1,x
COM2:=9600,n,8,1,x
COM3:=9600,n,8,1,x
COM4:=9600,n,8,1,x
EPT:=
FILE:=
LPT1.DOS=
LPT2.DOS=
```

[PrinterPorts]

The **[PrinterPorts]** section lists the active and inactive printers that can be accessed by Windows for Workgroups device drivers. This section also specifies the communications port or ports to which the devices are connected and their corresponding timeout values. This section can contain one or more entries in this format:

device= driver, port, DeviceTimeout, RetryTimeout [, other ports . . .]

Value	Definition
<i>device</i>	The name of the device.
<i>driver</i>	The driver filename, with path if necessary.
<i>port</i>	Specifies the port the device is connected to. If a device is not currently connected, the value is the NullPort= entry of the [Windows] section. There is a set of <i>port</i> , <i>DeviceTimeout</i> , and <i>RetryTimeout</i> values for each port that has a printer of the specified type.
<i>DeviceTimeout</i>	Determines the number of seconds that Print Manager waits for the device to be switched on.
<i>RetryTimeout</i>	Determines the number of seconds before attempting transmission retries. This value might need to be increased for network printers.

If the timeout values are not specified for a particular port, Windows for Workgroups uses the defaults listed in the **[Windows]** section for that port.

To make changes to this entry, choose the Printers icon from Control Panel or choose Printer Setup from the Options menu in Print Manager, then choose Connect.

Important When changing an entry in this section, be sure the entry does not conflict with a corresponding entry in the **[devices]** section. Problems can occur if a connected device does not appear in the **[PrinterPorts]** section.

[programs]

The **[programs]** section identifies the paths that Windows for Workgroups searches to start an application, in addition to those in the AUTOEXEC.BAT file. If you try to open a data file associated with an application, and Windows for Workgroups cannot find the program file needed to start the application, you are prompted to supply the path for the program file. The path you specify is automatically added to the **[programs]** section of WIN.INI. Windows for Workgroups searches the path for the program file the next time you try to start an application by opening an associated data file.

The **[programs]** section can contain one or more occurrences of the following entries to specify the drive, directory, and executable program file needed to start an application:

program file= *drive*:\ *directory*\ *program file*

Value	Definition
<i>program file</i>	Specifies the name of the program file that Windows for Workgroups was searching for when you tried to open the associated data file.
<i>drive</i>	Specifies the drive on which the program file is located.
<i>directory</i>	Specifies the directory in which the program file is located. The second <i>program file</i> keyname specifies the name of the executable program file, located in the specified drive and directory, that is needed to start the associated application.

To make changes to entries in this section, complete the dialog box in File Manager that prompts you for the path if you try to open an application that it cannot find.

[sounds]

The **[sounds]** section lists the system events that support sound, and the sound files that are assigned to each event. The **[sounds]** section can contain one or more occurrences of the following entry:

system event= *filename*, *description*

Specifies the sound file assigned to, and a description of, a *system event*. The *filename* keyname is the name of the sound file. The *description* keyname is a text string that describes the system or application event. To make changes to the filenames specified for the entries in this section, choose the Sound icon in Control Panel.

The entries that are valid for the components provided with Windows for Workgroups include (the default values are shown as created by the Windows for Workgroups Setup program):

```
SystemDefault= ding.wav,Default Beep
SystemExclamation= ding.wav,Exclamation
SystemStart= chimes.wav,Windows Start
SystemExit= chimes.wav,Windows Exit
SystemHand= ding.wav,Critical Stop
SystemQuestion= ding.wav,Question
SystemAsterisk= ding.wav,Asterisk
```



```
RingIn= ringin.wav,Incoming Call
RingOut= ringout.wav,Outgoing Call
```

The events listed above that are prefixed with the word “System” are system events recognized by Windows for Workgroups. The events listed above that are prefixed with the word “Ring” are events recognized by the Chat application provided with Windows for Workgroups.

The Mail program provided with Windows for Workgroups recognizes the following additional entry for the **[sounds]** section to identify a sound file to be used to provide notification when new mail arrives (this entry is not placed in the WIN.INI file by the Mail program and must be manually entered):



```
MailBeep= filename,New Mail Notification
```

[spooler]

The **[spooler]** section is used by Print Manager. Many entries that may be present in this section are defined privately by Print Manager and are not defined in this manual—these entries should not be changed.

**NoShareCommands= 0 | 1**

This entry is used to inform Print Manager whether the printer sharing menu commands should be displayed on the Printer menu. An entry of **1** (non-zero) tells Print Manager to not display the Share Printer As and the Stop Sharing Printer menu items on the Printer menu (and the related buttons on the toolbar)—this inhibits the user from being able to share or stop sharing printers connected to the local workstation. An entry of **0** tells Print Manager to display the printer sharing menu commands. The default is **0**.

[TrueType]

The **[TrueType]** section describes options that affect the use and display of TrueType fonts in your Windows-based applications. The **[TrueType]** section can contain the following entries:

OutlineThreshold= number-of-pels-per-em

Specifies the number of pels-per-em at which Windows for Workgroups renders TrueType fonts as outline fonts instead of as bitmap fonts. Using bitmap fonts is faster, but requires more memory. If your system is low on memory, decrease this value. Do not specify a value over **300**. If you do, you might encounter problems with TrueType fonts. The default is **256**. To make changes to this entry, edit WIN.INI.

TTEnable= 0 | 1

Controls whether TrueType fonts are available. Setting this value to **1** makes TrueType fonts available in your Windows for Workgroups applications. Setting this value to **0** turns off TrueType fonts so they are unavailable in applications. The default is **1**. To make changes to this entry, choose the Fonts icon from Control Panel.

TTIfCollisions= 0 | 1

Specifies whether to use TrueType fonts in place of other fonts if both types of fonts are installed on your system and have the same font name. For example, Windows for Workgroups provides both a bitmap and a TrueType version of the Symbol font. If this entry is set to **1**, Windows uses the TrueType version. If the default value of **0** is used, Windows does not use the TrueType version. To make changes to this entry, edit WIN.INI.

TTOnly= 0 | 1

Specifies whether to make only TrueType fonts available in Windows-based applications. If this value is set to **1**, only TrueType fonts are available. If this value is set to **0**, all fonts installed on your system are available. The default is **0**. To make changes to this entry, choose the Fonts icon from Control Panel.

[Windows]

Contains entries that affect these parts of the Windows for Workgroups environment:

- Applications that start when you start Windows for Workgroups
- Warning beep
- Printing
- Window border width
- Keyboard speed
- Mouse settings
- Definition of files as documents or programs

The [Windows] section can contain the following entries:

Beep= Yes | No

Specifies whether Windows for Workgroups sounds a warning beep when you attempt to do something that is not allowed. The default is **Yes**. To make changes to this entry, choose the Sound icon from Control Panel. For entries that control multimedia sound, see the description of [sound] section later in this chapter.

BorderWidth= number

Sets the width of the borders around all windows that have sizeable borders. The possible range is **1** (narrowest) to **49** (widest). The default is **3**. To make changes to this entry, choose the Desktop icon from Control Panel.

CoolSwitch= 0 | 1

Turns fast task switching on or off. The default is **1**. To make changes to this entry, choose the Desktop icon from Control Panel and check or clear the Fast "ALT+TAB" Switching option.

CursorBlinkRate= milliseconds

Indicates how much time elapses between each blink of the selection cursor. The default is **530**. To make changes to this entry, choose the Desktop icon from Control Panel.

DefaultQueueSize= number

Specifies the number of messages an application's message queue can hold. The default is **8**. To make changes to this entry, edit WIN.INI. (You should not need to change this entry.)

Device= *output-device-name, device-driver, port-connection*

Defines the default printer. The *output-device-name* can be any device name listed in the **[devices]** section. An explicit port and driver must be assigned to the device. The *device-driver* is the filename (without the extension) of the device driver file. The *port-connection* is any port in the **[ports]** section. The default is none (blank). To make changes to this entry, choose the Printers icon from Control Panel or choose Printer Setup from the Option menu in Print Manager.

DeviceNotSelectedTimeout= *seconds*

Specifies the default time Windows for Workgroups waits for a device to be switched on. An entry is saved for each printer listed in the **[PrinterPorts]** section. If the device is not switched on during this time, Windows for Workgroups won't print to the device. For some devices, Windows for Workgroups immediately posts an error message if the device is not already switched on. The default is **15**. To make changes to this entry, edit WIN.INI. This entry only defines the system default. To change the value for a particular printer, choose the Printers icon from Control Panel, which changes the corresponding value only in the **[PrinterPorts]** section.

Documents= *extensions*

Defines files to be considered "documents" by Windows for Workgroups. Use this entry to define only the document file extensions that are not listed in the **[extensions]** section. (Extensions listed in the **[extensions]** section are automatically considered documents.) The extensions listed in this entry are not associated with any application. Separate the filename extensions with a space and do not include the preceding periods. The default is none (blank). To make changes to this entry, edit WIN.INI.

DosPrint= *Yes | No*

Specifies whether Windows for Workgroups is to use MS-DOS interrupts during printing. When this entry is **Yes**, MS-DOS interrupts are used. If the value is **No**, printing output is sent directly to the port that the printer is assigned to. The default is **No**. To make changes to this entry, choose the Printers icon from Control Panel or the Printer Setup command from the Print Manager Options menu, then clear the Fast Printing Direct To Port option in the Printers Connect dialog box.

DoubleClickHeight= *pixels*

DoubleClickWidth= *pixels*

These entries specify the height and width (in pixels) that the mouse pointer can move between clicks in a double-click. If the mouse pointer is moved further up or down (or left or right), the two clicks are treated as separate single clicks. For example, if **DoubleClickHeight**= 10, the mouse pointer cannot move vertically more than 5 pixels between clicks for the two clicks to be recognized as a double-click. An odd number is rounded to the nearest even number. This setting is helpful if you are using a pen-tablet pointing device. The default is 4 for both entries. To change these entries, edit WIN.INI.

DoubleClickSpeed= *milliseconds*

Sets the maximum time between clicks of the mouse button the system permits for one double-click. The lower the value, the less time you have to click twice to double-click. The default is 452. To make changes to this entry, choose the Mouse icon from Control Panel.

KeyboardDelay= *milliseconds*

Establishes how much time elapses after you hold down a key before the key starts to repeat. The default is 2. To make changes to this entry, choose the Keyboard icon from Control Panel.

KeyboardSpeed= *milliseconds*

Sets how much time elapses between repetitions of a character on the display when you hold down a keyboard key. This entry does not work for all keyboards. The default is 31. To make changes to this entry, choose the Keyboard icon from Control Panel. If you delete this entry from WIN.INI, Windows Setup does not reset the speed. If you choose the Keyboard icon in Control Panel, the speed is still set even if you cancel the dialog box.

Load= *filename(s)*

Specifies one or more applications to run as icons when Windows for Workgroups is started. This entry is a list of application filenames or documents associated with an application, with each filename separated by a space. Specify the complete path if the file is not located in the WINDOWS directory. The default is none (blank). An analogous action to provide the same functionality offered by this entry is to add the application to the Startup group in Program Manager, then check Run Minimized in the Properties dialog box. To make changes to this entry, edit WIN.INI.

MenuDropAlignment= 0 | 1

Specifies whether menus open right-aligned or left-aligned with the menu title. If this entry is 0, menus open left aligned. If the entry is 1, menus open right-aligned. The default is 0. To make changes to this entry, edit WIN.INI.

MenuShowDelay= *milliseconds*

Specifies how long to wait before displaying a cascading menu. The default is **0** for 80386 computers and **400** for 80286 computers. To make changes to this entry, edit WIN.INI.

MouseSpeed= *0 | 1 | 2*

Sets the relationship between mouse and cursor movement when the value of either **MouseThreshold1=** or **MouseThreshold2=** is exceeded. When this occurs, Windows for Workgroups can cause cursor movement to accelerate according to the value of **MouseSpeed=**. If the value is **0**, there is no acceleration. If the value is **1**, the cursor is moved to twice the normal speed when mouse movement exceeds the value of **MouseThreshold1=**. If the value is **2**, the cursor is moved to twice the normal speed when the mouse movement exceeds the value of **MouseThreshold1=**, or four times the normal speed if mouse movement exceeds **MouseThreshold2=**. The default is **1**. To make changes to this entry, choose the Mouse icon from Control Panel.

MouseThreshold1= *pixels***MouseThreshold2=** *pixels*

These entries set the maximum number of pixels that the mouse can move between mouse interrupts before Windows for Workgroups alters the relationship between the mouse and cursor movement. If the mouse movement exceeds the threshold defined by **MouseThreshold=1** and if **MouseSpeed=** is greater than **0**, Windows for Workgroups moves the cursor at twice the normal speed. If the mouse movement exceeds the threshold defined by **MouseThreshold2=** and if **MouseSpeed= 2**, Windows for Workgroups moves the cursor at four times the normal speed. The defaults are **MouseThreshold=15** and **MouseThreshold2= 10**. To change these entries, choose the Mouse icon from Control Panel.

MouseTrails= *number*

Specifies the number of pointers shown on the screen when the MouseTrails option is selected in the Mouse dialog box in Control Panel. You can specify a number between **1** and **7**. This entry is supported only for the EGA, VGA, or SuperVGA display driver. The default is none (blank). To make changes to this entry, choose the Mouse icon from Control Panel. If the Mouse dialog box does not include the Mouse Trails option, you need to add this entry to WIN.INI to set mouse trails.

**NetMessage= 0 | 1**

Specifies whether Windows for Workgroups is to display a warning message if the network has already been started from the MS-DOS command prompt when Windows for Workgroups is run in 386 enhanced mode. Setting this value to **0** turns off the warning message. The default is **1**. To make changes to this entry, edit WIN.INI or select the option in the warning dialog box to prevent the warning message from being displayed in the future.

NullPort= string

Specifies the name used for a null port. This name appears in the Printers Connect dialog box in Control Panel when a device is installed (that is, the device driver is present) but is not connected to any port. This value also appears for such a device in the **[devices]** section of WIN.INI. The default is "**None**". To make changes to this entry, edit WIN.INI.

programs= extensions

Defines which files Windows for Workgroups considers applications. Separate the filename extension with a space and do not include the preceding periods. The default is: **com exe bat pif**. To make changes to this entry, edit WIN.INI.

Run= filename(s)

Enables Windows for Workgroups to run one or more specified applications when started. The value is a list of application filenames, or documents associated with applications, with each filename separated by a space. Specify the complete path if the application or document is not in the WINDOWS directory. The default is none (blank). An analogous action to provide the same functionality offered by this entry is to add the application to the Startup group in Program Manager. To make changes to this entry, edit WIN.INI.

ScreenSaveActive= 0 | 1

Specifies whether a screen saver is to be displayed if Windows for Workgroups is not actively being used. Set this value to **1** to use a screen saver; **0** turns off the screen saver. The default is **0**. To make changes to this entry, choose the Desktop icon from Control Panel.

ScreenSaveTimeout= seconds

Specifies the amount of time that Windows for Workgroups must be idle before the screen saver appears. The default is **120**. To make changes to this entry, choose the Desktop icon from Control Panel.

Spooler= Yes | No

Specifies whether the output to the printer that is generated by a Windows application is to be sent through Print Manager, or whether Print Manager will be used to support sharing locally-connected printers. Changing this value to **No** turns off Print Manager. The default is **Yes**. To make changes to this entry, choose the Printers icon from Control Panel and check or clear the Use Print Manager option. To support sharing locally-connected printers, Print Manager must be used and the value for this entry must be set to **Yes**.

SwapMouseButtons= 0 | 1

Specifies whether to swap the right and left mouse buttons. If the value is **1**, the buttons are swapped. The default is **0**. To make changes to this entry, choose the Mouse icon from Control Panel.

TransmissionRetryTimeout= seconds

Specifies the default amount of time to give Windows for Workgroups for attempted transmission retries. A value is saved for each printer in the **[PrinterPorts]** section. If a successful transmission does not occur during this time, Print Manager displays a message stating that the printer is not receiving characters. The default is **90** for PostScript printers; **45** for all other printers. To make changes to this entry, edit WIN.INI. This setting serves only as the system default. To change the value for a particular printer, choose the Printers icon from Control Panel or choose Printer Setup from the Options menu in Print Manager. Changing this setting with Control Panel or Print Manager, changes the corresponding value in the **[PrinterPorts]** section for the selected printer.

[WindowsHelp]

The **[WindowsHelp]** section includes settings that specify the size and placement of the help window and dialog boxes, and the color of text that displays a macro, pop-up window, or new panel when chosen. The **[WindowsHelp]** section can include the following entries:

M_WindowPosition= number, number, number, number, number

H_WindowPosition= number, number, number, number, number

A_WindowPosition= number, number, number, number, number

C_WindowPosition= number, number, number, number, number

These entries define the default size and position of the main help window, and the History, Annotate, and Copy dialog boxes. The first and second *number* values define the *x* and *y* coordinates of the upper-left corner for the default position of the main help window and the dialog boxes. The third and fourth *number* values define the default width and height (in pixels) of the help window and dialog boxes. The fifth *number* specifies when the main help window is maximized, where 1 is maximized, and 0 is the default height and width of the window. Changing this value has no effect on the size of the dialog boxes, because they cannot be maximized.

To make changes to this entry, move or size the main help window, or the History, Annotate, or Copy dialog boxes. The values are updated in WIN.INI automatically.

IFJumpColor= *red-value green-value blue-value*

When selected, specifies the color of text that leads to a new window of help information, located in a different Help file. To make changes to this entry, edit WIN.INI.

IFPopupColor= *red-value green-value blue-value*

When selected, specifies the color of text that displays a popup panel window (the type of panel used for glossary definitions) located in a different Help file. To make changes to this entry, edit WIN.INI.

JumpColor= *red-value green-value blue-value*

When selected, specifies the color of text that leads to a new window of help information. For example, specifying 000 000 000 as values results in black text on a white background. If no popup color is specified, the Jump Color also applies to text that, when chosen, displays a popup window. To make changes to this entry, edit WIN.INI.

MacroColor= *red-value green-value blue-value*

When selected, specifies the color of text that runs a help macro. To make changes to this entry, edit WIN.INI.

PopupColor= *red-value green-value blue-value*

When selected, specifies the color of text that displays a popup panel window (the type of panel used for glossary definitions). For example, specifying 000 000 000 as values results in black text on a white background. To make changes to this entry, edit WIN.INI.

SYSTEM.INI: System Initialization File

When you install Windows for Workgroups, Setup creates the SYSTEM.INI file. SYSTEM.INI contains global system information that Windows for Workgroups uses when it starts, and contains settings that you can customize to meet your system's hardware needs.

The SYSTEM.INI sections discussed in this chapter appear here in alphabetical order for easy reference, the sections may not appear in this sequence in the initialization file. The SYSTEM.INI file can contain the following sections:

Section	Purpose
[386Enh]	Contains information used by Windows for Workgroups in 386 enhanced mode.
[boot]	Lists drivers and Windows for Workgroups modules.
[boot.description]	Lists the names of devices you can change using Windows Setup.
[ClipShares]	Identifies the names of ClipBook pages that have been shared on this workstation.
[DDEShares]	Identifies the DDE shares defined in the DDE shares database that can participate in a Network DDE conversation.
[drivers]	Contains a list of aliases (or names) assigned to installable driver files.
[keyboard]	Contains information for the keyboard.
[mci]	Lists Media Control Interface (MCI) drivers.
[network]	Contains information for defining the Windows for Workgroups network configuration.
[NonWindowsApp]	Contains information used by MS-DOS-based applications.
[PasswordLists]	Contains settings that specify the location of the password-list files for each user who logs on to your computer.
[standard]	Contains information used by Windows for Workgroups in standard mode.

Setup assigns a value to each entry in the **[boot]** and **[keyboard]** sections, and to the **device=** entry and its synonyms in the **[386Enh]** section. These entries must appear in SYSTEM.INI for Windows for Workgroups to function properly. These entries do not have built-in values, but have specific values that Setup assigns during installation.

Many of the other entries described in this chapter are rarely needed and do not appear in the SYSTEM.INI file unless you add them. Most of these entries have a built-in value that is present whether or not the entry appears in SYSTEM.INI. You might need to change one or more of these values to improve the performance of Windows for Workgroups or a specific application.

The possible methods for changing values are noted for each entry in this chapter. Most of the SYSTEM.INI entries cannot be changed through Control Panel. You can change many of the settings by running Setup. Other values in SYSTEM.INI can be changed only by opening the file and editing it with a text editor such as Notepad.

The changes you make to the SYSTEM.INI file do not take effect until you restart Windows for Workgroups. Refer to “Editing .INI Source Files” earlier in this chapter.

[386Enh]

The **[386Enh]** section contains information specific to running Windows for Workgroups in 386 enhanced mode, including information used for virtual memory page swapping.

For entries in this section that specify virtual devices, the value can appear in two ways: either as the filename of a specific virtual device driver (with path if necessary), or as an asterisk (*) followed immediately by the device name to refer to a virtual device built into the WIN386.EXE file.

This section can contain the following entries:

32BitDiskAccess= Boolean

Turns 32-bit disk access on or off. The default is **Off** if your hard disk is capable of supporting 32-bit disk access. Otherwise, this setting is not present in SYSTEM.INI. To change this entry, choose the 386 Enhanced icon in Control Panel. If this setting is not present in SYSTEM.INI, the 32-Bit Disk Access check box does not appear in the Control Panel Virtual Memory dialog box.

32BitDiskWarning= Boolean

Used by the Control Panel's Virtual Memory dialog to determine whether to display a warning about using 32-bit disk access on portable computers. The purpose of this setting is to allow OEM who preinstall Windows for Workgroups to disable this warning which is not appropriate for preinstalled versions of Windows where the correct settings are known by the installer. If this setting is set to **Off**, then the 32-bit disk access warning is not displayed. The default is **On**—the 32-bit disk access warning is displayed. To change this entry, you must edit SYSTEM.INI. (You should not need to change this entry.)

A20EnableCount= number

Specifies the initial A20 enable count which identifies the A20 handler that HIMEM.SYS uses to access extended memory. For more information about the A20 enable count, see the documentation for your extended memory manager.

The default is the value computed by the Win386 loader. To change this entry, you must edit SYSTEM.INI. (You should not need to change this entry.)

AllEMSLocked= *Boolean*

If **On**, this entry locks the contents of the expanded memory that is used by all Windows and MS-DOS-based applications into memory (instead of swapping it to disk), and overrides PIF settings for EMS Memory Locked. Set this value to **On** if you are using a disk caching program that uses expanded memory. The default is **Off**. To change this entry, you must edit SYSTEM.INI.

AllVMsExclusive= *Boolean*

When **On**, this entry forces all applications to run in exclusive full screen mode, overriding all other settings in the application's PIFs. Setting **AllVMsExclusive= on** might help avoid UAEs when you are running network or memory-resident software that is not compatible with Windows for Workgroups. The default is **Off**. To change this entry, you must edit SYSTEM.INI.

AllXMSLocked= *Boolean*

If **On**, this entry locks the contents of the extended memory used by all Windows and MS-DOS-based applications into memory (instead of swapping it to disk), and overrides PIF settings for XMS Memory Locked. The default is **Off**. To change this entry, you must edit SYSTEM.INI. (You should not need to change this entry.)

AltKeyDelay= *seconds*

Specifies how much time Windows for Workgroups waits to process a keyboard interrupt after it processes an ALT interrupt. Some applications expect a slower processing rate than Windows for Workgroups in 386 enhanced mode usually uses. Increase this value if this type of an application has trouble handling the ALT key. The default is **.005**. To change this entry, you must edit SYSTEM.INI.

AltPasteDelay= *seconds*

Specifies how much time Windows for Workgroups waits before pasting any characters after the ALT key has been pressed. Some applications may require more time for the recognition of the ALT keystroke. The default is **.025**. To change this entry, you must edit SYSTEM.INI.

AutoRestoreScreen= *Boolean*

Specifies whether the display for MS-DOS-based applications is restored by Windows for Workgroups, or by the applications themselves when they become active. If this entry is **On**, Windows for Workgroups saves the display information for an MS-DOS-based application in memory when you switch away from the application. When you switch back to the application, Windows for Workgroups restores the screen.

If this entry is **Off**, the application must restore its own display by repainting the screen. This action requires less memory, but it can slow down performance since Windows for Workgroups can usually restore the screen faster. This entry only applies to VGA displays and affects only applications that notify Windows for Workgroups that they can update their screens automatically when Windows for Workgroups sends a display update call. The default is **On**. To change this entry, you must edit SYSTEM.INI.

BkGndNotifyAtPFault= *Boolean*

Specifies whether a notification is to be sent to an MS-DOS-based application or to Windows for Workgroups to prevent the application from attempting to access the display (which might corrupt the display of a different application). This notification can be sent while switching to a different application or when attempting to access the actual display. If this entry is **Off**, Windows for Workgroups sends a notification when switching between applications. Set this entry to **Off** if you are using a display that has special hardware (for example, 8514 and TIGA). If you are using a VGA display, setting this entry to **On** should work in most cases. The default is **On** for VGA displays; **Off** for 8514 displays. To change this entry, you must edit SYSTEM.INI.

CGA40WOA.FON= *filename*

Specifies the filename of the fixed-pitch display font used for MS-DOS-based applications with a display of 40 columns and 25 or fewer lines. The default is none (blank). To change this entry, exit Windows for Workgroups and run Setup from MS-DOS. Then change the Codepage or Display setting.

CGA80WOA.FON= *filename*

Specifies the filename of the fixed-pitch display font used for MS-DOS-based applications with a display of 80 columns and 25 or fewer lines. The default is none (blank). To change this entry, exit Windows for Workgroups and run Setup from MS-DOS. Then change the Codepage or Display setting.

CGANoSnow= *Boolean*

When **On**, Windows for Workgroups provides special handling which can prohibit snow from appearing on an IBM CGA display device. The default is **Off**. To change this entry, you must edit SYSTEM.INI.

COM1AutoAssign= *number* | *seconds*
COM2AutoAssign= *number* | *seconds*
COM3AutoAssign= *number* | *seconds*
COM4AutoAssign= *number* | *seconds*

These entries indicate the contention detection values for each connected communications port. These values are used by Windows for Workgroups to determine how to arbitrate requests for the use of a device by more than one application, at least one of which is an MS-DOS-based application. If the value is -1, Windows for Workgroups displays a warning message that asks you which application is to be given control of the port. If the value is 0, any application can use the device at any time. If the value is a positive integer less than 1000, this value represents the number of seconds that must pass after an application stops using the device, before another application can use the same device. The default is 2. To change this entry, choose the 386 Enhanced icon from Control Panel.

COM1Base= *address*
COM2Base= *address*
COM3Base= *address*
COM4Base= *address*

These entries specify the base (first) port for the serial port adapter you are using for both standard mode and 386 enhanced mode. Check your hardware documentation for the appropriate value. The defaults are **COM3Base= 3E8h** and the port address values in the BIOS data area for COM1, COM2, and COM4. To change this entry, choose the Ports icon in Control Panel and click the Advanced button for the selected COM port.

COMBoostTime= *milliseconds*

These entries specify the amount of time to allow a virtual machine to process a COM interrupt. If you are losing keyboard characters on the display while a communications application is running, you can try increasing this value. The default is 2. To change this entry, you must edit SYSTEM.INI.

COM1Buffer= *number*
COM2Buffer= *number*
COM3Buffer= *number*
COM4Buffer= *number*

These entries specify the number of characters that the device buffers on the corresponding communications port. Before changing one of these entries, make sure the corresponding **COMxProtocol=** entry has the proper value. Buffering may slow down communications on a port, but might be necessary to prevent some communications applications from losing characters at high baud rates. The size of the required buffer depends on the speed of the machine and the application's needs. The default is 128. Before increasing this value, see **COMxProtocol=**. To change this entry, you must edit SYSTEM.INI.

COM1FIFO= 0 | 1
COM2FIFO= 0 | 1
COM3FIFO= 0 | 1
COM4FIFO= 0 | 1

These entries specify whether the FIFO buffer of a COM port's 16550 Universal Asynchronous Receiver Transmitter (UART) is to be enabled (set to 1) or disabled (set to 0). If a serial port does not have a 16550 UART, this setting is ignored. These values are used by Windows for Workgroups for both standard mode and 386 enhanced mode. The default is 1.

COM1Irq= number
COM2Irq= number
COM3Irq= number
COM4Irq= number

These entries indicate which interrupt line is used by the device on the specified serial port. Check your hardware documents for the correct value. If there is a hardware conflict between ports, set a value of -1 to disable input for that COM port. The defaults for ISA and EISA machines are **COM1Irq= 4**, **COM2Irq= 3**, **COM3Irq= 4**, and **COM4Irq= 3**. For MCA machines, the defaults are **COM1Irq= 4**, **COM2Irq= 3**, **COM3Irq= 3**, and **COM4Irq= 3**. To change this entry, use the Advanced option of the Ports icon in Control Panel.

COM1Protocol= XOFF | blank
COM2Protocol= XOFF | blank
COM3Protocol= XOFF | blank
COM4Protocol= XOFF | blank

These entries specify whether Windows for Workgroups in 386 enhanced mode is to stop simulating characters in a virtual machine after the virtual machine sends an XOFF character. Set the value for a port to **XOFF** if a communications application using that port is losing characters while doing text transfers at high baud rates. Windows for Workgroups resumes simulating characters when the virtual machine sends another character after the XOFF character.

Leave this entry blank if the application does binary data transfers. Enabling this switch might suspend binary transmissions. The default is no entry which is the same as any entry other than **XOFF**. Windows for Workgroups does not check for XOFF characters if this entry is blank or set to the entry to anything other than **XOFF**. If the application continues to lose characters after this entry is set, try increasing the related **COMxBuffer=** value. To change this entry, you must edit SYSTEM.INI.

COMIrqSharing= *Boolean*

Specifies whether COM interrupt lines are sharable between multiple serial ports or with other devices. Set this switch if your machine uses the same interrupt for COM3 or COM4 and COM1 or COM2. The default is **On** for Micro Channel® and EISA machines; **Off** for all other machines. To change this entry, you must edit SYSTEM.INI.

COMMdrv30= *Boolean*

If **On**, the Virtual COMM Driver (VCD) uses its own copy of the interrupt handler for the serial communications driver. This improves the performance of the COM ports. Set this entry to **On** if you are using a Windows 3.0 serial communications driver (comm.dvr) or when you are using a third-party driver that is compatible with a Windows 3.0 communications driver. This entry is not used with a Windows 3.1 communications driver. Set this entry to **Off** if you are using the standard Windows 3.1 serial communications driver (identified as **device=*vcd**) or the Windows for Workgroups 3.1 serial communications driver (identified as **device=vcd.386**). The default is **Off**. To change this entry, you must edit SYSTEM.INI.

Device= *filename | *devicename*

Specifies which virtual devices are being used with Windows for Workgroups in 386 enhanced mode. This value can appear in two ways: either as the name of a specific virtual device file (with path if necessary) or as an asterisk (*) followed immediately by the device name to refer to a virtual device built into the WIN386.EXE file. Synonyms for **device=** are **display=**, **EBIOS=**, **keyboard=**, **network=**, **mouse=**, **secondnet=**, and **transport=**. Filenames usually include the .386 extension. Multiple device lines are required to run Windows for Workgroups in 386 enhanced mode. The default is none (blank). Setup assigns appropriate values based on your system configuration. To change this entry, you must edit SYSTEM.INI.

If Setup detects protected mode block devices for Western Digital compatible controllers, it adds the entries **device= *int13** and **device= *wdctrl** [**386Enh**]. A block device is a virtual device used in 386 enhanced mode to talk directly to disk drive controllers, bypassing MS-DOS and BIOS. To remove block device support, delete these entries in SYSTEM.INI. You do not need to delete files from your hard disk as these devices are appended to WIN386.EXE.

Display= *filename | *devicename*

Specifies the display device being used with Windows for Workgroups in 386 enhanced mode. This entry is a synonym for **Device=**. The default is none (blank). Setup assigns an appropriate value based on your system configuration. To change this entry, choose the Windows Setup icon from the Main Group window. In many cases, you may need to exit Windows for Workgroups and run Setup from MS-DOS.

DMABufferIn1MB= Boolean

If set to **On**, this entry indicates that the direct memory access (DMA) buffer memory is to be in the first 1 MB of memory (above 640K if possible) to be compatible with 8-bit bus master cards. The default is **Off**. To change this entry, you must edit SYSTEM.INI.

DMABufferSize= kilobytes

Specifies the amount of memory to be reserved for the DMA buffer. This memory is allocated above 640K if possible. Windows for Workgroups in 386 enhanced mode defaults to a DMA buffer size that handles disk access. The default is **16**. To change this entry, you must edit SYSTEM.INI.

DOSPromptExitInstruc= Boolean

If **On**, a message box appears when you start the MS-DOS prompt, displaying instructions on how to exit and switch away from the MS-DOS prompt. Set this entry to **Off** if you do not want to see the message. The default is **On**. To change this entry, you must edit SYSTEM.INI.

DualDisplay= Boolean

Normally, when Windows for Workgroups is in 386 enhanced mode, the memory between B000:0000 and B7FF:000F is used by the general system unless a secondary display is detected. This entry must be **On** if you are using a VGA-based color display and want EMM386.EXE to include this address space as an upper memory block (UMB). In addition to enabling this setting, you must include the **i=** option in the DEVICE=EMM386.EXE command line in your CONFIG.SYS file as follows:

```
DEVICE=EMM386.EXE i=B000-B7FF
```

If this entry is **Off**, the address range is available on EGA systems, not on VGA systems, because the VGA display device supports monochrome modes, which use this address space. To change this entry, you must edit SYSTEM.INI.

EBIOS= filename | *devicename

Specifies the extended BIOS device being used with Windows for Workgroups in 386 enhanced mode. This entry is a synonym for **Device=**. The default is the value that Setup assigns based on your system configuration. To change this entry, you must edit SYSTEM.INI.

EGA40WOA.FON= filename

Specifies the filename of the fixed-pitch display font used for MS-DOS-based applications with a display of 40 columns and more than 25 lines. The default is none (blank). To change this entry, exit Windows for Workgroups and run Setup from MS-DOS. Then change the Codepage or Display setting.

EGA80WOA.FON= filename

Specifies the filename of the fixed-pitch display font used for MS-DOS-based applications with a display of 80 columns and more than 25 lines. The default is none (blank). To change this entry, exit Windows for Workgroups and run Setup from MS-DOS. Then change the Codepage or Display setting.

EISADMA= Boolean | channel,size

Specifies the mode of operation of an extended DMA channel for Extended Industry Standard Architecture (EISA) machines only. This value can take one of two forms. If you set this entry to **Off**, Windows for Workgroups treats the machine as non-EISA avoiding all EISA-related logic. Set this value to **Off** if you cannot run Windows for Workgroups in 386 enhanced mode on an EISA machine. If you are using an EISA machine, you can also specify the default transfer size for one or more DMA channels. The channels can operate in these modes: 8-bit (**8**), 16-bit specified in words (**16w**), 16-bit specified in bytes (**16b**), or 32-bit (**32**). If you are not using an EISA machine, Windows for Workgroups ignores this entry. The default is **0,8; 1,8; 2,8; 3,8; 5,16w; 6,16w; 7,16w**. Each pair goes with its own **EISADMA=** entry. To change this entry, you must edit SYSTEM.INI.

EMMExclude= paragraph-range

Specifies a range of memory that Windows for Workgroups does not scan to find unused address space. This has the side effect of turning off the RAM and ROM search code for the range. The range (two paragraph values separated by a hyphen) must be between A000 and EFFF. This scanning can interfere with some adapters that use the same memory area. The starting value is rounded down and the ending value is rounded up to a multiple of 16K. For example, you could set **EMMExclude= C800-CFFF** to prevent Windows for Workgroups from scanning the addresses C800:0000 through CFFF:FFFF (the block that some VGA cards use to enhance performance). You can specify more than one range by including more than one **EMMExclude=** entry. The default is none (blank). To change this entry, you must edit SYSTEM.INI.

EMMInclude= paragraph-range

Specifies a range of memory that Windows for Workgroups treats as unused address space regardless of what may be there. **EMMInclude=** takes precedence over **EMMExclude=** if you specify ranges that overlap. The range (two values separated by a hyphen) must be between A000 and EFFF. The starting value is rounded down and the ending value is rounded up to a multiple of 16K. For example, you could set **EMMInclude= C800-CFFF** to ensure that Windows for Workgroups scans the addresses C800:0000 through CFFF:FFFF. You can specify more than one range by including more than one **EMMInclude=** entry. The default is none (blank). To change this entry, you must edit SYSTEM.INI.

EMMPageFrame= *paragraph*

Specifies the starting paragraph where the 64K page frame begins when Windows for Workgroups in 386 enhanced mode cannot find a suitable page frame. This entry allows an EMS page frame in an area containing some unused RAM or ROM. For example, you could set **EMMPageFrame= C400** to start the page frame at C400:0000. The default is none (blank). To change this entry, you must edit SYSTEM.INI.

EMMSize= *kilobytes*

Specifies the total amount of memory available to be mapped as expanded memory. The default value allocates the maximum possible amount of system memory as expanded memory. Specify a value for this entry if you run an application that allocates all of the available expanded memory. This is apparent when you run the application, and you cannot create new virtual machines. If this value is **0**, no expanded memory is allocated, but the EMM driver will load. To disable EMS and prevent the EMM driver from loading, use **NoEMMDriver= on**. The default is **65,536**. To change this entry, you must edit SYSTEM.INI.

**EnableSharingPopUps=** *Boolean*

Specifies whether a SHARE.EXE sharing violation message is to appear when a sharing violation occurs while using the VSHARE.386 device. If the value for this setting is **True**, the SHARE.EXE messages appear. Set this value to **True** if you are using an MS-DOS-based application that relies on the sharing-violation message. If the value for this setting is **False**, the SHARE.EXE message does not appear and you are not notified of a sharing violation. The default is **False**.

FileSysChange= *Boolean*

Indicates whether the File Manager automatically updates file information anytime an MS-DOS-based application creates, renames, or deletes a file. When this entry is **Off**, a virtual machine can be run exclusively even when it modifies files. Setting this entry to **On** can slow down system performance significantly. If you are sharing directories and someone else changes the contents of your directories using an MS-DOS-based application, File Manager does not update the directory or file information, even if this entry is **On**. The default is **Off** in 386 enhanced mode. The entry is not supported in standard mode. To change this entry, you must edit SYSTEM.INI.

Global= *device-name*

Defines MS-DOS devices that are loaded in CONFIG.SYS and need to be global to the system. The default is that all devices loaded in CONFIG.SYS are global to the system. But certain virtual devices might specify that an MS-DOS device is local (for example, MS\$MOUSE). Use this entry to override that local specification. The *device-name* value must exactly match the case of the device name in CONFIG.SYS or this entry does not work. Because most device names are in all capital letters this value is usually also in all capitals. To change this entry, you must edit SYSTEM.INI.

HardDiskDMABuffer= *kilobytes*

Specifies the amount of memory used for the Direct Memory Access (DMA) buffer. If you are using SMARTDrive and double buffering is turned on, this amount is determined automatically. You may need to change this value if you are using a hard disk that supports DMA and are not using SMARTDrive or if you do not have double buffering turned on. The default is **0** for AT architecture computers and **64** for Micro Channel computers or computers that use DMA channel 3 (unless you use SMARTDrive with double buffering in which case the default is **0**). To change this entry, you must edit SYSTEM.INI.

IdleVMWakeUpTime= *seconds*

Enables timer interrupts to periodically wake up idle virtual machines after the specified number of seconds has elapsed. If a virtual machine does not use timer interrupts (INT 8 or INT 1Ch), Windows for Workgroups does not usually force timer interrupts into a virtual machine unless it is active. This entry forces the timer interrupts to occur. The value for this entry is rounded down to the lowest power of 2 (for example, 1, 2, 4, 8, 16, 32, 64). The default is **8**. To change this entry, you must edit SYSTEM.INI.

IgnoreInstalledEMM= *Boolean*

If set to **On**, allows Windows for Workgroups to start in 386 enhanced mode even when an unknown expanded memory manager (EMM) is running. Starting Windows for Workgroups with an unknown EMM running can cause the system to fail if the memory-resident software was using expanded memory before Windows for Workgroups started. Set this entry only if no such software is installed or if you are sure it is not active when you run Windows for Workgroups. This entry applies only to EMMs servicing physical EMS hardware. Windows for Workgroups does not disable unrecognized 80386 expanded memory emulators. The default is **Off**. To change this entry, you must edit SYSTEM.INI.

InDOSPolling= *Boolean*

If set to **On**, prevents Windows for Workgroups from running other applications when memory-resident software has the InDOS flag set. Set this entry **On**, if the memory-resident software needs to be in a critical section to do operations off an INT 21 hook. Setting this entry to **On** slows down system performance slightly. The default is **Off**. To change this entry, you must edit SYSTEM.INI.

INT28Critical= *Boolean*

Specifies whether a critical section is needed to handle INT 28h interrupts used by memory-resident software. Some network virtual devices do internal task switching on INT 28h interrupts. These interrupts might hang some network software, indicating the need to add an **INT28hCritical=** entry. If you are not using such software, setting this entry to **Off** might improve Windows task switching. The default is **On**. To change this entry, you must edit SYSTEM.INI.

IRQ9Global= *Boolean*

If set to **On**, this entry converts IRQ 9 masks to global. Set this entry to **On** if your system hangs when it reads from a floppy drive or to make sure the system reads the floppy drive before starting Windows for Workgroups. The default is **Off**. To change this entry, you must edit SYSTEM.INI.

Keyboard= *filename | *devicename*

Specifies the keyboard driver for Windows for Workgroups in 386 enhanced mode. This entry is a synonym for **Device=**. The default is none (blank). Setup assigns an appropriate value based on your system configuration. To change this entry, choose the Windows Setup icon from the Main Group window.

KeyBoostTime= *seconds*

Specifies how much time an application gets to run with increased priority when it receives a keystroke. Use this entry to increase the response to keystrokes when several background applications are running. The default is **.001**. To change this entry, you must edit SYSTEM.INI.

KeyBufferDelay= *seconds*

Specifies the time to delay pasting keyboard input after the keyboard buffer is full. Some applications might require more than **.2** seconds. The default is **.2**. To change this entry, you must edit SYSTEM.INI.

KeyIdleDelay= *seconds*

Specifies how long Windows for Workgroups ignores idle calls after simulating a keystroke into a virtual machine. You can set this value to 0 to speed up keyboard input, although some applications might respond sluggishly. The default is .5. To change this entry, you must edit SYSTEM.INI.

KeyPasteCRSkipCount= *number*

Specifies the number of times that a read status INT 16 call is to return a status of *empty* for the keyboard buffer after pasting a carriage return and before pasting another character. When Windows for Workgroups pastes data from the Clipboard to an MS-DOS-based application, it must first paste the data to the BIOS keyboard buffer before pasting it into the application. This setting is used to slow down fast pasting from the Clipboard to the keyboard buffer so that the application can handle all incoming characters from the buffer. If you seem to lose characters, or if the screen does not update often enough while pasting information from the Clipboard, increase this value. This setting is related to **KeyPasteSkipCount=**. The default is 10. To change this entry, you must edit SYSTEM.INI.

KeyPasteDelay= *seconds*

Specifies how long to wait before pasting any characters after a key has been pasted. Some applications might require more time than .003 seconds for recognition of a keystroke. The default is .003. To change this entry, you must edit SYSTEM.INI.

KeyPasteSkipCount= *number*

Specifies the number of times that a read status INT 16 call is to return a status of *empty* for the keyboard buffer before pasting another character. When Windows for Workgroups pastes data from the Clipboard to an MS-DOS-based application, it must first paste the data to the BIOS keyboard buffer before pasting it into the application. This setting is used to slow down fast pasting from the Clipboard to the keyboard buffer enabling the application to handle all incoming characters from the buffer. If you seem to lose characters, or if the screen does not update often enough while pasting information from the Clipboard, increase this value. The default is 2. This setting is related to **KeyPasteCRSkipCount=**. To change this entry, you must edit SYSTEM.INI.

KeyPasteTimeout= *seconds*

Specifies how much time to allow an application to make the necessary BIOS calls for reading keyboard input before Windows for Workgroups changes from the fast paste (INT 16h) to the slow paste (INT 9h) mechanism. The default is 1. To change this entry, you must edit SYSTEM.INI.

KybdPasswd= Boolean

Specifies whether the Virtual Keyboard Device (VKD) is to support PS/2 8042 commands that implement password security. This entry only applies to 8042 keyboard controllers that are compatible with the PS/2 computer. The default is **On** for IBM PS/2 computers; **Off** for all other computers. To change this entry, you must edit SYSTEM.INI.

KybdReboot= Boolean

If this entry is **On**, Windows for Workgroups attempts to reboot the computer using a keyboard controller command. On some computers, this method is unreliable. If your computer hangs while rebooting, set this entry to **Off**. Windows for Workgroups quits and displays a prompt to press CTRL+ALT+DEL a second time if you attempt to reboot from the keyboard while Windows is running. The default is **On**. To change this entry, you must edit SYSTEM.INI.

Local= device-name

Defines MS-DOS device drivers that need to be local to each virtual machine. The *device-name* value must exactly match the case of the device name or this entry does not work. Because most device names are in all capital letters, this value is usually also in all capitals. If a device driver is local, a separate copy of it exists in each virtual machine, and each keeps different state information. Most device drivers do not function properly when this entry is used. An exception is CON, the MS-DOS console device, which should be local to avoid filling a buffer with input from multiple virtual machines. The default is none (blank). In a standard SYSTEM.INI file, Setup sets **Local= CON**. To change this entry, you must edit SYSTEM.INI.

LocalLoadHigh= Boolean

Determines how the upper memory blocks (UMBs) are used when Windows for Workgroups is running on MS-DOS version 5.0. If this entry is **Off**, Windows for Workgroups uses all of the upper memory area, leaving no extra UMBs available for virtual machines. If this entry is **On**, Windows for Workgroups does not use all of the upper memory area, and the UMBs are available locally to each virtual machine. The default is **Off**. To change this entry, you must edit SYSTEM.INI.

LocalReboot= On | Off

Specifies whether you can press CTRL+ALT+DEL to quit an application that caused an unrecoverable error in 386 enhanced mode. If this setting is **On**, you can quit the application, without restarting Windows for Workgroups. If this setting is **Off**, pressing CTRL+ALT+DEL restarts your entire system (as it usually does). To change this entry, you must edit SYSTEM.INI.

LPT1AutoAssign= *seconds*
LPT2AutoAssign= *seconds*
LPT3AutoAssign= *seconds*
LPT4AutoAssign= *seconds*

These entries apply to Windows version 3.0 only. If you have upgraded to Windows for Workgroups 3.1, these settings can still appear in your SYSTEM.INI file, but have no effect. The default is 60.

LRULowRateMult= *number*

Specifies the value that is used to determine the Least Recently Used (LRU) low paging rate sweep frequency which is computed by multiplying the value for **LRUSweepFreq=** by the value specified for **LRULowRateMult=**. Values between 1 and 65535 can be used. The default is 10. To change this entry, you must edit SYSTEM.INI. (You should not need to change this entry.)

LRURateChngTime= *milliseconds*

Specifies the length of time that Virtual Memory Manager (VMM) stays at high rate with no paging before switching to low rate, and the length of time VMM stays at low rate with no paging before turning the LRU sweep off. The default is 10,000. To change this entry, you must edit SYSTEM.INI. (You should not need to change this entry.)

LRUSweepFreq= *milliseconds*

Specifies the time between LRU sweep passes. This is also the high paging rate sweep frequency. The default is 250. To change this entry, you must edit SYSTEM.INI. (You should not need to change this entry.)

LRUSweepLen= *length-in-pages*

Specifies the length in pages of the region swept on each pass. Windows for Workgroups computes this value by dividing the value of **LRUSweepReset=** by the value of **LRUSweepFreq=**. The value must be at least 1. The default is 1024. To change this entry, you must edit SYSTEM.INI. (You should not need to change this entry.)

LRUSweepLowWater= *number*

Specifies whether the LRU sweeper is to be turned on. When the number of free pages drops below this value, the sweeper is turned on. The default is 24. To change this entry, you must edit SYSTEM.INI. (You should not need to change this entry.)

LRUSweepReset= *milliseconds*

Specifies the time desired for an ACC bit reset divided by 4 MB of pages. This means, the time to reset all ACC bits is *number of pages in system*+1023/1024 where 1024 pages = 4 MB. The minimum value is 100. The default is 500. To change this entry, you must edit SYSTEM.INI. (You should not need to change this entry.)

MapPhysAddress= *range*

Specifies the address range (in megabytes) in which the memory manager preallocates physical page-table entries and linear address space. Set a value for this entry if you are using an MS-DOS device driver that needs this contiguous memory (such as an older version of RAMDrive that uses extended memory). The default is none (blank). To change this entry, you must edit SYSTEM.INI.

MaxBPs= *number*

Specifies the maximum number of break points (a method for transferring control to Windows for Workgroups in 386 enhanced mode) that can be used by the Virtual Machine Manager. You may need to increase this value if you are using Microsoft C version 7.0 or higher with a debug version of Windows, or a third-party virtual device driver that requires more break points than the default value. The default is 200. To change this entry, you must edit SYSTEM.INI.

MaxCOMPort= *number*

Specifies the maximum number of COM ports supported in 386 enhanced mode. Change this value if you have more than four COM ports in your system. The default is 4. To change this entry, you must edit SYSTEM.INI.

MaxDMAPGAddress= *address*

Specifies the maximum physical page address that can be used for DMA. This entry works only with hard drives that support DMA. The default is 0FFFh for non-EISA computers and 0FFFFFFh for EISA computers. To change this entry, you must edit SYSTEM.INI.

MaxPagingFileSize= *kilobytes*

Specifies the maximum size for a temporary swap file. The default is 50 percent of the available disk space, but no more than three times the available physical memory. To change this entry, choose the 386 Enhanced icon in Control Panel. Then choose Virtual Memory to set a value.

MaxPhysPage= *hexadecimal-page-number*

Specifies the maximum physical page number that VMM can manage as a usable page allowing pages to be added at a physical address beyond what Virtual Memory Manager (VMM) recognized during initialization. If the value specified is less than what VMM determines, VMM ignores several physical pages that it would usually use, preventing the use of memory. This is useful if you are using a hardware device that cannot recognize all of the physical memory in your computer (for example, ISA DMA network cards cannot access physical memory above 16 MB). The default is determined by Windows for Workgroups based on the highest physical page number detected by VMM during initialization. To change this entry, you must edit SYSTEM.INI.

MCADMA= *Boolean*

Specifies whether Windows for Workgroups is to use the Micro Channel (MCA) extensions to DMA. This entry only applies to MCA computers. Set this entry to **Off** if you are using an MCA computer, but the DMA extensions are not implemented. The default is **On** for most MCA computers; **Off** for all other computers. To change this entry, you must edit SYSTEM.INI.

MessageBackColor= *vga-color-attribute*

Specifies the background color of message screens, such as the screen that appears when you press CTRL+ALT+DEL. The default is **1** (for blue). To change this entry, you must edit SYSTEM.INI.

MessageTextColor= *vga-color-attribute*

Specifies the color used to display text in message screens, such as the screen that appears when you press CTRL+ALT+DEL. The default is **F** (for white). To change this entry, you must edit SYSTEM.INI.

MinTimeSlice= *milliseconds*

Specifies the minimum time a virtual machine is allowed to run before other virtual machines can take over. A smaller value (such as 10 milliseconds) makes multitasking appear smoother, but diminishes the overall system performance. The default is **20**. To change this entry, choose the 386 Enhanced icon from Control Panel.

MinUnlockMem= *kilobytes*

Specifies the amount of memory that must remain unlocked and available for use when resuming a virtual machine, if more than one virtual machine is running. The default is **40**. To change this entry, you must edit SYSTEM.INI. (You should not need to change this entry.)

MinUserDiskSpace= *kilobytes*

Tells Windows for Workgroups how much disk space to leave free when creating a temporary swap file. Use this entry if your system's paging drive has less available space than Windows for Workgroups can use for paging. This entry has no effect if a permanent swap file exists. The default is 500. To change this entry, you must edit SYSTEM.INI.

Mouse= *filename | *devicename*

Specifies the virtual device that handles the mouse hardware in 386 enhanced mode. This entry is a synonym for **Device=**. The default is none (blank). Setup assigns an appropriate value based on your system configuration. To change this entry, choose the Windows Setup icon in the Main Group window.

MouseSoftInit= *Boolean*

Specifies whether Windows for Workgroups is to convert INT 33h function 0 hard initialization calls to function 33 soft initialization calls which do not reset the mouse hardware. Set this entry to **On** if you want to use a mouse with an MS-DOS-based application started in a window. Set this entry to **Off** if the cursor and screen information appear distorted when you are using the mouse with an application. If you set this entry to **Off**, you cannot use the mouse with an MS-DOS-based application started in a window. The default is **On**. To change this entry, you must edit SYSTEM.INI.

NetAsynchFallback= *Boolean*

If set to **On**, Windows for Workgroups attempts to save a failing NetBIOS request. When an application issues an asynchronous NetBIOS request, Windows for Workgroups attempts to allocate space in its global network buffer to receive the data. If there is insufficient space in the global buffer, Windows for Workgroups typically fails the NetBIOS request. If this entry is **On**, Windows for Workgroups attempts to save such a request by allocating a buffer in local memory and preventing any other virtual machines from running until the data is received and the timeout period specified by **NetAsynchTimeout=** expires. The default is **Off**. To change this entry, you must edit SYSTEM.INI. (This entry is ignored by VNB.386 and other protected-mode network protocols.)

NetAsynchTimeout= *seconds*

Specifies the timeout period when Windows for Workgroups enters a critical section to service an asynchronous NetBIOS request. This value is used only when **NetAsynchFallback=** is set. This value can include a decimal, such as 0.5. The default is 5.0.

NetDMASize= *kilobytes*

Specifies the DMA buffer size for NetBIOS transport software if a network has been installed. In this case, the buffer size is the larger value between this value and the value of **DMABufferSize=**. The default is **32** for Micro Channel machines (IBM PS/2 or compatible) and **0** for non-Micro Channel machines (IBM PC/AT or compatible). To change this entry, you must edit SYSTEM.INI. (This entry is ignored by VNB.386 and other protected-mode network protocols.)

NetHeapSize= *kilobytes*

Specifies the size (in 4K increments) of the data transfer buffers in conventional memory that Windows for Workgroups allocates for transferring data over a network in 386 enhanced mode. This entry is only needed if you are using real mode protocols. This entry is not required if you are using the vnb.386 protocol. All values are rounded to the nearest 4K value. The default value is **12**. To change this entry, you must edit SYSTEM.INI. (This entry is ignored by VNB.386 and other protected-mode network protocols.)

**Network=** *filename | devicename*

Specifies the virtual network drivers that are used when Windows for Workgroups is in 386 enhanced mode. The default is **vnetbios.386**, **vnetsup.386**, **vredir.386**, **vserver.386**, **vbrowse.386**, **vw.386**. This entry is a synonym for **Device=**. Setup assigns an appropriate value based on your system configuration.

NMIReboot= *Boolean*

If this entry is **On**, a reboot occurs when a non-maskable interrupt is received. The default is **Off**. To change this entry, you must edit SYSTEM.INI.

NoEMMDriver= *Boolean*

If this entry is **On**, prevents Windows for Workgroups in 386 enhanced mode from installing the expanded memory driver. This differs from setting **EMMSize= 0** which prevents UMBs from being allocated, but does not prevent the EMM driver from being loaded. The default is **Off**. To change this entry, you must edit SYSTEM.INI.

NoWaitNetIO= *On | Off*

Specifies whether to convert synchronous NetBIOS commands to asynchronous commands. When this entry is **On**, synchronous NetBIOS commands are converted. This can improve overall system performance when running several applications. The default is **On**. To change this value, you must edit SYSTEM.INI. (This entry is ignored by VNB.386 and other protected-mode network protocols.)

OverlappedIO= *Boolean*

If **On**, several virtual machines can make read and write requests to a disk before the first request has been completed. If this entry is **Off**, virtual machines cannot issue a request to read or write to a disk until any previous read and write requests have been completed. The default is **Off** if **InDOSPolling= on**. If not, the default is **On**. To change this entry, you must edit SYSTEM.INI. (You should not need to change this entry—Setup adds this entry if NetWare is configured as a secondary network.)

PageBuffers= *number*

Specifies the number of 4K page buffers that are used to store asynchronous read and write pages. This setting is used only if you are using 32-Bit Disk Access and a permanent swap file. You can specify a value between 0 and 32. Increasing this value can improve Windows for Workgroups performance. The default is 4. To change this entry, you must edit SYSTEM.INI.

PageOverCommit= *megabytes*

Specifies the multiplier that determines the amount of linear address space Virtual Memory Manager creates for the system. The amount of linear address space is computed by rounding up the amount of available physical memory to the nearest 4 MB and then multiplying that value by the value specified for **PageOverCommit=**. Increasing this value increases the amount of available linear address space causing the size of data structures to increase. This also increases paging activity proportionately and can slow down the system. You can specify a value between 1 and 20. The default is 4. To change this entry, you must edit SYSTEM.INI.

Paging= *Boolean*

Sets demand paging (virtual memory). Set this entry to **Off** only if you need the disk space that would be used for a temporary swap file. The default is **On**. To change this entry, you must edit SYSTEM.INI.

PagingDrive= *drive-letter*

Specifies the disk drive where Windows for Workgroups in 386 enhanced mode allocates a temporary swap file. This entry is ignored if you have a permanent swap file. If you don't have a permanent swap file and no drive is specified or the specified drive does not exist, Windows for Workgroups attempts to put the temporary swap file on the drive containing the SYSTEM.INI file. If the specified drive is full, paging is turned off. The default is none (blank). To change this entry, you must edit SYSTEM.INI.

PagingFile= *path-filename*

Specifies the path and filename for the temporary swap file that is created when you start Windows for Workgroups in 386 enhanced mode. This file is deleted when you quit Windows for Workgroups. This setting overrides the **PagingDrive=** entry. The default is **WINDOWS\WIN386.SWP**. To change this entry, you must edit SYSTEM.INI.

PerformBackfill= *Boolean*

Specifies whether to allocate a full 640K of memory to a computer that has less than 640K of base conventional memory. Setting this entry overrides the automatic checking done by Windows for Workgroups. The default is **On**—Windows for Workgroups automatically checks to see if it needs to fill in memory based on how MS-DOS is occupying memory. To change this entry, you must edit SYSTEM.INI. (You should not need to change this entry, because Windows for Workgroups can automatically detect whether to perform a backfill.)

PermSwapDOSDrive= *drive-letter*

Specifies the disk drive where Windows for Workgroups in 386 enhanced mode creates a permanent swap file. This setting is ignored if you are using a temporary swap file. The default is none (blank). To change this entry, choose the 386 Enhanced icon from Control Panel.

PermSwapSizeK= *kilobytes*

Specifies the desired size of a permanent swap file. The default is none (blank). To change this entry, choose the 386 Enhanced icon from Control Panel.

PerVMFILES= *number*

Specifies the number of private file handles Windows for Workgroups is to allocate to each virtual machine. Increase this value if an application does not have enough file handles to run. The total number of file handles, including the global handles specified in the **FILE=** statement in CONFIG.SYS, cannot exceed 255. If it does exceed 255, this value is rounded down. Set this value to **0** to prevent the allocation of any private file handles. The default is **10**. If the MS-DOS Share utility or the VSHARE.386 VxD is installed, this setting in SYSTEM.INI is ignored. To change this entry, you must edit SYSTEM.INI.

ReflectDosInt2A= *Boolean*

Indicates whether Windows for Workgroups is to consume or reflect MS-DOS INT 2A signals. If the value is **Off**, Windows for Workgroups consumes these signals and, for this reason, runs more efficiently. Set this entry to **On** if you are running memory-resident software that relies on detecting INT 2A messages. The default is **Off**. To change this entry, you must edit SYSTEM.INI.

ReservedHighArea= *paragraph-range*

Specifies a range of memory that Windows for Workgroups does not scan to find unused address space. This also turns off the RAM and ROM search code for the range. This scanning can interfere with some adapters that use the same memory area. The range (two paragraph values separated by a hyphen) must be between **A000** and **FFFF**. The starting value is rounded down and the ending value is rounded up to a multiple of 4K. For example, you could set **ReservedHighArea= E100-E3FF** to prevent Windows for Workgroups from scanning the first 12K of memory starting at E100. You can specify more than one range by including more than one **ReservedHighArea=** entry. The default is none (blank). To change this entry, you must edit SYSTEM.INI.

ReservePageFrame= *Boolean*

Tells Windows for Workgroups whether to give preference to EMS page frame space or conventional memory when it has to use one of these to allocate MS-DOS transfer buffers. This choice is necessary when Windows for Workgroups cannot find space between 640K and 1 MB other than by using EMS page frame space. If set to **On**, this entry preserves EMS page frame space at the expense of conventional memory. If you are not running MS-DOS-based applications that use expanded memory, set this entry to **Off** to give MS-DOS-based applications more conventional memory. The default is **On**. To change this entry, you must edit SYSTEM.INI.

ReserveVideoROM= *Boolean*

If this entry is **On**, video ROM exists in pages C6 and C7. If the text font appears scrambled when you start MS-DOS-based applications, set this entry to **On**. The default of **Off** enables Windows for Workgroups to detect video ROM correctly. (You should not need to change this entry.)

ROMScanThreshold= *number*

Specifies a parameter that is used to determine when an area of memory that has no ROM header or incorrect header information in the adapter area (usually between C000-FFFF) is ROM. The number indicates how many transitions (value changes) must occur within the memory area to determine when it is ROM. If the number of transitions is greater than the value for this entry, Windows for Workgroups recognizes the memory as ROM. If the number of transitions is less than the value, Windows for Workgroups recognizes the memory as usable memory. If this entry is **0**, Windows for Workgroups recognizes the memory area as ROM and then reserve it as unusable. The default is **20**. (You should not need to change this entry.)

ScrollFrequency= *number*

Specifies the number of lines you can scroll in an MS-DOS-based application that is running in Windows for Workgroups before its display is updated. The default is **2**. (You should not need to change this setting.)

**SecondNet=** *filename*

Specifies the virtual network drivers for the secondary networks you have configured on your workstation. The **secondnet=** entry is synonymous with **device=**. The default is none (blank). Setup sets this value to match your configuration. (For example, if NetWare is selected as the secondary network, the **secondnet=** entry will be defined as **secondnet=vipx.386, vnetware.386**.)

SGrabLPT= *port-number*

Routes all printer interrupts on the specified port to the system virtual machine, rather than to the current virtual machine. The default is none (blank).

SyncTime= *Boolean*

If **On**, Windows for Workgroups periodically synchronizes its time with the computer's CMOS clock. If this entry is **Off**, Windows for Workgroups usually maintains the correct time, unless **TrapTimerPorts= off** when you are running applications that can cause the system time to run faster or slower than the actual time. This setting is related to the **TrapTimerPorts=** entry. The default is **On**. To change this entry, you must edit SYSTEM.INI. (You should not need to change this entry.)

SystemROMBreakPoint= *Boolean*

Specifies whether Windows for Workgroups is to use ROM address space between F000:0000 and 1 MB for a break point. Windows for Workgroups in 386 enhanced mode usually searches this space to find a special instruction that is used as a system break point. If this address space contains something other than permanently available ROM, set this entry to **Off**. The default is **On** if Windows for Workgroups is started in the real mode of the 80386 (or higher) CPU. The default is **Off** if Windows for Workgroups is started in the virtual mode of the 80386 (or higher) CPU. Most 386 memory managers, such as Qemm™ and 386MAX, require this value to be set to **Off**. To change this entry, you must edit SYSTEM.INI.

SysMEMSLimit= *number | kilobytes*

Specifies how many kilobytes of expanded memory Windows for Workgroups is to be permitted to use. Setting this value to **0** prevents Windows for Workgroups from gaining access to any expanded memory. If this entry is **-1** Windows for Workgroups can gain all of the requested available expanded memory. The default is **2048**. To change this entry, you must edit SYSTEM.INI.

SysVMEMSLocked= *Boolean*

Indicates whether to swap the expanded memory being used by MS-DOS-based applications to the hard disk. Locking expanded memory can improve the performance of a specific MS-DOS-based application, but locking it slows down the rest of the system. The default is **Off**. To change this entry, you must edit SYSTEM.INI.

SysVMEMSRequired= *kilobytes*

Specifies how much expanded memory must be free to start Windows for Workgroups. Leave this value at **0** if there are MS-DOS-based application that are running under Windows for Workgroups and do not require expanded memory. The default is **0**. To change this entry, you must edit SYSTEM.INI.

SysVMV86Locked= *Boolean*

If set to **On**, the virtual mode memory used in the system's virtual machine remains locked in memory and cannot be swapped out to disk. Because Windows for Workgroups handles this process, **On** is usually not required. The default is **Off**. To change this entry, you must edit SYSTEM.INI.

SysVMXMSLimit= *number | kilobytes*

Specifies the maximum amount of memory the extended memory driver allocates to MS-DOS device drivers and memory-resident software in the system virtual machine. Set the value to **-1** to give an application all the available extended memory that it requests. The default is **2048**. To change this entry, you must edit SYSTEM.INI.

SysVMXMSRequired= *kilobytes*

Specifies how many kilobytes of extended memory must be reserved by the XMS driver to start Windows for Workgroups. Leave this value at **0** if there are no XMS users in the system virtual machine. The default is **0**. To change this entry, you must edit SYSTEM.INI.

TimerCriticalSection= *milliseconds*

Instructs Windows for Workgroups to go into a critical section around all timer interrupt codes and specifies a timeout period. Specifying a positive value enables only one virtual machine to receive timer interrupts at a time. Some networks and other global memory-resident software may fail unless this entry is used. Using this entry slows down performance and can make the system sluggish or seem to stop for short periods of time. The default is **0**. To change this entry, you must edit SYSTEM.INI.

TokenRingSearch= *Boolean*

If this entry **On**, Windows for Workgroups searches for a token ring network adapter on machines with IBM PC/AT architecture. Set this entry to **Off** if you are not using a token ring card and the search interferes with another device. The default is **On**. To change this entry, you must edit SYSTEM.INI.

TranslateScans= *Boolean*

If **On**, this entry indicates when Windows for Workgroups translates a keyboard's scan codes to make them compatible with standard IBM scan codes. This entry is necessary only for keyboards that generate non-standard scan codes. This entry is used only for the Switcher Screen and full screen message boxes. The default is **Off**. To change this entry, you must edit SYSTEM.INI.

**Transport=** *filename*

Specifies the virtual device driver file for the protected-mode network transport protocol that Windows for Workgroups uses in 386 enhanced mode. The **secondnet=** entry is synonymous with **device=**. The default entry is defined as **vnb.386** (Microsoft NetBEUI).

TrapTimerPorts= *Boolean*

If **Off**, this entry specifies whether Windows for Workgroups is to trap read and write operations that are performed by applications, to the system timer ports. If this entry is **Off**, Windows for Workgroups does not trap these operations, allowing applications that frequently read or write to the timer to run faster. This may interfere with ability of Windows for Workgroups to keep accurate system time. If this entry is **Off**, Windows for Workgroups can usually detect when an application has changed the timer interrupt interval and make any adjustments to the time. If your system's time appears to be running fast or slow, set this entry to **On** and Windows for Workgroups denies access to the timer ports. This ensures that the system time is accurate, but may cause applications that frequently access the timer ports to run slowly. You can also set **SyncTime= On** instead of this entry to cause Windows for Workgroups to check the time periodically and then make any necessary adjustments, regardless of when applications access the timer ports. The default is **On**. To change this entry, you must edit SYSTEM.INI.

UseableHighArea= *paragraph-range*

Specifies the range of memory that Windows for Workgroups scans for unused address space regardless of what may be there. **UseableHighArea=** takes precedence over **ReservedHighArea=** if you specify ranges that overlap. The range (two values separated by a hyphen) must be between **A000** and **FFFF**. The starting value is rounded down and the ending value is rounded up to a multiple of 4K. For example, you could set **UseableHighArea= E100-E3FF** to ensure that Windows for Workgroups scans the first 12K of memory starting at E100.

You may specify more than one range by including more than one entry for **UseableHighArea=**. The default is none (blank). To change this entry, you must edit SYSTEM.INI.

UseInstFile= *Boolean*

Specifies whether Windows for Workgroups is to look in the INSTANCE.386 file for information it can use to determine when data structures within MS-DOS need to be local. Two other methods give Windows for Workgroups this information: internal tables within the device and an INT 2Fh call documented in the OEM Adaptation Kit. Both methods are preferable to using this setting. This entry is provided only for compatibility with Windows/386 version 2.x. The default is **Off**. To change this entry, you must edit SYSTEM.INI. (You should not need to change this entry.)

UseROMFont= *Boolean*

If this entry is **On**, Windows for Workgroups uses the soft font stored in video ROM for displaying messages that appear when MS-DOS-based applications are running in a full screen and for displaying the text that appears when switching away from an MS-DOS-based application. Set this entry to **Off** if the font used to display messages is different from the font used in the application, or if random dots and shapes appear on your screen. The default is **On**. To change this entry, you must edit SYSTEM.INI.



V86ModeLANAs= *lane number, lane number*

Limits the real-mode NetBIOS LANA numbers that will be supported while Windows for Workgroups is running to only those LANA numbers listed. This setting is for real mode NetBIOS drivers only. This setting should not include any LANA numbers for protected mode protocols or NetBIOS drivers. This setting is useful with NetBIOS protocols which incorrectly identify themselves as being active on LANA numbers others than those assigned to them, for example, the NetWare NETBIOS.EXE TSR. Make sure that the values for this setting do not include LANA numbers assigned to protected mode protocols such as VNB.386. The default is none (blank). The LANAs for transport protocols are specified in PROTOCOL.INI. To change this entry, you must edit SYSTEM.INI.

VGAMonoText= *Boolean*

Tells Windows for Workgroups to ignore the video memory address space in VGA displays usually used only with monochrome adapters for VGA displays. When this entry is **Off**, Windows for Workgroups can use the B000h through B7FF range in upper memory if no hardware device is using these addresses and other applications do not use the monochrome display mode of the VGA adapter. The default is **On**. To change this entry, you must edit SYSTEM.INI.

VideoBackgroundMsg= Boolean

If this entry is **On**, Windows for Workgroups displays a message when a background application is suspended, or if its display cannot be updated properly because video memory is low. If this entry is **Off**, the warning message is turned off. This entry affects all MS-DOS-based applications currently running. The default is **On**. To change this entry, you must edit SYSTEM.INI.

VideoSuspendDisable= Boolean

Specifies whether to suspend an application running in the background if its display becomes corrupted for VGA displays only. If this entry is **On**, the application continues running. If this entry is **Off**, applications are suspended and a warning message appears if the **VideoBackgroundMsg= On**. The default is **Off**. To change this entry, you must edit SYSTEM.INI.

VirtualHDIrq= Boolean

If this entry is **On**, Windows for Workgroups in 386 enhanced mode terminates interrupts from the hard disk controller, bypassing the ROM routine that handles these interrupts. Some hard drives might require this entry to be **Off** for interrupts to be processed correctly. If this entry is **Off**, the ROM routine handles these interrupts, which slows the system's performance. The default is **On** for AT-compatible computers; **Off** for other computers. To change this entry, you must edit SYSTEM.INI.

WindowKBRequired= kilobytes

Specifies how much conventional memory must be free to start Windows for Workgroups. The default is 256. To change this entry, you must edit SYSTEM.INI.

WindowMemSize= number | kilobytes

Limits the amount of conventional memory Windows for Workgroups can use. The default value indicates that Windows for Workgroups can use as much conventional memory as it needs. Try entering a positive value less than 640 if there is not enough memory to run Windows for Workgroups in 386 enhanced mode. The default is -1. To change this entry, you must edit SYSTEM.INI.

WindowUpdateTime= milliseconds

Specifies the time Windows for Workgroups takes between updates of the display for an MS-DOS-based application running in a window. The default is 50. To change this entry, you must edit SYSTEM.INI.

WinExclusive= Boolean

If this entry is **On**, Windows for Workgroups obtains exclusive use of the computer's processing time when a Windows application is in the foreground.

If this entry is **On** and a Windows application is in the foreground, no MS-DOS-based application is run. The default is **Off**. To change this entry, choose the 386 Enhanced icon from Control Panel.

WinTimeSlice= *number, number*

Contains two numbers. The first number specifies the relative processing time (based on **MinTimeSlice=**) given to all Windows-based applications running in the foreground relative to the time allocated to all MS-DOS-based applications running in the background. The second number specifies the relative processing time given to all Windows-based applications running in the background when an MS-DOS-based application is running in the foreground. The range of numbers that can be entered is 1 through 10000. The important value is the ratio of these values to the corresponding settings in the PIFs of the active MS-DOS-based applications. The default is **100,50**. To change this entry, choose the 386 Enhanced icon from Control Panel.

WOAFont= *font filename*

Specifies which font files are loaded into memory when running MS-DOS-based applications. These fonts are also available to Windows-based applications as long as an MS-DOS-based application is running. If you want to make these fonts always available to Windows-based applications, add them to your system with the Fonts option in Control Panel. The default is **DOSAPP.FON**. To change this entry, exit Windows for Workgroups and run Setup from MS-DOS, then change the Codepage or Display setting.

XlatBufferSize= *kilobytes*

Specifies the size of the low memory buffer used to map MS-DOS calls from protected mode to virtual 386 enhanced mode. Specifying a higher value can improve the performance of protected mode Windows-based applications that read or write information in a large number of bytes (for example, some database applications). This may decrease the amount of memory available for running MS-DOS-based applications and may cause problems on networks that use named pipes. These networks may require a value of 4. The value you specify is rounded to the next 4K increment. For example, if you set this value to 5, Windows for Workgroups rounds it up to 8. The default is 8. To change this entry, you must edit SYSTEM.INI.

XMSUMBInitCalls= *Boolean*

Specifies whether Windows for Workgroups is to call the extended memory driver's UMB management routines. The default is usually **On**, but may be **Off** depending on the extended memory manager you are using. (You should not need to change this entry.)

[boot]

The **[boot]** section contains a list of the drivers and Windows modules that Windows for Workgroups uses to configure itself each time you start it.

Important All entries in this section with the exception of **CachedFileHandles=** are required. If you modify or delete one of these entries, Windows for Workgroups might not start or operate properly. Setup assigns all values based on your system configuration. There are no specific default values for these entries.

The **[boot]** section can contain the following entries:

286grabber= *filename*

Specifies the filename of the grabber, which is the device driver that makes an MS-DOS-based application visible when you run Windows for Workgroups in standard mode. The default is none (blank). To change this entry, choose the Windows Setup icon from the Main Group window and change the Display setting. If you are installing a device driver that is not included with Windows for Workgroups, exit Windows for Workgroups and run Setup from MS-DOS.

386grabber= *filename*

Specifies the filename of the grabber, which is the device driver that makes an MS-DOS-based application visible when you run Windows for Workgroups in 386 enhanced mode. The default is none (blank). To change this entry, choose the Windows Setup icon from the Main Group window and change the Display setting. If you are installing a device driver that is not included with Windows for Workgroups, exit Windows for Workgroups and run Setup from MS-DOS.

CachedFileHandles= *number*

Specifies the number of the most recently used executable files (.EXE) and dynamic-link library files (.DLL) that can remain open. Windows for Workgroups keeps these files open to enable the files to be accessed quickly, ensuring optimal performance. However, some networks have a limit on the number of files that can be open on a server at a time. If you have problems running Windows for Workgroups from a network server, use a lower number for this value. Valid numbers are 2 through 12. The default is 12. To change this entry, you must edit SYSTEM.INI.

comm.drv= *filename*

Specifies the filename of the serial communications driver you are using. The default is none (blank). To change this entry, you must edit SYSTEM.INI.

display.drv= *filename*

Specifies the filename of the display driver you are using. The default is none (blank). To change this entry, choose the Windows Setup icon from the Main Group window. If you are installing a device driver that is not included with Windows for Workgroups, run Setup.

drivers= *filename* | *aliasname*

Specifies the filenames or aliases of the installable drivers that load when Windows for Workgroups is started. An installable driver is a dynamic-link library that Windows for Workgroups treats like a device driver. If the drivers you want to use include parameters, specify the parameters by the alias name as defined in the **[drivers]** section of SYSTEM.INI, where you can list several filenames or aliases. The default is none (blank). Most Setup programs for Windows-based applications add these settings to the SYSTEM.INI file when installing drivers, so you shouldn't need to change the values. To change this entry, you must edit SYSTEM.INI.

fixedfon.fon= *filename*

Specifies the filename of the fixed system font used by Windows version 2.x applications. The default is none (blank). To change this entry, choose the Windows Setup icon from the Main Group window. This entry also changes when you change the Display setting in Setup.

fonts.fon= *filename*

Specifies the filename of the proportionately spaced system font used by Windows for Workgroups 3.1. The default is none (blank). To change this entry, choose the Windows Setup icon from the Main Group window. This entry also changes when you change the Display setting in Setup.

keyboard.drv= *filename*

Specifies the filename of the keyboard driver you are using. The default is none (blank). To change this entry, choose the Windows Setup icon from the Main Group window and change the setting for the Keyboard option.

language.dll= *library-name*

Specifies the name of the dynamic-link library that supplies language-specific functions. If no language library is specified, Windows for Workgroups uses the built-in U.S. English library. The default is none (blank). To change this entry, choose the International icon from Control Panel and change the setting for the Language option.

mouse.drv= filename

Specifies the filename of the mouse driver you are using. The default is none (blank). To change this entry, choose the Windows Setup icon from the Main Group window and change the setting for the Mouse option.

network.drv= filename

Specifies the filename of the Winnet network driver you are using. The default driver filename specified by Setup for Windows for Workgroups is **wfwnet.drv**.

oemfonts.fon= filename

Supplies the name of the font file for the OEM character set, which is associated with the Display setting. The default is none (blank). To change this entry, exit Windows for Workgroups and run Setup from MS-DOS. This entry also changes when you change the Codepage or Display setting in Setup.

shell= filename

Specifies the Windows shell program that runs when you start Windows for Workgroups. Setup initializes this value as **progman.exe**. Unless you change this entry, Program Manager runs when you start Windows for Workgroups. The default is none (blank). To change this entry, you must edit SYSTEM.INI.

sound.drv= filename

Specifies the filename of the system sound driver you are using. The default is none (blank). To change this entry, you must edit SYSTEM.INI.

system.drv= filename

Specifies the filename of the system hardware driver you are using. The default is none (blank). To change this entry, exit Windows for Workgroups and run Setup from MS-DOS, then choose a new System setting.

TaskMan.Exe= filename

Specifies the task switching application that appears when you press CTRL+ESC. The default is **taskman.exe**. To change this entry, you must edit SYSTEM.INI.

[boot.description]

The **[boot.description]** section is a list of strings that are used to describe the devices you can change when you run Setup. There is no reason to change these entries. If you do change values, you cannot use Windows Setup to update the drivers to newer versions.

[drivers]

This section contains a list of aliases (or names) assigned to installable driver files. The **[drivers]** section can contain this entry:

alias= driver-filename [parameters]

Assigns an alias name to an installable driver and specifies any parameters that are used by the driver. This entry is associated with the **drivers=** entry in the **[boot]** section of SYSTEM.INI. If the driver includes parameters, you must assign an alias and then specify the driver by the alias name in the **[boot]** section. You can define multiple aliases by specifying multiple alias lines in this entry. The default is none (blank). Most Setup programs for Windows-based applications add these settings to SYSTEM.INI when installing drivers, so you shouldn't need to change the values. To change this entry, choose the Drivers icon in Control Panel or use the installation program provided by the manufacturer.



[ClipShares]

The **[ClipShares]** section is used by the ClipBook Viewer to identify the names of ClipBook pages that have been shared for use by other Windows for Workgroups workstations. The ClipBook viewer should be used to change the information present in this section—this information should not be edited by hand.

Note The format and location of information present in the **[ClipShares]** section is subject to change in a future version of Windows. Modifying or creating ClipBook shares should be done only through the ClipBook Viewer.



[DDEShares]

The **[DDEShares]** section defines the DDE shares database and is used to identify the names of DDE shares that can participate in Network DDE conversations.

Note Currently the DDE shares database is located in the **[DDEShares]** section of the SYSTEM.INI file. However, the location and format used by the DDE shares database may change in a future version of Windows. The information present in this section should not be edited by hand or directly by an application, the API calls present in the Network DDE API library should be used to manipulate the DDE shares database.

[keyboard]

The **[keyboard]** section provides information about the keyboard.

Important All entries in this section are required. If you modify or delete one of these entries, Windows for Workgroups does not operate properly. There are no built-in default values for these entries. Setup assigns values based on the system configuration.

The **[keyboard]** section can contain the following entries:

keyboard.dll= filename

Specifies the name of a dynamic-link library that defines the layout for non-U.S. keyboards and keyboards that are not compatible with IBM-compatible systems. This entry is required for all keyboards except these U.S. keyboards: IBM XT™, PC/AT, or enhanced; AT&T type 301 or 302; and Olivetti 83-key. To change this entry, choose the Windows Setup icon from the Main Group window or the International icon from Control Panel.

oemansi.bin= filename

Specifies the name of a file that defines OEM/ANSI code-page translation tables for systems not using code page 437 (the U.S. OEM character set). To change this entry, exit Windows for Workgroups and run Setup from MS-DOS. This setting also changes when you change the Codepage or Display settings.

subtype= number

For some drivers, distinguishes the special features of keyboards that would otherwise have identical layouts. This value can also be used by other drivers. To change the **subtype=** entry, choose the Windows Setup icon from the Main Group window. See the **type=** entry for information about the type values.

These values are defined for **subtype=**:

Type value	Subtype value	Meaning
1	2	Olivetti M24 83-key or AT&T 6300 type 301 83-key.
1	4	AT&T type 302 sometimes used on the 6300 Plus.
2	1	Olivetti 102-key ICO used on M24 systems.

type= *number*

Specifies the keyboard type as one of these values.

Value	Meaning
1	IBM PC or XT compatible (83 keys)
2	Olivetti 102-key ICO
3	IBM AT compatible (84 or 86 keys)
4	IBM compatible, enhanced (101 or 102 keys)

If this entry is blank or missing, the driver selects a default type. For IBM-compatible keyboards (using the KEYBOARD.DRV driver), the default type is determined by the BIOS. To change this entry, choose the Windows Setup icon from the Main Group window.

[mci]

The **[mci]** section contains a list of the drivers that use the Media Control Interface (MCI) to play media files. These drivers are installed automatically when you run Setup. To change these values, choose the Drivers icon in Control Panel.

[NonWindowsApp]

The **[NonWindowsApp]** section contains entries that affect the performance of MS-DOS-based applications. This section can contain the following entries:

CommandEnvSize= *bytes*

Specifies the size of the COMMAND.COM environment. Since running batch files with the extension .BAT starts COMMAND.COM, this setting also applies to batch files. The value for this setting must be either 0 or between 160 and 32768. A value of 0 disables this setting. If this value is either too small or too large, the value is rounded up to 160 or down to 32768. If the value is less than the current size of the actual environment, this setting is disabled, as if it were set to 0. If you specified the environment size in a PIF file for COMMAND.COM, the PIF setting overrides this setting. The default is 0 with MS-DOS versions earlier than 3.2. Otherwise, the default value is the /e: option in the shell= command line in CONFIG.SYS. To change this value, you must edit SYSTEM.INI.

DisablePositionSave= Boolean

When this entry is **0**, the position and fonts used in an MS-DOS-based application are saved in the DOSAPP.INI file when you quit the application. If this entry is **1**, any settings that weren't saved previously in DOSAPP.INI are not saved. If this entry is **1**, the setting can be overridden for each MS-DOS-based application by selecting the Save Settings On Exit check box in the Fonts dialog box. The default is **0**. To change this entry, you must edit SYSTEM.INI.

FontChangeEnable= Boolean

If **1**, this entry provides the ability to change the fonts when running MS-DOS-based applications in a window on a system that uses version 3.0 grabbers (usually version 3.0 display drivers). Windows for Workgroups version 3.1 video grabbers (used in version 3.1 display drivers) include built-in support for changing fonts when running MS-DOS-based applications in a window. If you are using a 3.0 grabber that has not been updated to include the ability to change fonts and you want to use this feature, set this value to **1**. When this entry is **1**, your screen may lose characters and the cursor may change size and position slightly. The default is **1** on systems that use Windows for Workgroups 3.1 grabbers, and **0** on systems that use Windows 3.0 grabbers. To change this entry, you must edit SYSTEM.INI.

GlobalHeapSize= kilobytes

Specifies the size of a buffer in conventional memory that Windows for Workgroups allocates when running in standard mode. This buffer can be used to share information among all MS-DOS-based applications that are started from Windows for Workgroups. This setting has no effect when Windows for Workgroups is running in 386 enhanced mode. The default is **0**. To change this entry, you must edit SYSTEM.INI. (You should not need to change this setting.)

LocalTSRs= list-of-TSR-applications

Specifies which terminate-and-stay-resident (TSR) programs work properly if they are copied to each instance of a virtual machine. When started, Windows for Workgroups detects any TSR programs that are currently running. If the TSR is on the **LocalTSRs=** list, Windows for Workgroups copies the TSR in each virtual machine you run. Many TSRs do not run properly if they are added to this list. Make sure your TSR is fully compatible with Windows for Workgroups and can be copied to a virtual machine before adding it to the list. The default is **dosedit, ced**. To change this value, you must edit SYSTEM.INI.

MouseInDosBox= Boolean

Specifies whether the mouse is supported when running an MS-DOS-based application in a window. Mouse support for MS-DOS-based applications running in a window is automatically available if you are using a Windows version 3.1 grabber. Set this entry to **1** if you are using a Windows version 3.0 grabber and you want mouse support. If you do not want mouse support, disable this setting. The default is **1** if an MS-DOS mouse driver is loaded and has the extension .COM or .SYS, and supports using a mouse with an MS-DOS-based application. If not, the default is **0**. To change this entry, you must edit SYSTEM.INI.

NetAsynchSwitching= 0 | 1

Specifies whether you can switch away from an application running in Windows for Workgroups standard mode after it has made an asynchronous NetBIOS call. The default value of **0** specifies that such task switching is not possible. Switching away from some applications that make these calls might cause the system to fail. Once Windows detects an asynchronous NetBIOS call, it will not allow switching away from the application even if no more of these calls are made. Set this value to **1** if you are sure the applications you use will not receive network messages while you are switched away from them. To change this entry, you must edit SYSTEM.INI.

ScreenLines= number

Specifies the number of lines that are displayed on the screen when you run an MS-DOS-based application. An application that specifies a different screen mode can override this entry. The default is **25**. To change this entry, you must edit SYSTEM.INI.

SwapDisk= drive:directory

Provides the name of the disk drive and directory to which Windows for Workgroups in standard mode swaps MS-DOS-based applications. The default is the directory that TEMP environment variable points to. If there is no TEMP variable, the default is the root directory of your first hard drive (usually C:). If your system does not use a hard disk, the default is the root directory of the first floppy drive (usually A:). To change this entry, you must edit SYSTEM.INI.

**[network]**

This section contains settings that affect how your computer interacts with the network. To change the values of the settings described in the **[network]** section, use the Network option in Control Panel, except where specified.

The **[network]** section can contain the following entries:



AutoLogon= *Boolean*

Specifies whether you are automatically logged on when you start Windows for Workgroups. If you are using a logon password, the Windows For Workgroups Logon dialog box appears and asks you to enter your logon password. To log on, you must supply your password. If you have a blank logon password, you will be logged on and no prompt appears. If set to **No**, you are not automatically logged on at startup. The default is **Yes**. To make changes to this entry, choose the Network icon in Control Panel and select the On Startup check box for Network Logon.



AutoStart= *Full | Basic | Popup | Netbind | Netbeui | Workstation [, ...]*

Specifies the additional real-mode components that are loaded when the **net start** command is issued from the command prompt or in AUTOEXEC.BAT. The **Full** value is used to load full redirector support (unless otherwise specified, **Full** is the default value for the redirector support). The **Basic** value is used to load basic redirector support. The **Popup** value is used to load the popup interface. The **Netbind** value is used to bind protocols and network adapter drivers. The **Netbeui** value is used to load the NetBEUI protocol. The **Workstation** value is used to load the default redirector. If you want to specify more than one real-mode component, separate each value by a comma. By default, the **net start** command will perform only the **net bind** functionality. To change this entry, you must edit SYSTEM.INI.



CacheThisPassword= *Boolean*

Determines whether the Save This Password In Your Password List check box in the Enter Password or LAN Manager Log On dialog box is selected by default the next time the dialog box is displayed. If this entry is **Yes**, the check box is selected. If this entry is **No**, the check box is not selected. When you select or clear the check box, this value is changed in the SYSTEM.INI file. The default is **No**, or the last value specified by the user.



comment= *string*

Provides a description of your computer (maximum of 48 characters) that appears to other users when they are browsing the workgroup—the string value specified for this entry cannot contain commas. This description appears next to your computer name in the Connect Network Drive, Connect Network Printer, and Select Computer dialog boxes. The default is none (blank).

**ComputerName=** *name*

Specifies the name of your computer on the network. Your computer name must be unique, can a maximum of 15 characters long, and can contain letters, numbers, and any of the following characters:

! # \$ % & () - . @ ^ _ ' { } ~

Your computer name appears in the Connect Network Drive, Connect Network Printers, and Select Computer dialog boxes when browsing through workgroups for computers, directories or printers. The default is the name provided for your computer during Setup.

**EnableSharing=** *0 | 1*

Defines the state of the Enable Sharing check box in the Network Settings dialog box accessible from the Network icon in Control Panel. If this entry is **1**, the user selected the Enable Sharing check box and you can share resources. If this entry is **0**, the Enable Sharing check box is not selected and resources will not be shared. The default is **1**. To change this entry, change the state of the Enable Sharing check box present in the Network Settings dialog box, which can be accessed from the Network icon in Control Panel. When sharing is disabled by deselecting the Enable Sharing check box, the VSERVER.386 and VBROWSE.386 virtual device drivers are removed from the **network=** entry in the **[386Enh]** section of SYSTEM.INI.

**Exclude=** *lana number [, ...]*

Indicates that Windows for Workgroups should not use the protocols represented by the LANA numbers assigned to this setting. This setting takes precedence over the **LANAs=** or **V86ModeLANAs=** entries, so that a lana number listed on this line will never be used even if it is also listed in those entries. This setting should be used when a NetBIOS protocol is present but cannot be used by the Windows for Workgroups network components, for example, the NetWare NETBIOS.EXE TSR. The default is none (blank). To change this entry, you must edit SYSTEM.INI.

**KeepConn=** *seconds*

Specifies the number of seconds Windows for Workgroups is to wait before disconnecting an implicit connection that is no longer being used. This is useful if you are performing several directory searches or lists using an implicit connection, or several tasks that involve pipes. If the applications that you are using to perform these tasks run slowly, then increase this value. The default is **600**. (You should not need to change this setting.)



LANAs= *lana number [, ...]*

Limits the automatic detection of protocols to the specified list of LANA numbers if you start the network before starting Windows for Workgroups, and specifies that Windows for Workgroups is to use only those protocols present that are assigned LANA numbers included in this setting. This setting is useful with NetBIOS protocols which incorrectly identify themselves as being active on LANA numbers others than those assigned to them, for example, the NetWare NETBIOS.EXE TSR. The default is none (blank). To change this entry, you must edit SYSTEM.INI.



LMAnnounce= *Boolean*

If this entry is **Yes**, your computer announces its presence to LAN Manager computers in your workgroup. If computers in your workgroup that are running LAN Manager need to see your computer when they browse the network, set this value to **Yes**. The default is **No**, which tells the Windows for Workgroups server not to broadcast its presence to LAN Manager clients, and minimizes the level of network traffic. To change this entry, you must edit SYSTEM.INI.



LMLogon= *0 | 1*

Specifies whether the LAN Manager Logon dialog box is shown to permit you to log on to a LAN Manager domain when you start Windows for Workgroups. If this entry is **1**, and you have stored the domain password in your password list, Windows for Workgroups logs you on to a LAN Manager domain and runs your logon script, if you have one. If the domain password is not in your password-list, the LAN Manager Logon dialog box appears so that you can pick the domain you want to log on to and specify the password for that domain. The default value is **0**, indicating that you will not log onto a domain. To change this entry, change the state of the Log On to LAN Manager Domain check box in the LAN Manager Settings dialog box.



LogonDomain= *domain name*

Specifies the name of the default LAN Manager domain (workgroup) that validates your password, if you choose to logon to a LAN Manager domain when you start Windows for Workgroups (as specified by the **LMLogon=** entry). The default is the designated workgroup for your computer.

**LogonValidated= Boolean**

This entry indicates whether your logon was validated by a LAN Manager server when you last logged on. If you decide to change the **LMLogon=** value while you are still logged on by selecting or clearing the Log On To LAN Manager Domain check box in Control Panel, the **LogonValidated=** entry ensures that you are logged off properly when you end your Windows for Workgroups session, or if you choose to log off using Control Panel. The default is **No**. (You should not need to change this setting.)

**MaintainServerList= Yes | No | Auto**

Specifies whether your computer will maintain the list of computers in your workgroup and the names of workgroups defined on your network. If this entry is **Yes**, your computer will be used to maintain the list of computers in your workgroup and workgroups on your network, and increases the likelihood that your computer will become the "master" computer in your workgroup which maintains these lists. If this entry is **No**, your computer is never used to maintain these lists—use this setting if your computer has very little free memory, is connected to the rest of your workgroup only by a slow link (such as a telephone-line connection), or other special conditions apply which would cause performance problems. If this entry is **Auto**, your computer will maintain these lists when Windows for Workgroups determines that it is necessary. The default is **Auto**. (You should not need to change this value.) To change this entry, you must edit SYSTEM.INI. At least one computer in your workgroup must have a value of either **Auto** (the default) or **Yes** for this entry in order to ensure that the list of workgroups and computers on your network is available.

**Multinet= networkname [, ...]**

Specifies the name of one or more active secondary networks that you have added support for in Windows for Workgroups. The default is none (blank). You can change this entry using the Network icon in the Control Panel.

**NoSharingControl= 0 | 1**

Specifies whether the Enable Sharing section of the Network Settings dialog box, accessible from the Network icon in Control Panel, is enabled or disabled. If the value for this entry is **1**, the Enable Sharing section will be disabled and the user will not be able to enable or disable sharing. If the value for this entry is **0**, the Enable Sharing section is enabled and the user will be able to change the state of sharing resources (that is, the user can turn it on or off). The default is **0** (the user can change the state for sharing resources).

**NumBigBuf= *number***

Specifies the number of buffers used by the protected-mode redirector VREDIR.386 to cache data read from the network. Increasing this entry can improve the performance of network operations, but reduces the amount of memory available for applications. This entry is only used if the network is not started from MS-DOS before starting Windows for Workgroups, and Windows for Workgroups is running in 386 enhanced mode. By default, the redirector uses one eighth of the available physical memory free at the time when it is loaded, so the number of cache buffers is that amount of memory divided by the size of each buffer (which is 4096 bytes in size), rounded down. This entry has a minimum of 2 and a maximum of 4096.

**PrintBufTime= *seconds***

Specifies the number of seconds of idle printing time that Windows for Workgroups is to wait before indicating that the end of a print job has been reached when printing from an MS-DOS-based application. When printing to a network printer from an MS-DOS-based application, your documents do not start printing until the application finishes processing the print job. If you are using an MS-DOS-based application that processes print jobs quickly and you want your documents to print sooner, decrease this value. If you are using an MS-DOS-based application that takes longer to process print jobs or if your print job is broken into multiple documents, increase this value. The time an MS-DOS-based application is suspended is not counted. The default value is 45. To change this entry, you must edit SYSTEM.INI.

**priority= *number***

Specifies the relative priority value for sharing resources (VSERVER.386). The lower the number, the less time is given to sharing resources. The higher the number, the more time is given to sharing resources. The default priority is 80. To make changes to this entry, choose the Network icon in Control Panel and change the position of the Performance Priority slider bar.

**reconnect= *Boolean***

Defines the default state for the Reconnect At Startup check box in the Connect Network Drive or Connect Network Printer dialog box. The Reconnect At Startup check box is used to determine whether the given network connection is re-established when Windows for Workgroups is re-started. If this entry is **Yes**, the Reconnect At Startup check box in the Connect Network Drive or Connect Network Printer dialog box is selected the next time the dialog box is displayed. If this entry is **No**, the check box is not selected. When you select or clear the check box, the value for this setting changes in the SYSTEM.INI file to reflect its current state. The default is **Yes**.



reshare= *Boolean*

Defines the default state for the Reshare At Startup check box in the Share Directory or Share Printer dialog box. The Reshare At Startup check box determines whether the given share is automatically re-shared at startup. If this entry is **Yes**, the Reshare At Startup check box in the Share Directory or Share Printer dialog box is selected the next time the dialog box is displayed. If this entry is **No**, the check box is not selected. When you select or clear the check box, this value is changed in the SYSTEM.INI file to reflect its current state. The default is **Yes**.



username= *name*

Specifies the default logon name (maximum of 20 characters) that is used to log on to Windows for Workgroups. The value for this setting changes to the logon name you specify when you log on to Windows for Workgroups for the first time. The default is your computer name until you log on for the first time. Then the default value becomes the logon name you specify in the Welcome To Windows For Workgroups dialog box.



WorkGroup= *name*

Specifies the workgroup (maximum of 15 characters) that your computer belongs to. You can change this entry using the Network icon in the Control Panel.



[PasswordLists]

This section contains settings that specify the location of the password-list files for each user who logs on to your computer. Each password-list file contains a list of the passwords you can use to connect to password protected resources. Each entry in the **[PasswordLists]** section has this format:

username= drive:directory\PasswordListFilename

Value	Definition
<i>username</i>	The logon name of the user.
<i>drive</i>	The drive that contains the password-list file.
<i>directory</i>	The name of the directory where the password-list file is located.
<i>PasswordListFilename</i>	Specifies the name of the password-list file.

[standard]

The **[standard]** section contains entries that are specific to running Windows for Workgroups in standard mode. This section can contain the following entries:

FasterModeSwitch= 0 | 1

If set to **1**, Windows for Workgroups in standard mode uses a faster method of switching from protected to real mode on many 80286-based computers. When this entry is **1**, Windows for Workgroups responds more quickly to hardware interrupts, allowing better throughput for interrupt-intensive applications, such as communications applications. In addition, set this entry to **1** if you are using a Zenith Z-248 system and are losing characters while typing, or if you are using an Olivetti M-250-E and lose mouse functionality. The default is **0**. To change this entry, you must edit SYSTEM.INI.

Note The value for **FasterModeSwitch** has no effect on 80386-based computers. Some early IBM AT and compatible computers do not have the BIOS support necessary to use this setting. Setting this entry to **1** on these computers can cause them to hang when starting Windows for Workgroups.

Int28Filter= number

Specifies the percentage of INT 28h interrupts, which are generated when the system is idle, that are visible (or reflected) to software that is loaded before Windows for Workgroups. Windows reflects every *n*th interrupt, where *n* is the value of this entry. For example, a value of **1** reflects every INT 28h interrupt, a value of **2** reflects every second INT 28h interrupt, a value of **3** reflects every third INT 28h interrupt, and so on. Increasing this value might improve Windows for Workgroups performance, but may interfere with some memory-resident programs such as a network software.

Set this value to **0** to prevent INT 28h interrupts from being reflected. Setting this value too low (from 1 to 9), adds to system overhead and interferes with communications applications. The default is **10**. To change this entry, you must edit SYSTEM.INI.

MouseSyncTime= milliseconds

Specifies the number of milliseconds that can elapse between mouse data bytes before Windows for Workgroups in standard mode assumes that a mouse data packet is complete. This setting only affects Windows for Workgroups standard mode on computers with an IBM PS/2 mouse interface. The default is **500**. To change this value, you must edit SYSTEM.INI.

NetHeapSize= *kilobytes*

Specifies the size of the data transfer buffers that Windows for Workgroups in standard mode allocates in conventional memory for transferring data over a network. If an application is not running correctly, your network may require a larger buffer than the default value. Increasing this value decreases the amount of memory available to applications. If network software is not running, this entry is ignored and no memory is allocated. The default is **20**. To change this entry, you must edit SYSTEM.INI.

PadCodeSegments= *0 | 1*

If this entry is **1**, Windows Kernel pads code segments with 16 bytes. This prevents the last instruction in the segment from being too close to the segment limit for 80286 C2 stepping. Set this value to **1** only for 80286 stepping. Unfortunately, there is no easy way of tell what is required for stepping a 80286 chip. Try setting this value to **1** if your 80286 system hangs in standard mode. The default is **0**. To change this entry, you must edit SYSTEM.INI.

Stacks= *number*

Specifies the number of interrupt reflector stacks that are used by the standard mode MS-DOS Extender (DOSX) to map an MS-DOS or BIOS API from real mode to protected mode. If you receive a "Standard Mode: Stack Overflow" message, try increasing this number. You can specify a number between **8** and **64**. The default is **12**. To change this entry, you must edit SYSTEM.INI.

StackSize= *kilobytes*

Specifies the size of the interrupt reflector stacks used by the standard mode MS-DOS Extender. The default is **384**. To change this entry, you must edit SYSTEM.INI. (You should not need to change this entry.)

PROTOCOL.INI: Network Protocol Initialization File



The Setup program builds a file called PROTOCOL.INI in the WINDOWS directory which defines the parameters used by the protocol and network adapter drivers that Windows for Workgroups supports. PROTOCOL.INI is created and modified by Setup from information present in the NETWORK.INF information file or from an OEMSETUP.INF file (if used). A complete description of the parameters supported by each of the network adapters drivers that are provided with the Windows for Workgroups product is included in Appendix B—this information may be modified using the options available from the Network icon in Control Panel.

Important Do not modify the PROTOCOL.INI file by hand, unless absolutely necessary. Windows for Workgroups relies on the format and configuration information present in PROTOCOL.INI to execute and to install other network components—inadvertent changes to PROTOCOL.INI may damage the integrity of the Windows for Workgroups environment. To update PROTOCOL.INI, you can change the entry without editing the PROTOCOL.INI file using the Network icon on Control Panel.

The Protocol Manager (PROTMAN.DOS) passes device drivers the information from the PROTOCOL.INI file. PROTOCOL.INI also contains network adapter configuration information, such as the input/output (I/O) address, direct memory access (DMA), and interrupts. The PROTOCOL.INI file contains sections for **[network.setup]**, **[protman]**, and separate sections for each configured network adapter and network protocol.

When multiple network cards are used, some entries, such as interrupt settings and shared memory addresses, may need fine tuning or adjustments to avoid hardware conflicts. Since the Setup program cannot anticipate every possible conflict, watch for error messages when you start the computer after installation. For example, if a network adapter card and a video controller card both try to use the same memory address, you must adjust one of the cards to a different address, either via the software or the switches on the card (or both, which is necessary in most cases). The PROTOCOL.INI entries must agree with the jumper setting on each card. The network adapter card settings may be changed using the options available from the Network icon in the Control Panel—do not edit the PROTOCOL.INI file directly.

[network.setup]

The **[network.setup]** section provides information from Setup for the network installation. The **[network.setup]** entries are shown in the following example.

```
[network.setup]
version= 0x3100
netcard= ms$ee16, 1, MS$EE16
transport= ms$netbeui, MS$NETBEUI
lana0= ms$ee16, 1, ms$netbeui
```

Entry	Definition
version=	Current network software version. The default version 0x3100 is assigned by Setup.
netcard=	The network adapter driver name for each card that is used in the network as defined in NETWORK.INF.
transport=	The name of the network transport driver protocol for Windows for Workgroups.
lana#=	Identifies the binding between the network adapter and the network protocol, as configured by Setup.

[protman]

Provides the settings for the Protocol Manager. This is the format for this section:

Entry	Definition
drivername=	The PROTMAN\$ entry defines the driver name for the Protocol Manager.
priority=	The priority entry determines the order in which incoming frames are processed. If used, the highest priority is given to the first protocol stack, MS\$NETBEUI.

Here is an example of entries in this section for a Windows for Workgroups workstation configured with NetBEUI:

```
[protman]
drivername= PROTMAN$
priority= MS$NETBEUI
```

[netcard]

Lists the set of parameters for an NDIS network adapter. A **[netcard]** section is present for each network adapter card configured in the computer, and the specific entries present in this section will vary depending on the network adapter card installed.

Here is an example of entries in this section for the Intel EtherExpress 16 or 16TP:

```
[ms$ee16]
drivename= EXP16$
IOADDRESS=0x300
IRQ=5
IOCHRDY=Late
TRANSCIEVER=Thin Net (BNC/COAX)
```

[protocol]

Defines the settings used by a network protocol. A *[protocol]* section is present for each network transport protocol installed on the computer, and the specific entries present in this section will vary depending on the protocol installed.

This is the format for entries common to each configured protocol:

Entry	Definition
bindings=	The entry indicates to which network adapter card drivers each transport binds. The netcard name for the network driver and protocol must appear in the bindings= entry for at least one of the transport drivers. The bindings= entry may specify one or more <i>[netcard]</i> sections (separated by commas).
lanabase=	The entry to define the first LANA number the protocol is to accept. Refer to <i>lanan#</i> in the [network.setup] section.

Here is an example of the entries present in this section for the Microsoft NetBEUI protocol:

```
[MS$NETBEUI]
drivename= NETBEUI$
sessions= 10
NCBS= 32
bindings= MS$EE16
lanabase= 0
```

PROGMAN.INI: Program Manager Initialization File

The Program Manager initialization file, PROGMAN.INI, usually has entries for **[settings]** and **[groups]** sections, which describe what is to appear in the Program Manager window when you run Program Manager. To change the appearance and behavior of Program Manager, use the Program Manager menu commands.

A system administrator might also choose to add a third section, **[restrictions]**, for custom installations. For more information about creating custom entries for PROGMAN.INI for multiple systems during Windows for Workgroups installation, see “Configuring Program Manager for a Network Workgroup” in Chapter 9, “Tips for Configuring Windows for Workgroups.”

[settings]

The **[settings]** section usually has these entries:

```
[Settings]
Window= 60 28 565 388 1
SaveSettings= 1
MinOnRun= 0
AutoArrange= 1
Startup=
Order= 8 7 2 3 5 1 6
```

Entry	Definition
Window=	Four numbers that indicate the position of the window when Program Manager is opened, followed by a 1 if the window is maximized.
SaveSettings=	1 if the Save Settings On Exit command on the Options menu is checked in Program Manager. In this case, Program Manager saves the current configuration when you close Windows for Workgroups.
MinOnRun=	1 if the Minimize On Use command is checked on the Options menu in Program Manager. In this case, the Program Manager is shown as an icon when you run another application.
AutoArrange=	1 if the AutoArrange command is checked on the Options menu in Program Manager. In this case, the icons in each group are automatically arranged when you run Program Manager.
Startup=	Name of the group that serves as the “startup” group. If this entry is blank, the Startup group created in Windows Setup is the startup group.
Order=	A list of numbers separated with spaces, indicating the order in which the groups are drawn in the window.

[groups]

The PROGMAN.INI file also has a section named **[groups]**, which has entries such as these:

```
[Groups]
Group1= C:\WINDOWS\ACCESSOR.GRP
Group2= C:\WINDOWS\GAMES.GRP
Group3= C:\WINDOWS\ALDUS.GRP
Group5= C:\WINDOWS\WORDFORW.GRP
Group6= C:\WINDOWS\MAIN.GRP
Group7= C:\WINDOWS\STARTUP.GRP
Group8= C:\WINDOWS\APPLICAT.GRP
```

Entry	Definition
Groupx=	A filename (with full path) for the .GRP file that contains the group created during Setup, or created when an application was installed, or that you created in Program Manager.

To change these settings, we recommend that you use the commands on the Program Manager menu.

[restrictions]

For network system administration, you can also add a **[restrictions]** section to PROGMAN.INI to restrict user actions. The **[restrictions]** section can have these entries:

```
[restrictions]
NoRun=
NoClose=
NoSaveSettings=
NoFileMenu=
EditLevel=
```

Entry	Definition
NoRun=	1 disables the Run command on the File menu. The Run command appears dimmed on the File menu and the user cannot run applications from Program Manager unless the applications are set up as icons in a group.
NoClose=	1 disables the Exit Windows command on the File menu. Users cannot quit Program Manager through the File Menu or the Control menu (the Exit Windows and Close commands are dimmed), or by using ALT+F4.
NoSaveSettings=	1 disables the Save Settings on Exit command on the Options menu. The Save Settings command appears dimmed on the Options menu and any changes that the user makes to the arrangement of windows and icons are not saved when Windows is restarted. This setting overrides the SaveSettings= entry in the [settings] section of the PROGMAN.INI file.
NoFileMenu=	1 removes the File menu from Program Manager. All of the commands on that menu are unavailable. Users can start the applications in groups by selecting them and pressing ENTER, or by double-clicking the icon. Unless you have also disabled the Exit Windows command, users can still quit Windows for Workgroups by using the Control menu or ALT+F4.
EditLevel= <i>n</i>	<p>Sets restrictions for what users can modify in Program Manager. You can specify one of the following values for <i>n</i>:</p> <ul style="list-style-type: none"> 0 allows the user to make any change. (This is the default.) 1 prevents the user from creating, deleting, or renaming groups. If you specify this value, the New, Move, Copy, and Delete commands on the File menu are not available when a group is selected. 2 sets all restrictions in EditLevel= 1, plus prevents the user from creating or deleting program items. If you specify this value, the New, Move, Copy, and Delete commands on the File menu are not available. 3 sets all restrictions in EditLevel= 2, plus prevents the user from changing command lines for program items. If you specify this value, the text in the Command Line box in the Properties dialog box cannot be changed. 4 sets all restrictions in EditLevel= 3, plus prevents the user from changing any program item information. If you specify this value, none of the areas in the Properties dialog box can be modified. The user can view the dialog box, but all of the areas are dimmed.

To enable any of the commands or remove any of the **EditLevel=** restrictions, either remove the entry from the PROGMAN.INI file, or set the value to 0.

CONTROL.INI: Control Panel Initialization File

The CONTROL.INI file contains several sections that specify the options you set by choosing icons in Control Panel.

Section	Purpose
[current]	Specifies the current color scheme.
[Color Schemes]	Defines the colors for each element of specific color schemes as set by choosing the Color icon.
[Custom Colors]	Defines the custom colors in the color palette as set by choosing the Color icon.
[patterns]	Defines the color values for the bitmap patterns as set by choosing the Desktop icon.
[MMCPL]	Specifies values related to the multimedia items in Control Panel.
[Screen Saver.*]	Specifies the density, warp speed, and password protection values for the screensaver.
[userinstallable.drivers]	Specifies values related to installable drivers used for multimedia.
[drivers.desc]	Specifies MIDI Mapper and Time, the multimedia control devices.
[installed]	Specifies the current version of Microsoft Windows for Workgroups and the installed printers.

To change any of these items, choose the related icons in Control Panel.

WINFILE.INI: File Manager Initialization File

The WINFILE.INI file contains several sections that specify the options used by File Manager.

Section	Purpose
[Settings]	Defines the default settings used by File Manager.
[Restrictions]	Defines the settings for restricting user access to File Manager functions.

[Settings]

This section specifies the options you can set by choosing menu commands in File Manager. For example, these settings might appear in WINFILE.INI:

```
[Settings]
Window= 0, 6, 800, 600,,, 1
dir1= 48, 7, 543, 236, -1, -1, 3, 30, 201, 1808, 150, B:\*. *
Face= MS Sans Serif
Size= 8
LowerCase= 1
ConfirmDelete= 0
ConfirmSubDel= 0
ConfirmReplace= 1
ConfirmMouse= 1
ConfirmFormat= 1
```

Entry	Definition
window=	The size and position of the window and whether it is maximized when opened.
dir1=	The current directory settings.
face=	The name of the typeface used for desktop items. The default is MS Sans Serif .
size=	The point size for the typeface. The default is 8 .
LowerCase=	1 if filenames are to appear in lowercase.
ConfirmDelete=	1 if the user is to be prompted to confirm file deletion requests.
ConfirmSubDel=	1 if the user is to be prompted to confirm subdirectory deletion requests.
ConfirmReplace=	1 if the user is to be prompted to confirm file replacement requests.
ConfirmMouse=	1 if the user is to be prompted to confirm mouse drag-and-drop requests.
ConfirmFormat=	1 if the user is to be prompted to confirm formatting requests.

To change values for these settings, use the menu commands in File Manager.

[Restrictions]

For network system administration, you can also add a **[Restrictions]** section to WINFILE.INI to restrict user actions. The **[Restrictions]** section contains the **NoShareCommands=** entry.



NoShareCommands= 0 | 1

This entry is used to inform File Manager whether the file sharing menu commands should be displayed on the Disk menu. An entry of **1** (non-zero) tells File Manager to not display the Share As and the Stop Sharing menu items on the Disk menu (and the related buttons on the toolbar)—this inhibits the user from being able to share directories on the local workstation. An entry of **0** tells File Manager to display the sharing menu commands. The default is **0**.

MSMAIL.INI: Microsoft Mail Initialization File



The MSMTP.INI file is the Microsoft Mail initialization file that is used by the Mail application provided with Windows for Workgroups. MSMTP.INI can contain the following sections:

Section	Purpose
[Address Book]	Specifies entries used by the Address Book support functions for the Mail program.
[Custom Commands]	Specifies a custom command that can be installed into one of the Mail menus at runtime.
[Custom Message]	Specifies a custom message type that is installed into a Mail menu at runtime.
[Microsoft Mail]	Defines the configuration of the Mail program, and the Microsoft Mail transport and name service.
[MMF]	Affects the automatic compression of the Mail message file.

Many of the entries described in this section of this chapter are assigned built-in, default values, and won't be present in your MSMTP.INI file. To change the appearance and behavior of the Mail application, use the Mail menu commands instead of editing the MSMTP.INI file directly. Some of options that you specify in the Mail application are stored in your mail message file (MMF) instead of the MSMTP.INI.

[Address Book]

Entries present in your MSMTP.INI in this section are used by the Address Book support functions in the Mail program. Most of the entries for this section use default internal values specified by the Mail program, and the **[Address Book]** section may not be present in your MSMTP.INI file. The entries in this section control the default address directory that will be displayed in the Address Book. Do not change any of these entries if they appear.

[Custom Commands]

Each entry in this section specifies a custom command that can be installed into one of the Mail menus at runtime. This section may appear both in the MSMTP.INI in your WINDOWS directory and in SHARED.INI in the directory pointed to by the **SharedExtensionsDir=** entry of MSMTP.INI.

The **[Custom Commands]** section can contain one or more entries in the following format:

```
tag= version;menu;name;position;DLL name, ordinal;command;
    event map;status text;Help file;help context;
```

For example:

```
IC1= 3.0;help;&Out of Office;10;<ExtsDir>BIN-EXT\OOF.DLL;3;;Out of Office Email Notification
```

Value	Definition
<i>tag</i>	The descriptive identifier for the command.
<i>version</i>	The version of Mail that the custom command is compatible with.
<i>menu</i>	Names the menu where the custom command is to be added, such as File or Help.
<i>name</i>	The command name that is to appear on the menu. You may include an ampersand just before the letter that is to serve as an ALT-key accelerator.
<i>position</i>	The zero-relative position within its menu where the command is to appear. A value of -1 places the command at the end of the menu.
<i>DLL name</i>	Name or path of the DLL where the custom command resides. This entry can optionally be followed by a comma and the ordinal of the entry point to call (default is ,1).
<i>command</i>	The command string passed as one of the parameters to the DLL entry point for the command.
<i>event map</i>	A sequence of up to 16 digits identifying the Mail events that the custom command supports. Each can be 0 or 1 to indicate if the DLL is to be called for a specific event. Currently three events are defined; the rest are reserved and must be zero (or, as in the examples above, this whole section can be empty): The first digit means Mail startup. The second digit means Mail exit. The third digit means the arrival of a new message.
<i>status text</i>	Text to be displayed in the Mail status bar when the user cursors to the command in the menu.
<i>Help file</i>	Windows Help file to be invoked when the user presses F1 while the command is selected. The specified filename is passed to the Windows help program. (optional)
<i>help context</i>	Passed to the Windows help program along with the Help file name. Use -1 (Help file index) if there is no specific entry in the Help file for this command. (optional)
	There is one special token that may appear within certain fields:
<ExtsDir>	Expands to the value of SharedExtensionsDir= in MSMAIL.INI. Used to refer to DLLs that are installed in a shared extensions directory on the network. Valid for the DLL name, command, and Help file subfields.

[Custom Messages]

This section is similar in many ways to the **[Custom Commands]** section. Each entry specifies a custom message type that is installed into a Mail menu at runtime. This section can appear both in the MSMAIL.INI in the WINDOWS directory and in SHARED.INI in the directory pointed to by the **SharedExtensionsDir=** entry of MSMAIL.INI.

The [Custom Messages] section can contain one or more entries in the following format:

```
class = version;menu;name;position;DLL name;command;operation map;
      status text;Help file;help context;
```

Value	Definition
<i>class</i>	A string uniquely identifying the message type. Mail places this string in messages, and calls custom message DLLs based on its value.
<i>version</i>	Identifies the version of Mail that the custom message is compatible with.
<i>menu</i>	Names the menu where the custom command for the message type is to be added, such as File or Mail.
<i>name</i>	The command name that is to appear on the menu. As usual, you may include an ampersand just before the letter that is to serve as an ALT-key accelerator.
<i>position</i>	The zero-relative position within its menu at which the command is to appear. A value of -1 places the command at the end of the menu.
<i>DLL name</i>	Name or path of the DLL in which the custom command resides.
<i>command</i>	Command string passed as one of the parameters to the DLL entry point for the command.
<i>operation map</i>	Sequence of up to 16 digits. Each can be 0, 1, or 2. 0 indicates that Mail is to perform its standard operation on the custom message. 1 indicates that the DLL is to be called to handle the operation. 2 indicates that the operation is to be completely disabled. Currently eight operations are defined; the rest are reserved and must be zero. In the following list, 0 is the leftmost digit: <ul style="list-style-type: none"> 0 Compose (menu command defined in this entry) 1 File.Open 2 Mail.Reply 3 Mail.Reply to All 4 Mail.Forward 5 File.Print 6 File.Save as 7 Arrival of new mail
<i>status text</i>	Text to be displayed in the Mail status bar when the user cursors to the command in the menu.
<i>Help file</i>	Windows Help file to be invoked when the user presses F1 while the command is selected. The specified filename is passed to the Windows help program. (optional)
<i>help context</i>	Passed to the Windows help program along with the Help file name. Use -1 (Help file index) if there is no specific entry in the Help file for this command. (optional)
<ExtsDir>	There is one special token that may appear within certain fields: Expands to the value of the SharedExtensionsDir= entry in MSMAIL.INI. Used to refer to DLLs that are installed in a shared extensions directory on the network. Valid for the DLL name, command, and Help file fields.

[Microsoft Mail]

This section is used to define the appearance and behavior of the Mail program.

CheckLatencyInterval= *seconds*

Affects the mail spooler's latency checking, which is intended to prevent spooler background processing from interfering with foreground work. If the entered length of time passes without the spooler having any work to do, the latency algorithm is reinitialized. The default is **30** seconds.

DemosEnabled= *0 | 2*

Specifies whether the Demos menu option is to be displayed in the Help menu. If the value of the entry is **2**, the Demos menu option is displayed in the Help menu. If this entry is **0**, the Demos menu option is not shown. The default is **0**. (The Mail demos are not provided with Windows for Workgroups, and are shipped as part of the *Microsoft Mail and Schedule+ Extensions for Windows for Workgroups* product.)

ExportMmfFile= *filename*

Identifies the path and filename for a MMF file pointing to the last place a mail folder was exported to. This entry is written by the Mail program and is used as a default value that is displayed in the Export Folders dialog box when you select the Export Folders option in the File menu of the Mail program.

FixedFont= *facename, size, 0 | 1, 0 | 1*

Identifies a fixed-pitch font that is used to display the body text of a mail message. This entry has four parts, each separated by a comma: typeface name (not the font file name), point size, flag for bold, and flag for italic. The Change Font option on the View menu toggles between the Normal font and the Fixed font. The default is **Courier New, 9, 0, 0**.

ForceScanInterval= *seconds*

Affects the mail spooler's latency checking, which is intended to prevent the spooler background processing from interfering with foreground work. If the designated length of time passes without the spooler getting an opportunity to do outstanding work, idle time is requested more frequently (every **ScanAgainInterval=** seconds) and eventually idle time is used whenever it can. The default is **300** seconds (5 minutes).

IdleRequiredInterval= *seconds*

Affects the mail spooler's latency checking which is intended to prevent the spooler's background processing from interfering with foreground work. The spooler defers its work temporarily if the system has serviced an interactive request such as a keyboard entry or mouse movement within this interval to avoid starting a transfer when the user is busy. The default is **2** seconds.

LocalMMF= 0 | 1

Specifies the location where the user's mail message file (MMF) is created when the user runs Mail for the first time. If this entry is **1**, the user's MMF file is created locally in the WINDOWS directory rather than on the post office the first time Mail is run. If this entry is **0**, the user's Mail messages are stored in the post office on the server. Together with the **NoServerOptions=** entry, MMF files are prevented from being stored in the post office—this is the recommended configuration for the Mail program provided with Windows for Workgroups. The default is **1**.

Login= mailbox name

This entry identifies the default User Name (up to 10 characters) that is displayed in the Mail Sign In dialog box used to log into the Mail program. If you set both the **Login=** and **Password=** entries, the Login dialog is not displayed when you start Mail, and your mailbox is immediately displayed. If you set just **Login=**, Mail prompts for your password only. The default is none (blank).

MailBeep= filename

Specifies the pathname of a Windows .WAV file to change the sound that Mail uses to notify the user when new mail arrives. This entry is ignored if the Sound Chime option in the Mail Options dialog box is not checked. The default is none (blank)—Mail beeps twice when new mail arrives.

MailTmp= pathname

Set the entry to a directory where Mail can place temporary copies of attached files. When you launch an application by double-clicking on a file attached to a mail message, Mail copies the file out to this directory and invokes the application. The default is the value of the TEMP environment variable. If there is no TEMP variable, the default is the WINDOWS directory.

MAPIHELP= filename

This entry points to the MAPI Help file and is used when the user request help in any of the dialog boxes displayed by the MAPI support functions. This entry is defined when the Mail program is run for the first time. The default value points to the MSMAIL.HLP file in your WINDOWS directory (for example, **C:\WINDOWS\MSMAIL.HLP**).

Multi-Message= 0 | 1

Defines the setting for the Print Multiple Notes on a Page check box in the Print dialog box, which appears when you print messages from Mail. Set this entry to **0** for that option to appear unchecked by default. The default of **1** allows the option to be checked by default. This entry is written out by the Mail program each time the state of the check box is changed when printing a mail message.

netbios= 0 | 1

This entry is used to enable NetBIOS notification of new mail delivery. When NetBIOS notification is used, the Windows for Workgroups workstation that is sending a mail message to another Windows for Workgroups workstation sends a NetBIOS notification message to the destination workstation to tell the Mail program running on that machine that a new mail message was sent to the workstation—the Mail program on the destination workstation can then check the workgroup post office for the new mail message. Set this entry to **1** to enable NetBIOS notification—NetBIOS notification provides quicker response to the arrival of new mail from users on your local post office. Set this entry to **0** to disable NetBIOS notification—if NetBIOS notification is disabled, the Mail client needs to regularly check for the arrival of new mail messages on the post office. The default is **1**.

NewMsgsAtStartup= 0 | 1

Specifies whether Mail is to check for new mail messages in the foreground as soon as the user logs in. Set this entry to **1** to have Mail download new messages as quickly as possible when it is started. If this entry is **0**, Mail will check for new messages in the background (as is normally the case when the Mail application is being used). The default is **0**.

NextOnMoveDelete= 0 | 1 | -1

If this entry is **1**, Mail automatically opens the next message in a folder after you delete or move an open message. If set to **-1**, mail automatically opens the previous message. This facilitates quick scanning through the inbox. If set to **0**, Mail closes the Read Note window once you move or delete the message, and you must Enter or double-click to open the next message. Use **1** for messages sorted in ascending order (in the order received) and **-1** for messages sorted in descending order (most recent message first). The default is **1**.

NormalFont= facename, size, 0 | 1, 0 | 1

Identifies the default font (normally proportionally spaced) that is used to display Mail messages. This entry has four parts, each separated by a comma: typeface name (not the font file name), point size, flag for bold, and flag for italic. The Change Font option in the View menu toggles between the Normal font and the Fixed font. The latter is useful for viewing messages that were created using a fixed-pitch font. Both entries affect only message body text, not the message envelope text or folder lists: Mail uses Helv 10 for that purpose. The default is **Helv, 10, 0, 0**. The **[FontSubstitutes]** section of the WIN.INI file defines the mapping of the Helv font to a font present on the Windows for Workgroups workstation.

NoServerOptions= 0 | 1

If this entry is **1**, the Server button in the Mail Options dialog box is unconditionally disabled. On the Mail server, this button calls up another dialog box that enables the user to relocate the MMF file. Together with the **LocalMMF=** entry, this entry prevents MMF files from being stored on the post office—this is the recommended configuration for the Mail program provided with Windows for Workgroups. The default is **1** (the Server button and dialog box are disabled).

OfflineMessages= pathname

This entry is defined by the Mail transport provider when you choose to store your message file somewhere other than the post office. This entry locates the file quickly (without a File System Browse dialog box) when you start up without connecting to the post office. The entry is removed when you store your message file at the post office. The default is the path specified in the Mail Options Server dialog for a local message file. (If this entry is present in your MSMAIL.INI file, then an error occurred when you told the Mail program to move your MMF file—this entry can be used to determine where your MMF file can be found).

OldStorePath= pathname

Contains the original path to a file that was originally stored in a place other than the post office. This entry is written out temporarily by the Mail transport while you are moving your message file (using the Mail Options Server dialog box). This entry is removed once the move completes successfully and only appears if the system crashes during a move.

Password= password

Use this entry and the **Login=** entry to provide Mail with your account information, without being required to type this information into the Mail Sign In dialog box each time. If there is no password, leave the *password* value for this entry blank—do not omit the entry. Omitting the entry means you want to type your password each time in the Mail Sign In dialog box when you start Mail. The default is none (blank)—you are prompted for your password by the Mail program.

PollingInterval= minutes

Gives the default for the Check for New Mail Every *n* Minutes option in the Mail Options dialog box. The value the user enters for the Check for New Mail Every *n* Minutes option is written to the user's mail message file (MMF)—this value is used to define how often the Mail spooler will check for new mail messages. The default is **10**.

Printer= *printer name, driver name, port*

This is the printer which appears in the Mail Print dialog box and is used by Mail when printing messages. Mail initially copies the value identifying the default printer from your WIN.INI file and saves the value in the MSMAIL.INI file. The default is the default printer you specify using the Windows control panel.

PrintFont= *pointsize*

Identifies the default point size for the defined font used by the Mail program to print Mail messages. The actual font used to print the mail message is dependent on whether the Mail message is set to be displayed using the Fixed Font or the Normal Font. The default point size is 10.

PumpCycleInterval= *seconds*

Permits the spooler to check for new mail more often than once per minute, or to override the polling interval value defined in the user's mail message file. The default is 60 seconds, or the value specified in the Mail Options dialog box (whose granularity is minutes).

ReplyPrefix= *string (enclosed in quotes)*

If this entry is present, Mail distinguishes your comments from the original message when you reply to mail. When you reply, the original message text is copied to the body of the reply message and each line of the original is prefixed with the string specified by this entry—the specified string should be enclosed within double quotation marks (for example, "| "—a vertical bar followed by a space). The default is none (blank).

ScanAgainInterval= *seconds*

Affects the mail spooler latency checking to prevent spooler background processing from interfering unduly with foreground work. When the spooler defers work because of higher priority, interactive tasks, it rechecks the availability of the system at this interval. The default is 2.

ServerPassword= *password*

This entry identifies the password used to connect to the server specified by the **ServerPath=** entry. This entry should be used to specify the password for the file share if the form of the **ServerPath=** entry is specified using the universal naming conventions (UNC) and the server, share, or the path name contains spaces—if this entry is present, the value for the **ServerPath=** entry will be interpreted literally and any spaces present in the value for the entry will be used when dynamically connecting to the workgroup post office. If the **ServerPassword=** entry is used, do not specify a password for the **ServerPath=** entry (the password will be misinterpreted and the resulting path to the workgroup post office will be invalid). The **ServerPassword=** entry is written by the Mail program when connecting for the first time to the workgroup post office and is stored in the MSMAIL.INI file in encrypted format—do not change this entry.

ServerPath= *pathname*

ServerPath= *\\server\share\path password*

ServerPath= *server/share:path*

If this entry is present, Mail searches for the post office in the specified directory. The first form, with a normal pathname, works on all networks—the connection to the file share containing the network post office must be made before running the Mail program. The second form (UNC) works only on Windows for Workgroups, and Microsoft networks and compatibles—if you use the UNC form, Mail connects dynamically (without using a drive) to the file server where the post office resides (if the specified UNC name contains any spaces, the password for the share needs to be specified for the **ServerPassword=** entry). The third form works only on Novell NetWare networks and will connect dynamically using an unused drive letter to the NetWare file server where the post office resides. If no value is specified for the **ServerPath=** entry, Mail will prompt the user for a path to the workgroup post office and will write a new value for this entry.

SharedExtensionsDir= *pathname*

SharedExtensionsDir= *\\server\share\path password*

If this entry is present, Mail searches for shared custom commands and messages in the specified directory. The SHARED.INI file in that directory identifies the shared extensions to load. Additional extensions can be entered in your own MSMAIL.INI. Administrators often provide a common share point for extensions to Mail to simplify updating the extensions and this entry makes it work. The first form with a normal pathname works on all networks. The second form works only on Windows for Workgroups, and Microsoft networks and compatibles. If the second form is used, Mail connects dynamically (without using a drive) to the file server where the shared extensions reside. The default is none (blank).

SharedFolders= 0 | 1

Enables the use of Mail shared folders. If this entry is **1**, the user can use access shared folders. If this entry is **0**, shared folders are unavailable to the user. The default is **1**.

SpoolerBackoffInterval= milliseconds

Specifies the amount of time the mail spooler waits before retrying an operation that has failed due to a transient mail server error condition, such as a locked file. The default is **2000** (two seconds).

SpoolerReconnectInterval= seconds

Specifies the amount of time the mail spooler waits before retrying an operation that has failed due to a fatal mail server error condition, such as a lost network connection. The default is **60** (one minute).

WG= 0 | 1

Specifies whether the version of the Mail executable running on the workstation is the Mail program that is provided with Windows for Workgroups. This entry is used internally by the Mail application and is written by the Mail program when it is executed. The default is **1**.

Window= left right top bottom zoom toolbar statusbar scrollbars

The zero-relative position within its menu where the main Mail window is to appear. This entry consists of eight numbers that govern the display of the main Mail window. This information is written when you exit Mail and the changes you made while Mail was running are lost. The defaults for this entry are the window size and location determined by Windows, zoom state normal (restored), Toolbar on, Status bar on, Scroll bars on.

The format for this entry is as follows:

Value	Definition
<i>left right top bottom</i>	The first four numbers are pixel coordinates for the four sides of the main window: left, right, top, and bottom.
<i>zoom</i>	The zoom value is: <ol style="list-style-type: none"> 1 main window is in a normal (restored) state 2 maximized (zoomed) 3 minimized (by icon)
<i>toolbar statusbar scrollbars</i>	Determine when the tool bar, status bar, and scroll bars are displayed on the main window: <ol style="list-style-type: none"> 0 corresponding bar is not displayed 1 bar is displayed

[MMF]

Most entries in this section affect automatic compression of the Mail message file, which by default have the extension .MMF. When enabled, automatic compression uses idle time on your PC to recover disk space freed by the deleted messages and returns the disk space to the file system. You should not need to change the default values for entries in this section.

Kb_Free_Start_Compress= *kilobytes*

The message store manager background compression algorithm starts when at least this much recoverable space is detected in your message file. Both this entry and **Percent_Free_Start_Compress=** are always active. The first entry to trigger starts the compression. The default is **300**.

Kb_Free_Stop_Compress= *kilobytes*

The message store managers, background compression algorithm stops when there is less than the indicated amount of recoverable space in your message file. This avoids the unnecessary difficulty in trying to recover the last little bit of free space. Both this entry and **Percent_Free_Stop_Compress=** are always active. The first entry to trigger stops the compression. The default is **100**.

No_Compress= *0 | 1*

Specifies whether background MMF compression is to be disabled. Set this entry to **1** to disable background compression of the MMF message store. Set this entry to **0** to enable background compression of the MMF message store. The default is **0** (background compression is enabled).

Percent_Free_Start_Compress= *percent*

Background compression starts when the amount of recoverable space rises above this percentage of the total file size. Both this entry and **Kb_Free_Start_Compress=** are always active. The first one to trigger starts the compression. The default is **10** (ten percent).

Percent_Free_Stop_Compress= *percent*

Background compression stops when the amount of recoverable space falls below this percentage of the total MMF file size. Both this entry and **Kb_Free_Stop_Compress=** are always active. The last one to trigger stops the compression. The default is **5** (five percent).

Secs_Till_Fast_Compress= *seconds*

The background compression algorithm has a fast mode and a slow mode. Background compression begins in the slow mode to avoid impacting system response time. After a number of seconds of system inactivity indicated by this entry, the compression switches to fast mode. Any user activity changes the setting back to slow mode. The default is **600** seconds (ten minutes) of system inactivity.

SHARED.INI: Shared Mail Settings Initialization File



The SHARED.INI file permits the sharing of Mail **[Custom Commands]**, and **[Custom Messages]** with other Windows for Workgroup workstations. The SHARED.INI file and the associated custom commands and custom messages reside on the workgroup post office and are referenced in the MSMAIL.INI file by the **SharedExtensionsDir=** entry in the **[Microsoft Mail]** section. Refer to Chapter 12, "Mail," in this manual for more information concerning the relationship between MSMAIL.INI and SHARED.INI.

SHARED.INI can contain the following sections:

Section	Purpose
[Custom Commands]	Specifies a custom command that can be installed into one of the Mail menus at runtime.
[Custom Messages]	Specifies a custom message type that is installed into a Mail menu at runtime.

Consult the previous section, "MSMAIL.INI: Microsoft Mail Initialization File," for a description of these two sections.

SCHDPLUS.INI: Schedule+ Initialization File



The SCHDPLUS.INI file is used to define settings used by Schedule+. Schedule+ uses the SCHDPLUS.INI file to track basic information about the user's schedule, such as display and general option settings, current window positions, and printer information. The SCHDPLUS.INI file contains the following sections:

Section	Definition
[Microsoft Schedule]	Defines the appearance and behavior of Schedule+.
[Microsoft Schedule+ Appt Books]	Indicates the number and list of other users' Appointment Books that were open when you exited Schedule+.
[Microsoft Schedule+ Archives]	Indicates the number and list of Archive files that were open when you exited Schedule+.

When used, most of these entries are assigned built-in, defaults. You should not need to change the Schedule+ settings. To change the appearance and behavior of Schedule+, use the appropriate Schedule+ menu commands.

[Microsoft Schedule]

This section defines the appearance and behavior of Schedule+.

AppPath= *pathname*

Specifies the location of the Schedule+ program and execution files. Microsoft Mail uses this path to find Schedule+ when you receive a meeting request. The default is the WINDOWS directory.

AppointmentView= *left right top bottom*

Specifies the coordinates for the position of the Appointment Book window on the screen. These four numbers are written by the Schedule+ application when you exit, and are used to restore the window to the last displayed position. These numbers are pixel coordinates for the four sides of the Appointment Book window: left, right, top, and bottom.

ApptBookColor= *colornumber(1-17)*

Specifies the preference setting for the background color of the Appointment Book. The default color is yellow. The color number corresponds (in order) to the colors shown in the Display dialog box available from the Options menu.

ApptBookLinesColor= *colornumber(1-17)*

Specifies the preference setting for the color of the lines in the Appointment Book. The default color is black. The color number corresponds (in order) to the colors shown in the Display dialog box available from the Options menu.

CopyTime= *minutes*

Specifies the time interval that Schedule+ copies your online .CAL file to your local .CAL file (occurs in idle time). The default is that your online .CAL file is copied to your local workstation every **15** minutes.

CreateFileFirstTime= *0 | 1*

Specifies whether an online calendar (.CAL) file should be created for a first-time Schedule+ user. If this entry is **1**, an online calendar (.CAL) file is created the first time a user signs on to Schedule+. If **0**, an online calendar file is not created. The default is **1**.

DefaultPrinter= *printer name, driver name, port*

Indicated the current default printer port and its network path as specified in the Windows Control Panel. This is the default printer Schedule+ uses for printing schedule information.

DefaultPrivateTask= *0 | 1*

Defines the default state of the Private check box. If this entry is **1**, the Private check box for new tasks is checked by default. If this entry is **0**, the Private check box for new tasks is not checked by default. The default is **1**.

DefaultRemindAgain= *0 | 1*

Defines the default state of the Remind Again check box. If this entry is **1**, you are reminded again of your appointments at the requested intervals. If this entry is **0**, you are reminded of your appointment only once. The default is **0**.

DefaultRemindAgainAmount= *timeunits*

Specifies the default number of time units to wait (interval) before reminding you of your appointment again. The default is **5** time units.

DefaultRemindAgainUnits= *minutes | hours | days | weeks | months*

Specifies the type of time units you are using in the **DefaultRemindAgainAmount=** entry. The default is in minutes.

DemosEnabled= 0 | 1

Specifies whether the Demos menu option is to be displayed in the Help menu. If the entry is **1**, the Demos menu option is displayed in Help menu. If this entry is **0**, the Demos menu option is not shown. The default is **0**. (The Schedule+ demos are not provided with Windows for Workgroups, and are shipped as part of the *Microsoft Mail and Schedule+ Extensions for Windows for Workgroups* product.)

ExportNoNotes= 0 | 1

Indicates whether the user chose to export notes. If this entry is **0**, notes are exported. If this entry is **1**, the notes are not exported. The default is **0**.

ExportRange= 0 | 1

Indicates the range of schedule information to be exported. If this entry is **0**, the entire schedule file is exported. If this entry is **1**, a particular range is written out. This entry is written by the driver used to save Schedule+ information into a given file format.

ExportType= 0 | 1

Indicates the current default file type for exporting your schedule. If this entry is **0**, the default file type for exporting your schedule is the Schedule+ format. If the entry is **1**, the file type for export is Text. The default is **0**.

ImportDoNotAddDuplicates= 0 | 1

Indicates whether the user chose to import duplicate appointments. If this entry is **0**, duplicate appointments are imported. If this entry is **1**, you duplicate appointments are not imported. The default is **0**.

ImportDoNotAskAboutConflicts= 0 | 1

Indicates whether the user chose to be asked about conflicting appointments during the import process. If this entry is **0**, you are prompted for each conflicting appointment during the import process—in this case, you are asked whether or not to add each conflicting appointment. A value of **1** indicates that you are not asked about conflicts; they are added automatically. The default is **0**.

ImportType= 0 | 1

Indicates the current default file type for importing a schedule file. If this entry is **0**, the file type for importing your schedule is the Schedule+ format. If the entry is **1**, the file type is the Windows Calendar format. The default is **0**.

LargeFont= 1 | 2

Specifies the preference setting for the font size of text displayed in the Appointment Book and Planner. If this entry is **1**, the font size of text is 10 points. If this entry is **2**, the font is 8 points. The default is **2**.

LocalPath= path

Specifies the location of the last user's local calendar (.CAL) file.

LocalUser= username

Specifies the name of the last user to use the Schedule+ workstation software on this computer.

MainWindow= left right top bottom

Specifies the coordinates for the position of the Schedule+ application window on the screen. These four numbers are written by the Schedule+ application when you exit, and are used to restore the Schedule+ window to the last displayed position. These numbers are pixel coordinates for the four sides of the main window: left, right, top, and bottom.

NoStatusBar= 0 | 1

Indicates the preference setting for displaying the status bar. If this entry is **1**, status bar is not displayed. If set to **0**, the status bar is displayed. Default is **0**.

OtherColor= colornumber (1-17)

Specifies the preference setting for the color of other users' appointments in the Planner. The default color is red. The color number corresponds (in order) to the colors shown in the Display dialog box available from the Options menu.

PageBackgroundColor= colornumber (1-17)

Specifies the preference setting for the background color of the Schedule+ window. The default color is gray. The color number corresponds (in order) to the colors shown in the Display dialog box available from the Options menu.

PlannerColor= colornumber (1-17)

Specifies the preference setting for the background color of the Planner window. The default color is white. The color number corresponds (in order) to the colors shown in the Display dialog box available from the Options menu.

PlannerLinesColor= colornumber (1-17)

Specifies the preference setting for the color of the lines in the Planner. The default color is black. The color number corresponds (in order) to the colors shown in the Display dialog box available from the Options menu.

PollTime= *centiseconds*

Specifies the frequency for updating the screen. The default time is 6000 centiseconds.

ReminderPollTime= *minutes*

Specifies the frequency for polling the server for alarm changes. The default is 15.

RequestSummary= *left right top bottom*

Specifies the coordinates for the position of the Messages window on the screen. These four numbers are written by the Schedule+ application when you exit, and are used to restore the Messages window to the last displayed position. These numbers are pixel coordinates for the four sides of the Message window: left, right, top, and bottom.

ShowActiveTasks= *0 | 1*

Indicates whether the Task list is showing all tasks or only active tasks, as specified from the Tasks menu. If only active tasks are displayed, this value is 1. The default is 0, indicating that all tasks are displayed.

StartupOffline= *0 | 1*

Specifies whether Schedule+ should start up using the offline scheduling information, or whether the online schedule should be used. If this entry is 1, Schedule+ is started offline. The default is 0, indicating that Schedule+ is started online.

TaskSortOrder= *0 | 1 | 2*

Specifies the current sort order for tasks. If this entry is 0, tasks are sorted by priority, if 1 the tasks are sorted by due date. If 2, the tasks are sorted by description. The default is 0.

TaskSortSecond= *0 | 1 | 2*

Specifies the secondary sort order for tasks. If this entry is 0, the second sort order is by priority, if 1 the second sort is by due date. If 2, the second sort is by description. The default is 0.

UpdatePostOfficeTime= *centiseconds*

Specifies the frequency for updating the post office on the server after a change is made. The default is 6000 centiseconds (sixty minutes).

UserColor= *colornumber(1-17)*

Specifies the preference setting for the color of your own appointments in the Planner. The default color is blue.

ViewNotByProject= 0 | 1

Indicates whether the tasks in the Task list are currently displayed by project. If this entry is **1**, the tasks are not displayed by project. The default is **0**.

WindowOrder= 0 | 1 1 | 0

Indicates the current display order of Schedule+ windows. The Schedule+ window is represented by **0**, and the Messages window is **1**. The first value for the **WindowOrder=** entry indicates the window that is on top, and the second entry identifies the window that is behind the top window.

[Microsoft Schedule+ Appt Books]

This section is used to track the open Appointment books of other Schedule+ users that you had open when you exited Schedule+.

Count= number

Indicates the number of other users' Appointment Books you had open when you exited Schedule+.

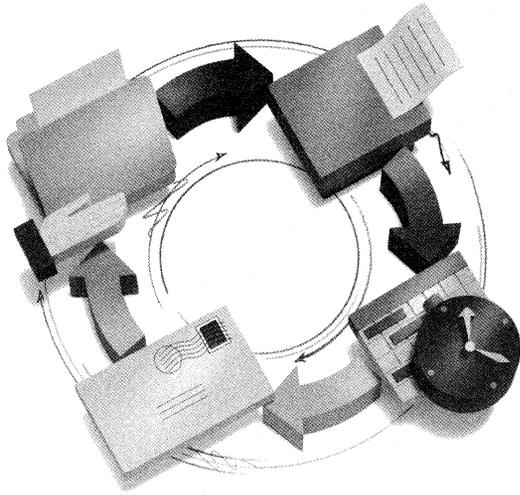
[Microsoft Schedule+ Archives]

This section is used to track the open Archive files that you had open when you exited Schedule+.

Count= number

Indicates the number of Archive files you had open when you exited Schedule+.

Special Topics



Chapter 7 Additional Windows for Workgroups Information **7-1**

About Windows for Workgroups Information	7-2
Setup and Installation Issues	7-2
Network Information	7-4
Information about Using Windows for Workgroups	7-12

Chapter 8 Network Integration with Microsoft LAN Manager and Novell NetWare **8-1**

About Network Integration with Microsoft LAN Manager and Novell NetWare	8-3
Microsoft LAN Manager and Compatible Networks	8-4
Novell NetWare Networks	8-13

Additional Windows for Workgroups Information

This chapter discusses some special information that describes features, functionality, and issues related to Windows for Workgroups 3.1. This information includes topics designed to help further an understanding of Windows for Workgroups functionality, and topics based on feedback that resulted from the Windows for Workgroups 3.1 beta test program.

Contents of This Chapter

About Windows for Workgroups Information	7-2
Setup and Installation Issues	7-2
Windows for Workgroups Setup and Pop-up Messages	7-2
Running a Shared Copy of Windows for Workgroups from the Network	7-3
Preparing for a “Clean” Installation of Windows for Workgroups	7-3
Updating Multi-Boot Programs	7-4
Adding and Removing Components Doesn't Delete .INI Files	7-4
Network Information	7-4
Controlling Windows for Workgroups Server Functionality	7-4
Persistent Network Connections	7-6
Password-List Files	7-7
Browsing Workgroup Resources	7-8
Disabling the NetBEUI Protocol	7-8
NetBIOS—LANA Discussion	7-9
Flavors of Redirector, NET Command Loading Options	7-10
Remote Access to a Windows for Workgroups Workgroup	7-11
Information about Using Windows for Workgroups	7-12
Sharing/Accessing a CD-ROM Player	7-12
Windows for Workgroups Priority Values	7-13
Preventing the StartUp Group from Loading	7-17
Universal Naming Conventions (UNC)	7-17
Additional Parallel Port Entries (LPT4–LPT9)	7-20
Disabling File-Handle Caching	7-20
Running MS-DOS-Based Network-Specific Applications	7-20

About Windows for Workgroups Information

This chapter discusses some special information that describes features, functionality, and issues related to Windows for Workgroups 3.1. This information in this chapter provides further discussion of topics covered elsewhere in this manual, and includes material based on feedback that resulted from the Windows for Workgroups 3.1 beta test program, which involved over 9,000 sites.

The contents of this chapter is divided into the following three major sections:

- Setup and Installation Issues
- Network Information
- Information about Using Windows for Workgroups

Setup and Installation Issues

This section describes some of the topics related to the setup and installation of Windows for Workgroups.

Windows for Workgroups Setup and Pop-up Messages

Many networks have pop-up message facilities so you can see messages sent from other servers or workstations. In most cases, these pop-up programs will cause Windows for Workgroups Setup to fail if a message is received while Windows for Workgroups is being installed. Many networks provide utilities that allow them to work with Windows for Workgroups after installation, but these are generally not available during Setup. You should make sure any such pop-up service is disabled before installing Windows. Please consult specific entries regarding your network in the sections later in this chapter, or consult your network documentation for instructions.

Note The MS-DOS redirector provided with Windows for Workgroups and with the Workgroup Connection do not support the LAN Manager messenger service, and thus won't be affected by message alerts.

Running a Shared Copy of Windows for Workgroups from the Network

Make sure the user's personal WINDOWS directory and the shared WINDOWS directory are on the user's path, and in that order. The Setup program will automatically take care of this if the user chooses to have Setup make modifications to the AUTOEXEC.BAT file.

You might need to mark applications stored on a network directory as read-only to prevent sharing violations when they are run by more than one user at a time. You can set the read-only attribute from within the Windows for Workgroups File Manager or by using the MS-DOS **attrib** command. (Your network may also have an attribute program that you can use.) Most applications require read-only status before they can be shared.

If the user chooses to have Setup make modifications to AUTOEXEC.BAT, Windows for Workgroups Setup modifies AUTOEXEC.BAT to make sure the WINDOWS directory appears on the path ahead of any conflicting directories. However, because of the way some networks implement their startup procedures, a network can change this path when a user logs on. If this happens, modify the network logon procedure so that the paths are set correctly in Windows.

Also, verify user rights for the directories where any Windows for Workgroups .INI and .GRP files are located, and verify that any shared .GRP files are marked read-only. Ensure that the mapping for all network drives is correct and that all search drives are correctly designated.

Preparing for a "Clean" Installation of Windows for Workgroups

If you presently have Windows for Workgroups on your computer and you want to perform a "new" installation to another directory without first removing your existing version of Windows for Workgroups, you will need to remove the reference to WORKGRP.SYS in your CONFIG.SYS file if you are performing an Express install. When Windows for Workgroups Setup is run, Setup will look at the CONFIG.SYS file to see if the WORKGRP.SYS file is referenced for a **device=** entry. If WORKGRP.SYS is present, Setup will copy the PROTOCOL.INI file from the location where the old WORKGRP.SYS file resides and will not perform netcard detection—the values in the existing PROTOCOL.INI file will be used. If you are performing a Custom install, Windows for Workgroups will prompt you with the netcard information, which you may change at this time.

If you want to do a “clean” install to a new directory, the WORKGRP.SYS line needs to be deleted from the CONFIG.SYS file before running Setup to install the new configuration—note, remarking out the WORKGRP.SYS line is not sufficient, you will need to remove the WORKGRP.SYS reference from the CONFIG.SYS file (see the next topic in this section).

Updating Multi-Boot Programs

Many multi-boot programs, such as BOOTCON, will delimit their sections in CONFIG.SYS using REM statements. If you are using one of these programs and are receiving errors when booting, or when entering Windows, check to ensure the Windows for Workgroups drivers are installed correctly in the CONFIG.SYS file.

Adding and Removing Components Doesn't Delete .INI Files

The Setup program for Windows 3.1 can be used to add or remove selected Windows components when being run within Windows. However, when removing components that have created .INI files, such as Mail and Schedule+, the associated .INI files will not be deleted.

After Removing a Windows component you will want to backup, then delete the associated .INI file. This will ensure that when the component is added back into Windows it will install with defaults.

Deleting the .INI file is not necessary when replacing corrupt .EXE or .HLP files for a component, by reinstalling the desired component.

Network Information

This section provides additional information about using the network components provided with Windows for Workgroups.

Controlling Windows for Workgroups Server Functionality

Windows for Workgroups provides configuration options that allows system administrators to control the functionality that a Windows for Workgroups workstation user can perform in relation to sharing locally-connected resources. These configuration options allow a system administrator to disable the ability for a workstation to share local files, share locally-connected printers, or to

disable sharing both files and printers altogether. The purpose of these configuration options is to disable the user interface for accessing the sharing functionality, if any resources are shared on the workstation before the state of these entries is changed, the user will not be able to stop sharing the given resources.

Controlling Sharing

The user interface for sharing resources can be completely disabled in Windows for Workgroups using the **NoSharingControl=** entry in the **[network]** section of the SYSTEM.INI file. If the **NoSharingControl=** entry is used to disable resource sharing, Windows for Workgroups will not allow the user to change the Enable Sharing section of the Network Settings dialog box, which is accessible from the Network icon in Control Panel.

The valid values for the **NoSharingControl=** entry are described in the "SYSTEM.INI: System Initialization File" section of Chapter 6, "Windows for Workgroups Initialization Files."

If you want to disable either file sharing or printer sharing, but not both, the following mechanisms are provided to configure the sharing functionality supported in Windows for Workgroups.

Controlling File Sharing

In order to disable the file sharing user interface present in File Manager, the **NoShareCommands=** entry in the **[restrictions]** section of the WINFILE.INI file can be used to tell File Manager to not display the Share As and the Stop Sharing menu items on the Disk menu (and the related buttons on the toolbar). If the Share As and the Stop Sharing menu items are not accessible to the user, the user will not be able to share a directory on the local workstation, nor will the user be able to stop sharing a directory that was previously shared.

The valid values for the **NoShareCommands=** entry is described in the "WINFILE.INI: File Manager Initialization File" section of Chapter 6, "Windows for Workgroups Initialization Files."

Controlling Printer Sharing

In order to disable the printer sharing user interface present in Print Manager, the **NoShareCommands=** entry in the **[spooler]** section of the WIN.INI file can be used to tell Print Manager to not display the Share Printer As and the Stop Sharing Printer menu items on the Printer menu (and the related buttons on the toolbar). If the Share Printer As and the Stop Sharing Printer menu items are not accessible to the user, the user will not be able to share a printer

connected to the local workstation, nor will the user be able to stop sharing a printer that was previously shared.

The valid values for the **NoShareCommands=** entry is described in the “WIN.INI: Windows for Workgroups Initialization File” section of Chapter 6, “Windows for Workgroups Initialization Files.”

Persistent Network Connections

The persistent network connections established on your workstation during a Windows for Workgroups session are store in a binary file called CONNECT.DAT file, which is located in your WINDOWS directory.

If persistent network connections is enabled, then when you connect to a network drive or network printer, the network connection is stored in the CONNECT.DAT file. The values in your CONNECT.DAT file are used to restore the network connections that were active in your last Windows for Workgroup session. When you disconnect from a network drive or network printer, the network connection is removed from the CONNECT.DAT file.

Since the CONNECT.DAT file is a binary file, the contents can't be modified directly by the user, except by connecting to or disconnecting from a network resource or using the **net use /persistent** command to manipulate the CONNECT.DAT file.

The syntax for the **net use /persistent** command and the supported options is as follows: (more information on the **net use** command can be viewed by typing **net use /?** from the MS-DOS command prompt)

NET USE /PERSISTENT:{YES | NO | LIST | SAVE | CLEAR} [/YES]

/PERSISTENT	Specifies which connections should be restored the next time you log on to Workgroup Connection. It must be followed by one of the following values:
YES	Specifies that the connection you are making and any subsequent connections should be persistent.
NO	Specifies that the connection you are making and any subsequent connections should not be persistent.
SAVE	Specifies that all current connections should be persistent.
LIST	Lists your persistent connections.
CLEAR	Clears your persistent connections.

You may also delete the CONNECT.DAT file in order to clear the persistent network connection information from your workstation.

Password-List Files

Each time you log on to a workstation, Windows for Workgroups offers to create a password-list file if you do not already have one present at that workstation. Each time you enter a password used to connect to a shared resource or to log on to a LAN Manager domain, a check box is present in the password prompt which lets you indicate that the password entered should be stored in the password-list file. If you try to connect to the same resource or log on to the same domain in the future, Windows for Workgroups attempts the operation using the stored password, and if that password is still valid then you are not prompted to re-enter the password. If the password is no longer valid, then you are prompted to update the password-list file with the current password.

You can specify a logon password which serves as the master password used to encrypt that file, or leave the logon password blank, which causes Windows for Workgroups to skip the initial logon prompt normally displayed when Windows starts, but leaves the other passwords stored in your password-list file unprotected. The password-list file is encrypted with a very secure algorithm, using your logon password as the encryption key, so if you forget your logon password you must delete your password-list file in order to be able to log on with your user name.

You can see what password-list files have been created by looking at the **[Password Lists]** section of SYSTEM.INI. You can also tell Windows for Workgroups not to maintain password-list files at all by setting **PasswordCaching=No** in the **[network]** section of SYSTEM.INI. Note that passwords used for access to Mail, for Novell NetWare resources, and for other applications are not normally stored in the password-list file.

An example of the **[Password Lists]** section of the SYSTEM.INI file looks like:

```
[Password Lists]
JOHNS=C:\WINDOWS\JOHNS.PWL
*Shares=C:\WINDOWS\Shares.PWL
```

The ***SHARES=** entry is always present, and points to the SHARES.PWL file. This is a normal password-list file, but the entries in it identify the printers and directories that should be shared when Windows for Workgroups starts, including the share names, comments, and passwords for those resources. This file is encrypted using the same technique applied to other password-list files in order to prevent users from reading the passwords stored in the file, but the password used to encrypt the SHARES.PWL is maintained by Windows for Workgroups and does not need to be set by the user.

Browsing Workgroup Resources

In the Connect dialogs in File Manager and Print Manager, and the Select Computer dialog in applications such as Chat and ClipBook Viewer, you can view a list of the workgroups on your network, and the computers active in each workgroup.

In each workgroup, one computer assumes the job of maintaining the list of workgroups and the list of servers in its workgroup, and several computers may act as “backups” who also maintain this list. It is automatically determined which computers perform this function, and if any of these computers is removed from the network, other computers are automatically chosen to perform these functions. (The **MaintainServerList=** entry in the **[network]** section of SYSTEM.INI is used to define the role your workstation will play in maintaining the list of workgroups and servers in each workgroup.)

If, for some reason, no computers in your workgroup are performing these functions, you will receive an error (6118) when you start Windows for Workgroups indicating that the list of servers is not currently available. To correct this error, be sure that at least one computer in your workgroup is running Windows for Workgroups, sharing is enabled on that computer, the network was not started before Windows on that computer, and the **maintainserverlist=** switch is set to yes or auto (the default). Additionally, it can take up to 15 minutes on a busy network for the list to be constructed, so wait several minutes and try again to see if the problem has been automatically corrected.

Disabling the NetBEUI Protocol

The NetBEUI protocol can be prevented from loading by specifying the **/n** option for the WORKGRP.SYS driver. The **/n** switch indicates that support for the NetBEUI protocol should not be loaded. If WORKGRP.SYS is started with this switch, neither the real-mode NetBEUI implementation (started with NET START NETBEUI, or automatically when a real-mode redirector is started, if no other network protocol is present) nor the protect-mode implementation VNB.386 will be able to load.

Therefore, this switch should only be used when another network protocol is to be loaded and the NetBEUI protocol is not desired. The support for all Windows for Workgroups redirectors (the BASIC and FULL versions in real mode, and VREDIR.386 in Enhanced-mode Windows) is always loaded when WORKGRP.SYS is installed.

NetBIOS—LANA Discussion

Each combination of a NetBIOS network protocol and a network adapter forms a logical network over which computers may communicate with one another. For example, your computer may have both a token-ring adapter and an Ethernet adapter, and you may use the NetBEUI protocol on the Token-Ring network and both the NetBEUI and the TCP/IP protocols on the Ethernet network. In this case, your computer has three logical networks to which it is connected. These logical networks are assigned numbers called “lana numbers” which Windows for Workgroups uses to communicate with them. You can see the lana numbers in use by looking at the **[network.setup]** section of the PROTOCOL.INI file. In the above example, you might see the following:

```
[network.setup]
version=0x3100
netcard=exp16$,1,EXP16$
netcard=ibmtok$,1,IBMTOK$
transport=ms$netbeui,MS$NETBEUI
transport=ms$tcpip,MS$TCPIP
lana0=ibmtok$,1,ms$netbeui
lana1=exp16$,1,ms$netbeui
lana2=exp16$,1,ms$tcpip
```

A protocol may be in use simultaneously on more than one network adapter. The adapters on which a protocol will be used are listed on the **BINDINGS=** line in the protocol's section of PROTOCOL.INI. Each protocol uses one lana number for each network adapter listed on the **BINDINGS=** line. All lana numbers used by a single protocol will follow in sequence; for protocols like NetBEUI which use the **LANABASE=** parameter, the lana numbers used begin with the specified **LANABASE** number and count upwards. In the previous example, NetBEUI had **lanabase= 0** and TCP/IP had **lanabase= 2**.

In addition, some NetBIOS applications may require a network protocol to be available on LANA number 0. By default, Windows for Workgroups places NetBEUI on **lana1** if other protocols are installed. This is done because some network transport protocols will not work properly unless they are specified as **lana0** (NetBEUI can work with LANA values in the range 0–7).

If you have a NetBIOS application that requires the network transport to be set to **lana0**, you must manually edit the PROTOCOL.INI file in some cases.

Note If you must edit the PROTOCOL.INI file by hand, make a backup of the file before you do so—changing parameters or making unwanted changes in the PROTOCOL.INI file may make your Windows for Workgroups network configuration unusable and may require that the network components be re-installed.

Flavors of Redirector, NET Command Loading Options

Windows for Workgroups includes three different options for connecting to the network, depending on your needs and your hardware configuration. If you run Windows in 386 enhanced mode, and do not start the network before starting Windows, the VREDIR.386 driver maintains your connections to the network. The VNB.386 driver loads and acts as your network protocol, by default. This option provides full network functionality including support for LAN Manager APIs and named pipes, and can dynamically support as many network connections as you need. Additionally, it provides the maximum amount of free memory for MS-DOS applications, since these drivers take no memory from your MS-DOS Prompts.

If you start the network before starting Windows by typing NET START WORKSTATION, then the “Full” redirector is started, in the default configuration. (You can explicitly load this option with the NET START FULL command.)

The NETBEUI option also loads and acts as your network protocol unless you have already loaded one or more other protocols. This option provides full network functionality including support for LAN Manager APIs and named pipes, but you must configure your network protocol to support the number of network connections and NetBIOS commands you need. (The defaults are large enough for nearly all users.) It is also possible to exhaust the redirector's allocated memory, which will limit the number of connections you can make in very heavy usage conditions. This option occupies approximately 100K of memory.

If you start the network before starting Windows by typing NET START BASIC, then the “Basic” redirector is started. The NETBEUI option also loads and acts as your network protocol unless you have already loaded one or more other protocols. This option provides limited network functionality, not including support for LAN Manager APIs and named pipes, but including full support for sharing files and printing to network printers. With this option, you must configure your network protocol to support the number of network connections and NetBIOS commands you need. (The defaults are large enough for nearly all users.) It is also possible to exhaust the redirector's allocated memory, which is sufficient to support only a limited the number of connections and open files. This option occupies approximately 80K of memory. If you use this option, network functionality will not be available within Windows for Workgroups. If you use this option with Microsoft Workgroup Connection and Microsoft Windows 3.1, you can use setup to install the MS-Networks option to make network functionality available within Windows.

Windows for Workgroups Setup adds the command NET START to your AUTOEXEC.BAT. By default, this command is equivalent to the NET START NETBIND command, and performs the “NETBIND” function needed to complete loading of protocol and network-adapter drivers. You can cause the NET START command to load other network options by editing the **Autostart=** line in the [network] section of SYSTEM.INI.

Remote Access to a Windows for Workgroups Workgroup

There are several connectivity options available for use with Windows for Workgroups that are designed to allow a physically remote workstation configured with Windows for Workgroups to access workgroup resources. This section discusses two possible scenarios: when a remote user only needs access to Mail in order to exchange messages with other workgroup members; and when a remote user wants access to shared workgroup resources, including files, printers, Mail messages, and Schedule+ scheduling information.

Remote Access to Mail

To provide a remote user with access to Mail, the *Microsoft Mail and Schedule+ Extensions for Windows for Workgroups* can be used to upgrade a Windows for Workgroups postoffice to a Microsoft Mail 3.x postoffice. Once a Windows for Workgroups postoffice has been upgraded, the Microsoft Mail EXTERNAL application can be used to allow a remote workstation to dial into the postoffice using the *Microsoft Mail Remote Workstation* software.

Remote Access to Shared Resources

In order to remote access shared resources in a workgroup, it is necessary for the remote workstation to have access to the network. There are several mechanisms that can be used to achieve this functionality. Once a remote workstation establishes a connection with the local Windows for Workgroups workgroup, the remote workstation has full access to the network as if the workstation was a locally-attached node. For example, the remote workstation has the ability to send and receive Mail messages, update and share Schedule+ calendar information, schedule meetings, and access shared files and printers, as if connected locally.

The Microsoft Remote Access Services (RAS) for use with LAN Manager provides the ability for a remote workstation to dial into a LAN Manager server to provide access to the shared resources available on the network.

Microsoft is also working with third-parties to provide additional solutions to support access to the shared resources of a workgroup from a physically remote workstation. An example of one of the available solutions consists of a device that sits on an Ethernet network and provides dial-in support for a remote Windows for Workgroups workstation to use the NetBEUI protocol over an asynchronous connection.

Information about Using Windows for Workgroups

This section discusses general information and tips about using Windows for Workgroups.

Sharing/Accessing a CD-ROM Player

In 386 enhanced mode, Windows for Workgroups provides the ability to share drives connected to the local workstation, including CD-ROM players. In order to *share* a CD-ROM, the Microsoft CD-ROM Extensions (MSCDEX.EXE) version 2.21 (or higher) needs to be loaded on the workstation that the CD-ROM drive is physically connected to.

In order to support sharing the CD-ROM with other workstations, the */s* option needs to be specified when running MSCDEX is started in order to support sharing of the CD-ROM drive. If a version of MSCDEX is specified in your AUTOEXEC.BAT file when Windows for Workgroups is installed, the Setup program will add the */s* option to the MSCDEX line, and Windows for Workgroups will update the MSCDEX.EXE file to the latest version provided with Windows for Workgroups.

Windows for Workgroups workstations that want to *access* the shared CD-ROM player do not necessarily need to load the MSCDEX program on their local workstations. However, there are some old applications that will not support accessing information off of a CD-ROM (even a remote one) unless the MSCDEX driver is loaded on the local workstation—these old applications check for the presence of the MSCDEX driver and may not execute unless they find the driver in memory. If you have an old application that requires the MSCDEX driver to be loaded, you may run MSCDEX, but do not specify the /s option (since you aren't sharing the CD-ROM on your workstation).

Windows for Workgroups Priority Values

This section describes how Windows for Workgroups interprets the priority values assigned to the Windows system virtual machine (VM), to each executing DOS VM, and for the VSERVER.386 server process. The relationship between the priority value assigned to the server process and the server utilization activity displayed in WinMeter is also discussed in this section.

Note The discussion of priority values presented in this section is also relevant to Windows 3.1. The only difference between this discussion of priority values as used in Windows 3.1 and Windows for Workgroups 3.1 is that the server process in Windows for Workgroups counts as a separate executing task (Windows 3.1 has no such task).

Windows for Workgroups associates a priority value from 1 to 10,000 for the Windows system virtual machine (VM) where all Windows applications are executed, for each executing DOS VM, and for the VSERVER.386 server process. The 386 Enhanced icon in Control Panel is used to change the priority value for the system VM, a Program Information File (PIF) is used to identify the priority setting for an MS-DOS-based application that is executing in a DOS VM, and the slider bar available in the Network Settings dialog box accessible from the Network icon in the Control Panel is used to change the priority assigned to the server process.

Windows for Workgroups uses the priority values assigned to the Windows system VM, a DOS VM, and the server process, to determine the amount of time that is to be allocated to each running task. The actual processing time allocated to these three components of Windows for Workgroups is based on the ratio of the three (or more depending on the number of DOS VMs active in the system) priority values that are in effect at a given time, to the summation of the active priority values for all tasks executing in the system.

The processing time values (as determined by the VM priority value divided by the summation of the priority values) are used to calculate the maximum timeslice a VM will be given. When no shared resources are being used, the

server process releases its timeslice to allow other tasks in the system to execute. When no Windows messages are waiting to be dispatched to Windows-based applications, the Windows system VM will release its timeslice. A DOS VM will use its allocated timeslice or will release it early depending whether an MS-DOS-based application uses one of the various mechanisms to get keyboard input from the user. When the MS-DOS-based application checks for keyboard input, the Windows scheduler will interpret this action as a signal for the DOS VM as releasing its timeslice. Thus, even when an MS-DOS command prompt is sitting idle, although it consumes a lot of CPU time just in the MS-DOS keyboard polling loop, it doesn't use 100%—each “get keyboard character” call releases the timeslice and gives the system a chance to go “idle.”

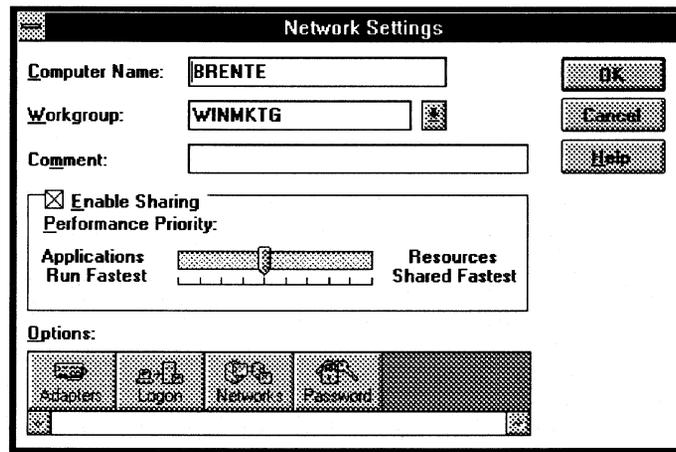
The Windows system VM has two priority values that may be assigned to it: one when the system VM is executing in the foreground, and one when the system VM is executing in the background. The Windows system VM is executing in the foreground when a Windows-based application is active, and is executing in the background when a MS-DOS-based application is active.

A DOS VM has two priority values that may be assigned to it by using a PIF: one when the DOS VM is executing in the foreground, and one when the DOS VM is executing in the background. A DOS VM is executing in the foreground when an MS-DOS-based application is active, and is executing in the background when a Windows-based application is active.

The Windows for Workgroups server process can be considered to always be running in the background. The priority value assigned to the server process is based on the position of the Performance Priority slider bar present in the Network Settings dialog box that can be accessed from the Network icon in Control Panel. Figure 7.1 shows the Network Settings dialog box. When the slider bar position is changed, and the user presses the OK button, a new priority value is written for the **priority=** entry for the **[network]** section of the SYSTEM.INI file. The priority values range from a low priority value giving more processing time to other tasks executing in the system, to a high priority value giving more processing time to the server process. The assigned priority values for each setting of the slider bar in Windows for Workgroups 3.1 is, from left to right: 40, 80, 110, 175, 200, 250, 300, 500, 5000, and 9000.

Figure 7.1

Network Settings dialog box available from the Network icon in Control Panel showing the Performance Priority slider bar



Relationship of Priority Values

In order to put the priority values into context, let's examine some sample scenarios that will help to illustrate how the ratio between the priority values is used to determine the amount processing time allocated to each executing task.

For the first scenario, let's assume only Windows-based applications are executing and the following tasks have these assigned priority values (these numbers are based on the defaults assigned by Windows for Workgroups):

Task	Foreground Priority	Background Priority
Windows system VM	100	50
Windows for Workgroups server process		40 (slider all the way to the left)

Windows for Workgroups calculates the time to allocate to a given task based on the ratio of the priority values for the other tasks executing in the system, to the summation of the active priority values for all tasks executing in the system. In the first scenario, we have Windows-based applications running in the Windows system VM, which has a priority value of 100 (because no DOS VMs are executing, the Windows system VM is executing in the foreground). The Windows server process is assigned the priority value of 40 based on the position of the Performance Priority slider bar.

In order to calculate the amount of time Windows allocates for each given task, the active priority values are first summed together—in this case, the summation of the active priority values is $100 + 40 = 140$. The amount of

processor time that is allocated to the Windows system VM is determined by the ratio of the priority for the Windows system VM, which is 100, to the summation of the active priority values, which is 140. Thus, the amount of time allocated to the Windows system VM is $\frac{100}{140} \cong 0.71$, which indicates that approximately up to 71% of the processing time will be spent in the Windows system VM. Conversely, the amount of time allocated to the Windows for Workgroups server process is $\frac{40}{140} \cong 0.29$, which indicates that approximately up to 29% of the processing time will be spent in the server process to handle the sharing of resources.

For a second scenario, let's assume the following tasks are executing in the system, and the tasks have these assigned priority values (these numbers are based on the defaults assigned by Windows for Workgroups—except for the Windows for Workgroups server process, which has been assigned a new priority value):

Task	Foreground Priority	Background Priority
Windows system VM	100	50
MS-DOS-based application	100	50
MS-DOS-based application	100	50
Windows for Workgroups server process		200 (slider at the fifth tick mark from the left)

Windows for Workgroups calculates the time to allocate to a given task based on the ratio of the priority values for the other tasks executing in the system, to the summation of the active priority values for all tasks executing in the system. In the second scenario, we'll assume a Windows-based application is presently active in the Windows system VM (we could just as easily have selected an MS-DOS-based application to be active since the foreground priority values in this scenario are all the same), which has a priority value of 100. Since a Windows-based application is active, the background priority values for the DOS VMs will be used, thus each have an assigned priority value of 50. The Windows server process is assigned the priority value of 200 based on the position of the Performance Priority slider bar at the fifth tick mark from the left (assuming the first tick mark is at the far left).

In order to calculate the amount of time Windows allocates for each given task, the active priority values are first summed together. In this case, the summation of the active priority values is $100 + 50 + 50 + 200 = 400$. The amount of processor time that is allocated to the Windows system VM is determined by

the ratio of the priority for the Windows system VM, which is 100, to the summation of the active priority values, which is 400. Thus, the amount of time allocated to the Windows system VM is $\frac{100}{400} = 0.25$, which indicates that 25% of the processing time will be spent in the Windows system VM. The amount of time allocated to each DOS VM is $\frac{50}{400} = 0.125$, which indicates that up to 12.5% of the processing time will be spent in each DOS VM. The amount of time allocated to the Windows for Workgroups server process for this scenario is $\frac{200}{400} = 0.5$, which indicates that up to 50% of the processing time will be spent in the server process to handle the sharing of resources.

Preventing the StartUp Group from Loading

To disable the Startup group from functioning in Windows for Workgroups the SHIFT key must be pressed and held down immediately *after* the logon prompt (that is, press and hold down the SHIFT key when you press the OK button in the Log On dialog box, until the Program Manager is displayed on the screen).

In Windows for Workgroups 3.1, the Startup Group is used to load applications automatically at the time Windows for Workgroups is started. By holding down the SHIFT key *after* supplying your requested logon information (that is, your user name and password), Windows for Workgroups will not load the applications specified in the StartUp group.

If you have a null logon password (that is, no password is defined for the specified user), you need to hold the SHIFT key down when you start Windows for Workgroups.

Universal Naming Conventions (UNC)

Windows for Workgroups supports the use of universal naming conventions (UNCs) for entries in Program Item Properties, Program Group Properties, PIF command lines, File Manager, and in the File Open and File Save As common dialog boxes.

UNC redirection is:

- The ability to use a network resource, without formally connecting to it, using the MS-DOS NET USE command.
- The use of a program such as File Manager or Print Manager to access network resources.

UNC redirection is supported in Windows for Workgroups and Microsoft LAN Manager version 1.1 or later, and on 100-percent compatible networks.

Using UNC In Program Item And Group Properties

To have access to a program or group (.GRP) located on a network server, without formally connecting to the resource, use UNC redirection in the command lines, as follows:

UNC Redirection for a Group (.GRP) File

1. From Program Manager's File menu, choose New.
2. Select Program Group, and choose OK.
3. In the Group File text box, type the UNC address for the .GRP file. For example:

```
\\servername\sharename\path\MYGROUP.GRP
```

UNC Redirection for a Program Item

1. From Program Manager's File menu, choose New.
2. Select Program Item, and choose OK.
3. In the Command Line text box, type the UNC address for the program—for example:

```
\\servername\sharename\path\WINPROG.EXE
```

When using UNC addressing for programs, you may receive the following message:

```
The specified path points to a file that may not be
available during later Windows sessions. Do you
want to continue?
```

If you choose Yes, then the icon for the program item is built. UNC redirection for programs is dependent upon the modules or parts that the program uses, where they are located, and how they are loaded. This may not function for all applications.

UNC Redirection in Program Information Files (PIFs)

To have access to an MS-DOS-based application located on a network server, without formally connecting to the resource, use UNC redirection in the PIF's command line, as follows:

1. Run the PIF Editor.
2. In the Program Filename text box, enter the UNC address for the application:

`\\servername\sharename\path\DOSPROG.EXE`

UNC redirection for programs is dependent upon the modules or parts that the program uses, where they are located, and how they are loaded. This may not function for all applications.

This can work even if the executable file relies on another file to run (for example, if MS-DOS 5.0's EDIT.COM and QBASIC.EXE [the later is needed to run the former] are both copied onto the network share, UNC redirection will work properly).

UNC Redirection for Referencing Files

To copy or move files and directories in File Manager to a network share without formally connecting to the resource, or to reference a file in the File Open or File Save As common dialogs, you can use UNC redirection as follows:

To Copy or Move File(s) or Directory(s) in File Manager

1. Run File Manager.
2. Select the file you want to copy or move.
3. From the File menu, choose Move or Copy.
4. In the To text box, type the UNC address—for example:

`\\servername\sharename\path\`

To Reference a File in the File Open or File Save As Common Dialogs

1. Select the Open or Save As options from the File menu.
2. In the File Name text box, type the UNC address—for example:

`\\servername\sharename\path\`

Additional Parallel Port Entries (LPT4–LPT9)

Windows for Workgroups supports up to nine parallel port entries to be defined as LPT1 through LPT9. This allows up to nine connections to be defined for network printers. However, Windows for Workgroups only defines LPT1 through LPT3 by default in the **[Ports]** section of the WIN.INI file. In order to support LPT4 through LPT9, the appropriate entries must be manually added to the **[Ports]** section of the WIN.INI file by the user.

Although Windows for Workgroups supports the use of LPT4–LPT9, most MS-DOS-based applications only support the use of LPT1–LPT3.

For example, adding the following entries to the **[ports]** section of WIN.INI will allow additional LPT ports to be used by Windows for Workgroups and will show up in the Connect dialog box accessible from the Printers icon in Control Panel:

```
[ports]
LPT4:=
LPT5:=
LPT6:=
LPT7:=
LPT8:=
LPT9:=
```

Disabling File-Handle Caching

By default, Windows for Workgroups keeps a minimum of 14 handles open, which may limit the number of users that can run a shared copy of Windows for Workgroups from a single server. You can modify SYSTEM.INI to decrease the number of cached file handles. The minimum is 2. If you decrease the number of cached file handles, Windows for Workgroups may slow down, but more users should be able to access Windows for Workgroups from a single server.

To decrease the number of cached file handles, set the value of the entry **CachedFileHandles=** in the **[boot]** section of SYSTEM.INI. For your network, you may also be able to increase the number of file handles available on the server where the shared copy of Windows for Workgroups resides.

Running MS-DOS-Based Network-Specific Applications

If you have problems running a network-specific application that is an MS-DOS-based application, you may need to run the application in exclusive mode. You can use PIF Editor to do this. If the application creates or deletes files, it may not be running exclusively even if it has been set to run in exclusive mode.

may not be running exclusively even if it has been set to run in exclusive mode. This is because Windows for Workgroups is alerted every time a file is manipulated. You can prevent this alert by setting **FileSysChange=no** in the **[386Enh]** section of SYSTEM.INI.

If you have problems running an MS-DOS-based application or running the MS-DOS Prompt, you may need to adjust the **TimerCriticalSection=** entry in the **[386Enh]** section of SYSTEM.INI. The **TimerCriticalSection=** value prevents switching or multitasking for a specified time period (in milliseconds), ensuring that only one MS-DOS-based application at a time receives timer interrupts. For example, to specify a 10,000 millisecond time-out, set **TimerCriticalSection=10000**.

In some cases, you may need to remove or comment out this entry (for example, if you are using Novell NetWare as a secondary network configuration).

Note Using the **TimerCriticalSection=** entry may slow down Windows for Workgroups.

Network Integration with Microsoft LAN Manager and Novell NetWare

Windows for Workgroups provides network interoperability with Microsoft LAN Manager and compatible networks and Novell NetWare. This chapter provides a discussion of the network interoperability support provided by Windows for Workgroups, and describes special information to assist configuring and integrating Microsoft LAN Manager and Novell NetWare with Windows for Workgroups.

Related information

- *Windows for Workgroups Resource Kit: Chapter 1, “Networks—A Technical Discussion;” Chapter 2, “Windows for Workgroups Architecture;” Chapter 5, “Windows for Workgroups Setup Information Files;” Chapter 6, “Windows for Workgroups Initialization Files;” Chapter 7, “Additional Windows for Workgroups Information.”*

Contents of This Chapter

About Network Integration with Microsoft LAN Manager and Novell NetWare	8-3
Microsoft LAN Manager and Compatible Networks.....	8-4
LAN Manager Log On Support.....	8-4
Microsoft NetBEUI Implementation.....	8-5
Messenger Support	8-6
Named Pipe Support	8-6
Encryption Service.....	8-7
Browsing Between Windows for Workgroups and LAN Manager	8-7
Adding LAN Manager Servers to a Windows for Workgroups Environment.....	8-8
Adding Windows for Workgroups to Existing LAN Manager Environments	8-11
Information about Using Microsoft Comm Server 1.1	8-12
Novell NetWare Networks	8-13
Novell NetWare Drivers.....	8-13
MSIPX: NDIS-Compliant IPX Implementation.....	8-15
ArcNet Support.....	8-17
Token Ring and Source Routing	8-18
NetWare Drive Assignments.....	8-18

Novell ODI Drivers	8-19
Novell NetBIOS.....	8-19
Additional Information About Novell NetWare.....	8-20

About Network Integration with Microsoft LAN Manager and Novell NetWare

The networking functionality of Windows for Workgroups is split into two components: a client and a server. This chapter discusses the capabilities of Windows for Workgroups when interoperating with other network operating systems (NOS) and some issues related to working with these NOS environments.

Note Although Windows for Workgroups will communicate with any SMB protocol-compliant file server (and vice versa), both the Windows for Workgroups workstation and the remote computer must be using the same transport protocol (for example, NetBEUI) in order to communicate.

Figure 8.1 illustrates the major networks in which a Windows for Workgroups workstation can act as a client. These include a Windows for Workgroups server, a Windows NT server, a LAN Manager for OS/2 server, a LAN Manager for UNIX server, and a Microsoft Network (MS-NET) server. In addition, a Windows for Workgroups workstation can also connect to any server that supports the SMB protocol, including a DEC Pathworks server, a 3Com 3+Open server, and an IBM LAN server. Using the dual-network configuration support, a Windows for Workgroups workstation configured to use the MSIPX protocol also has access to a Novell NetWare server.

Figure 8.1

Server interoperability available for Windows for Workgroups client

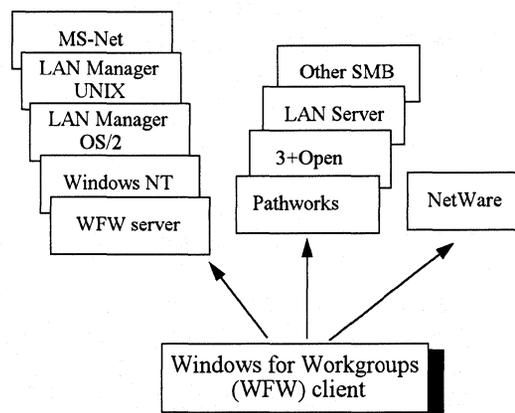
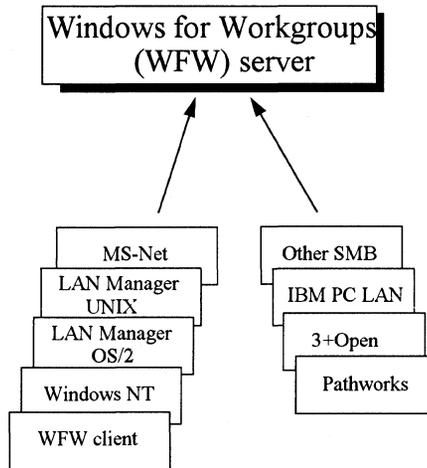


Figure 8.2 illustrates the major networks for which a Windows for Workgroups workstation can act as a server. Any network client that supports the SMB protocol and has a compatible network redirector loaded has access to a Windows for Workgroups server (that is, a Windows for Workgroups workstation running in 386 enhanced mode). These include a Windows for Workgroups workstation, a Windows NT workstation, a LAN Manager for OS/2 workstation, a LAN Manager for DOS workstation, and a Microsoft

Network (MS-NET) workstation. Other SMB-compatible network clients include a DEC Pathworks workstation, a 3Com 3+Open workstation, and a workstation using the IBM PC LAN Program (PCLP).

Figure 8.2

Interoperability available for clients to Windows for Workgroups server



Microsoft LAN Manager and Compatible Networks

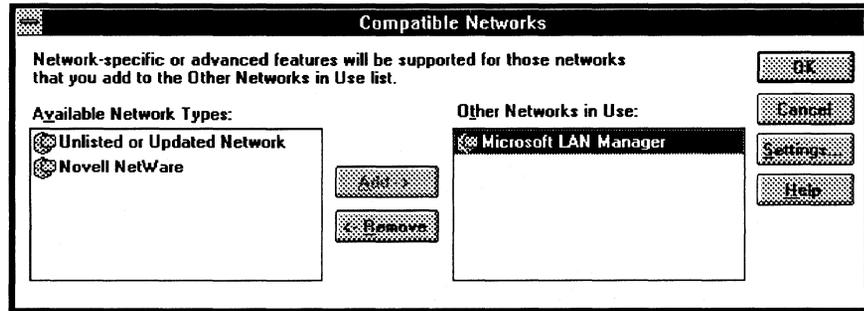
Windows for Workgroups integrates extremely well with LAN Manager networks. Both use the same network transports, providing connectivity across a broad range of hardware platforms. Similarities far outweigh the differences between Windows for Workgroups and LAN Manager. Differences between specific Windows for Workgroups and LAN Manager components are discussed in this section.

LAN Manager Log On Support

Selecting Microsoft LAN Manager in the Compatible Networks dialog box provides support for user authentication by a LAN Manager server when logging on to a LAN Manager domain (used for user-level security on a LAN Manager server) and running LAN Manager logon scripts when logging on.

Figure 8.3

Selecting Microsoft LAN Manager in the Compatible Networks dialog box configures LAN Manager support

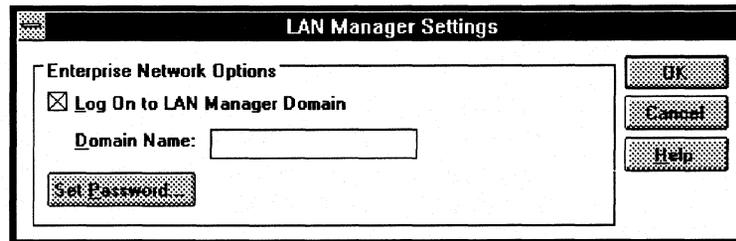


LAN Manager Log On Security

The LAN Manager Settings dialog box is used to identify the name of the LAN Manager Domain in which the Windows for Workgroups user's account is defined. A server designated as a "domain controller" in this domain performs user authentication when the user logs on the network and provides for user-level security. The Set Password button is used to change the user's password on the LAN Manager server.

Figure 8.4

The LAN Manager Settings dialog box is used to relate the Windows for Workgroups workstation with a LAN Manager domain



LAN Manager Script Support

Specifying Microsoft LAN Manager in the Compatible Networks dialog box (see Figure 8.3) provides support for using LAN Manager logon scripts when logging on to a LAN Manager Domain. The logon script support is provided by the LMSCRIPT.EXE application (and accompanying LMSCRIPT.PIF) which is copied to the SYSTEM subdirectory when LAN Manager support is enabled through the Compatible Networks dialog box.

Microsoft NetBEUI Implementation

The Microsoft NetBEUI transport protocol provided with Windows for Workgroups is functionally identical to the NetBEUI transport shipped as part of Microsoft LAN Manager version 2.x. There are some differences, however, with the way the protocol is implemented in the Windows for Workgroups product.

First, the NetBEUI implementation is provided as both a real-mode device driver and a protected-mode virtual device driver (VxD). The real-mode device driver is used when running Windows for Workgroups in standard mode and is loaded before starting Windows for Workgroups (by typing the **net start workstation** command). The protected-mode VxD is used when Windows for Workgroups is running in 386 enhanced mode on computers with the Intel 80386 or higher CPU. The biggest end-user benefit of the VxD implementation is that it frees up conventional memory to offer more available memory in an MS-DOS virtual computer.

Another difference between real-mode implementation of NetBEUI and VxD implementation is how the **sessions** and **ncbs** parameter is supported in 386 enhanced mode. The **sessions** parameter is used to identify the maximum number of connections to remote computers that can be supported from the redirector. The **ncbs** parameter is used to identify the maximum number of NetBIOS commands that can be used. The VxD implementation of NetBEUI dynamically adjusts the value of the **sessions=** and **ncbs=** parameter as needed during the use of Windows for Workgroups, and thus the **sessions=** and **ncbs=** entries under the **[MSSNETBEUI]** heading in PROTOCOL.INI are ignored when using the VxD version of NetBEUI.

Messenger Support

The services that the NET.EXE SMB network shell provides with Windows for Workgroups (used by the *Workgroup Connection* product to allow an MS-DOS workstation to access a resource shared by a Windows for Workgroups workstation) includes the redirector. (See Chapter 1 for details.)

The NET.EXE shell doesn't support the messenger SMB service, and thus does not receive SMB-messenger type messages from other workstations or servers. For example, a Windows for Workgroups workstation doesn't receive status messages or server alerts from a LAN Manager workstation.

Note Because Windows for Workgroups does not support the SMB messenger service, the Microsoft LAN Manager Winpopup and Netpopup message alert utilities will not operate with Windows for Workgroups.

Named Pipe Support

The Windows for Workgroups redirector supports client-side use of named pipes. Named pipes are the basis for the communication mechanism between the client and advanced client/server applications such as Microsoft SQL Server and Microsoft Comm Server. No modifications are needed to access SQL Server or Comm Server from a Windows for Workgroups workstation.

The client-side software should be installed as if installing it under Windows 3.x (follow the instructions described in the documentation provided with your application software). See the “Flavors of Redirector, NET Command Loading Options” section in Chapter 7, “Additional Windows for Workgroups Information” for a more information about the redirector support for named pipes.

Encryption Service

No separate encryption service is provided as part of the Windows for Workgroups network redirector, as is provided with the LAN Manager MS-DOS client. Encryption is built into the redirector and does not require any additional RAM to operate, unlike the LAN Manager MS-DOS client.

Browsing Between Windows for Workgroups and LAN Manager

A Windows for Workgroups *workgroup* is analogous to a LAN Manager *domain* in the fact that it's a logical grouping of workstations. A Windows for Workgroups workgroup, however, doesn't share any of the advanced security features offered as part of a LAN Manager domain.

Allowing Windows for Workgroups Workstations to See LAN Manager Servers in a Domain

In order for Windows for Workgroups workstations to view a list of LAN Manager servers in a given domain (by using the **net view** command), there must be at least one Windows for Workgroups workstation on the network that sets the **workgroup=** entry in the **[network]** section of the SYSTEM.INI file to the name of the LAN Manager domain. (The **workgroup=** entry can be changed from the Networks option in the Windows Control Panel.) Once a Windows for Workgroups workstation is a member of the LAN Manager domain, this workstation will distribute the names of the LAN Manager servers defined in the domain to other Windows for Workgroups workstations on the network. This configuration needs to be duplicated for each LAN Manager domain to support viewing LAN Manager servers by Windows for Workgroups workstations.

Allowing LAN Manager Workstations to See Windows for Workgroups Servers

By default, a Windows for Workgroups workstation running in 386 enhanced mode won't announce its presence to LAN Manager clients and servers on the network in order to reduce the amount of broadcast traffic on the network. This

means that if a user on a LAN Manager client workstation types the **net view** command to view the available servers on the network, the Windows for Workgroups workstation won't be displayed in the returned list.

How the Windows for Workgroups workstation announces its presence is defined by the **LMAnnounce=** entry in the **[network]** section of SYSTEM.INI. By using **LMAnnounce=yes**, the Windows for Workgroups workstation running in 386 enhanced mode will announce its presence to LAN Manager clients and servers. The default, **LMAnnounce=no**, means the workstation won't announce its presence on the network—this default is chosen to minimize the level of network traffic.

Adding LAN Manager Servers to a Windows for Workgroups Environment

There are a number of very powerful reasons for adding LAN Manager servers to Windows for Workgroups environments. LAN Manager servers provide the server component of a client/server environment, and a number of products are available that exploit this type of environment. These products typically centralize a function to make greater use of a valuable resource or specialized environment. Three of these products are SQL Server 4.2, Comm Server 1.1, and Remote Access Services for LAN Manager. Windows for Workgroups can make use of these and other client/server products to increase the functionality of the workgroup.

In addition to being the platform for client/server products, LAN Manager servers can also be used as special security servers. Security can be applied in two ways. The first is logon security, by which LAN Manager can be used to provide logon security and logon scripting facilities. LAN Manager servers also have file sharing security options, enabling LAN Manager to act as a super secure file platform.

LAN Manager Security Options

When installing a LAN Manager server, an important decision is the type of security the server will implement. LAN Manager servers provide two types of security. *Share-level security* is very much like the security provided on Windows for Workgroups workstations. Anyone who wants to use a shared resource must know the password assigned to the resource. Unlike the Windows for Workgroups password structure, however, there is only one password on LAN Manager shares. Also, unlike Windows for Workgroups, it is possible to have multiple shares defined to the same resource on a LAN Manager server, each with a different password.

The second type of security LAN Manager servers provide is *user-level security*, a much more common and more powerful form of security used with LAN Manager servers. In this security system the password is associated with the user, not the shared resource. Thus a user has a single password for all shared resources to which he or she has access on all servers in a domain. The resources the user may use and the level of access the user has is based on access-levels set by the domain administrator.

The remainder of this discussion assumes the LAN Manager server is running with user-level security and describes the implications of using a Windows for Workgroups workstation to access a LAN Manager server.

Creating a Server-Based Security Environment

Each Windows for Workgroups user does not have to be added to the LAN Manager user-security database to use resources on the LAN Manager server. LAN Manager uses the concept of a *Guest* account so that users not identified to LAN Manager are allowed to use resources through a Guest account. Usually this account is granted access to a very small set of resources, such as a public disk directory.

If a Windows for Workgroups user is to have more permissions on a server than the guest account, a user account must be created on the server. Note that not all user-account attributes available with LAN Manager workstations are available with Windows for Workgroups workstations.

- LAN Manager has the ability to “age” passwords and set a maximum age for a password. In LAN Manager environments, a message would be sent to the user when the password is about to expire. Because Windows for Workgroups doesn't support messaging, the user won't receive the message.
- Windows for Workgroups passwords are used differently from passwords on LAN Manager workstations. The user may change their LAN Manager password in two different ways. First, the user may use the MS-DOS console command **net password**. Second, the user may change their LAN Manager password by using the Network Control application. The Set Password option is part of the Networks Setting's dialog box. (See Figure 8.4.)

Note An expired password on the LAN Manager server prohibits the user from logging on to their Windows for Workgroups workstation. If the LAN Manager logon fails, the Windows for Workgroups logon will fail as well. If your password on the LAN Manager your account is defined on expires, you may log on to your Windows for Workgroups workstation by disabling the Log On to LAN Manager Domain check box in the LAN Manager Settings dialog box accessible from the Control Panel.

- The LAN Manager server administrator can set an expiration date for a user or disable a user's account. This stops the user from accessing resources on the LAN Manager server. They can still access resources shared by other Windows for Workgroups workstations.
- LAN Manager workstations have access to a facility known as “Logon Scripts,” which is a pre-defined script (batch file) run on the LAN Manager workstation once their user ID and password are authenticated. Windows for Workgroups workstations run logon scripts on the local workstation if using the LAN Manager domain logon feature. (To use this feature, from the Network Control application, choose LAN Manager Settings from the Networks menu.)

Users on LAN Manager servers can collectively be given access-rights to resources through *group* accounts. Groups make the management of security of a LAN Manager server much easier. A group can be given access to a resource, say a file. When a user is added to the group, the user is automatically given access to that file. This is a convenient method of giving resource permissions to everyone in a workgroup.

A single LAN Manager server may be shared by multiple workgroups. The use of groups is an easy way to control access permissions among the various groups.

For example, suppose the server is used by two workgroups called *Finance* and *Manufacturing*. The server contains files for the Finance workgroup, including a chart of accounts file, the accounts payable ledger file, and so on. It also contains files for the Manufacturing workgroup, including the build schedule, current component inventories, and the latest vendors' price sheets.

The administrator for the server can establish groups so that a person in the Finance workgroup has access to the financial files but not the manufacturing files. Likewise, the Manufacturing group would be granted access to their own files but not the financial information. This way, no one in Finance could accidentally delete Manufacturing's build schedule, for example.

Using groups also makes it easier to perform administrative tasks for new and relocated employees. For example, when a new person joins the Finance group,

the server administrator needs only to create the user's account and add it to the Finance group, giving that new employee all the same permissions as the rest of Finance. When an employee is transferred from Finance to Manufacturing, the administrator can simply move the user account from one group to the other. This saves the laborious task of adding individual resource permissions.

Once the user accounts for each user in a workgroup are created and associated with the resources to which they have access (either directly or through groups), the Windows for Workgroups user can access LAN Manager server resources. LAN Manager and Windows for Workgroups share the same underlying NOS (Network Operating System) protocol, called SMBs (Server Message Blocks—for more information, see Chapter 1). Though a person using the Windows for Workgroups workstation does not “logon” to the network like LAN Manager users do (that is, the password a user specifies provides access to the local workstation and is not validated by a server as is supported by LAN Manager), access to a given resource is always verified by the LAN Manager server to confirm that the user has been granted permission to use the resource, even for Windows for Workgroups workstations.

Adding Windows for Workgroups to Existing LAN Manager Environments

Windows for Workgroups workstations can immediately gain access to resources shared by LAN Manager servers, as long as the LAN Manager server is using the Microsoft NetBEUI transport protocol. Designed to function in a cooperative environment (without a central-authority server), Windows for Workgroups treats some aspects of networking differently than LAN Manager.

Controlling Access

There are two very simple ways to give Windows for Workgroups users access to a LAN Manager server:

- Put the resources they want to use within access of the Guest account
- Create a group account for everyone in the workgroup

Windows for Workgroups users typically use guest privilege on LAN Manager servers running with user-level security. As mentioned before, the Guest account (depending on how the administrator sets it up) generally has access to public shares. As an alternative, you can define one group account for all of the workgroup's user IDs. This way, you can grant access to workgroup users to specific directories on the server and prevent access by others not in the workgroup.

A third alternative, of course, is to create individual user accounts for each person in a Windows for Workgroups workgroup, just as you would for anyone using a LAN Manager workstation. This option is more time-consuming than the previous two suggestions.

LAN Manager Password Support

Windows for Workgroups users can change their passwords on the LAN Manager server from their workstations. If password aging is used on the LAN Manager server, the user must use the **net password** command to change the password stored on the LAN Manager server. The Change Password button in the Network Control Panel, which is used to change the password for the local Windows for Workgroups workstation, doesn't affect the password defined for the user on the LAN Manager server—the Networks button in the Network Control Panel provides access to the LAN Manager Settings dialog box, which is used to change the user's password for the specified LAN Manager domain.

Information about Using Microsoft Comm Server 1.1

There are no known problems running Comm Server Windows client with Windows for Workgroups, however there are some known issues with using other aspects of Microsoft Comm Server version 1.1 with Windows for Workgroups version 3.10. Here is a summary of the known issues:

- The Comm Server client setup program assumes the presence of the LANMAN.INI file, which is used by LAN Manager, but not by Windows for Workgroups. Windows for Workgroups uses the SYSTEM.INI file to store values that LAN Manager stores in the LANMAN.INI file. Setting up the Comm Server client software on a Windows for Workgroups 3.10 workstation requires that the program be installed manually. The *Comm Server Installation Guide* explains how to perform a manual installation of the Comm Server client program.
- The Comm Server DOS client must be set up to load the NAP within a Windows DOS box if the Windows for Workgroups 3.10 workstation is running in 386 enhanced mode.
- The Comm Server DOS client is limited to accessing one server at a time in version 3.10 of Windows for Workgroups.

The following practical limitations are also known:

- You cannot run Rumba® 5250 emulator and IBM PC Support through Comm Server at the same time with Windows for Workgroups, except in standard mode of Windows. Either function works fine on its own.
- You cannot have sessions active to more than one Comm Server with the DOS client. This limitation does not apply to the Windows client.

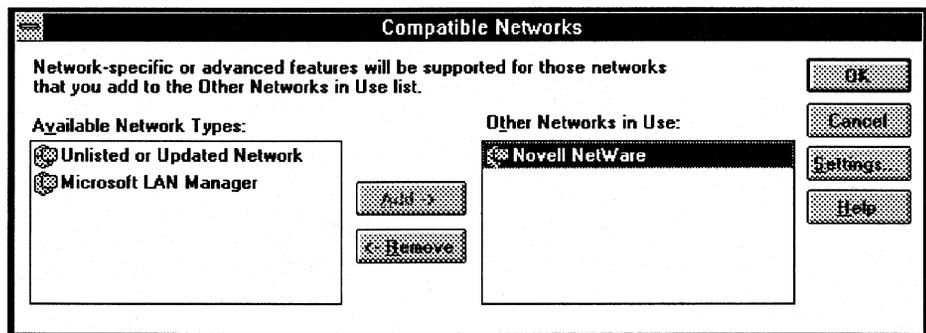
Novell NetWare Networks

Windows for Workgroups provides components for interoperating with Novell NetWare networks version 2.10 and above. The components allow a Novell NetWare client workstation to be upgraded to a Windows for Workgroups workstation, providing access to both Novell NetWare servers and to Windows for Workgroups servers (or allows a Windows for Workgroups workstation to be configured to access a Novell NetWare server). Once a Novell NetWare workstation is upgraded to Windows for Workgroups, the newly configured workstation will support both the Novell IPX transport protocol and the Microsoft NetBEUI transport protocol. Let's first discuss the steps to configure the Windows for Workgroups workstation to use the NetWare integration components.

The first step is to select Novell NetWare in the Compatible Networks dialog box, as shown in Figure 8.5. This copies the necessary Novell files to your WINDOWS and SYSTEM directories. Windows for Workgroups provides support for a Windows for Workgroups client using both NetBEUI and MSIPX to access a Windows for Workgroups server. A native Novell NetWare workstation using the IPX protocol can access only a Novell NetWare server and not a Windows for Workgroups server. (Chapter 1 of this manual describes how Windows for Workgroups supports a dual-network protocol configuration.)

Figure 8.5

Selecting Novell NetWare in the Compatible Networks dialog box configures the NetWare support



The Settings button in the Compatible Networks dialog box, as shown in Figure 8.5, is used to access the NetWare driver settings information supported by the NetWare Winnet driver.

Novell NetWare Drivers

The Novell NetWare integration provided by Windows for Workgroups is comprised of several components. All of the components necessary for integrating NetWare with Windows for Workgroups are provided in the

Windows for Workgroups product—any extra components not listed are available from Novell but are not necessary in a typical Windows for Workgroups configuration. The Novell components provided as part of the Windows for Workgroups product are made available by Microsoft to ensure that the user is running with the latest release of each necessary driver.

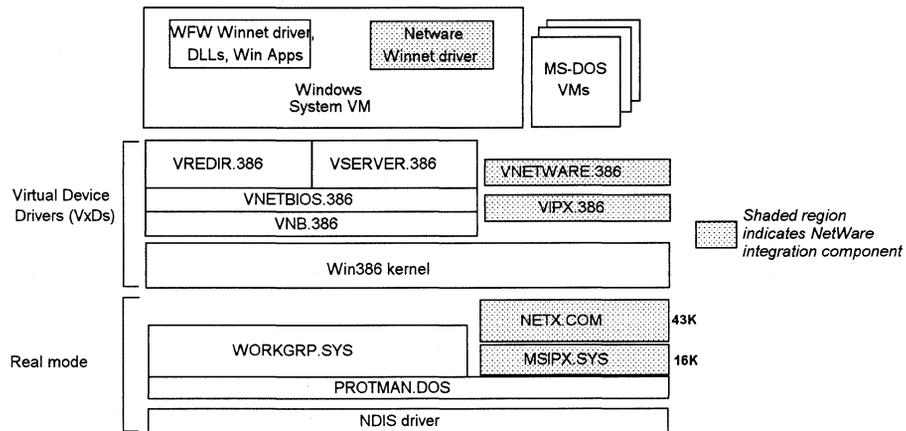
The following table summarizes the components that make up the NetWare integration and provides a description of the driver (including the developer of the driver). All of these drivers are included with Windows for Workgroups.

Driver name	Driver description
NETX.COM (version 3.26)	DOS-independent version of the Novell NetWare shell (Novell)
MSIPX.COM (version 3.10)	NDIS-compliant implementation of IPX transport protocol (Novell/Microsoft)
MSIPX.SYS (version 3.10)	Stub that binds the MSIPX transport protocol to the NDIS driver (Microsoft)
VIPX.386	VxD that virtualizes the IPX protocol to provide support for multiple MS-DOS virtual computers—unchanged from generic NetWare and Windows 3.1 (Novell)
VNETWARE.386	VxD that virtualizes NETX (shell) and also provides some NCP buffer management to support multiple MS-DOS virtual computers—unchanged from generic NetWare and Windows 3.1 (Novell)
NETWARE.DRV	NetWare Winnet driver—unchanged from generic NetWare and Windows 3.1 (Novell)
NWPOPUP.COM	Provides user interface for NetWare's messaging—unchanged from generic NetWare and Windows 3.1 (Novell)

Figure 8.6 illustrates how the Novell NetWare components integrate with the Windows for Workgroups components to enable connectivity to both Windows for Workgroups servers and NetWare servers from a Windows for Workgroups 386 enhanced mode workstation. (A standard mode configuration of Windows for Workgroups would look similar, however, the virtual device drivers would not be used—the real-mode MS-DOS device drivers are the same.)

Figure 8.6

Windows for Workgroups components that provide NetWare integration—shown for 386 enhanced mode



MSIPX: NDIS-Compliant IPX Implementation

MSIPX is a full implementation of the IPX transport protocol designed to provide interoperability between Windows for Workgroups and Novell NetWare workstations.

You may recall from the discussion of networking protocols in Chapter 1 of this manual, that the Network Driver Interface Specification (NDIS) provides for the ability for multiple transport protocols to share a single network adapter card. The protocol manager is then responsible for arbitrating the network traffic on the network adapter card.

IPX is the proprietary transport protocol used by Novell NetWare. Many installations of NetWare use the IPX.COM monolithic protocol stack as the IPX protocol implementation. This implementation of IPX is referred to as a monolithic transport protocol stack because it incorporates not only the Transport and Network layer drivers, and also the Data Link layer drivers in a single combined entity. (Not all of the network layer is in IPX.COM—some of it is in NETX.COM.)

IPX.COM is an MS-DOS TSR that runs the IPX transport and is built for use with one specific network adapter card using workstation-specific information. This is done using a linker (NLINK) to create IPX.COM from an object module containing standard IPX transport code (IPX.OBJ) and an object module specific to the network adapter being used (for example, NE2000.OBJ). The information in IPX.COM identifies the type of network adapter card, the network adapter card driver, and the configuration information for the network adapter card (for example, the IRQ settings, base memory address, and the base port address as appropriate). Because the IPX.COM file contains information specific to hardware installed on a given workstation, this file can't easily be

moved from one workstation to another unless the workstations are configured identically (that is, same network adapter card, same interrupts, and so on.).

Novell provides their implementation of the IPX transport protocol to work with their ODI network card drivers, which are not supported for use with Windows for Workgroups. As Novell's implementation of the IPX transport protocol stack works with ODI, there are no provisions for working directly with NDIS and its ability to support multiple network transport protocols running on a single network adapter card.

In order to provide a wide range of interoperability options, Microsoft developed code to tie the IPX protocol to NDIS-compliant drivers, which is used to build a new version of IPX.COM. This implementation of IPX, called MSIPX (to differentiate it from the IPX.COM provided by Novell), breaks the IPX monolithic driver into its major components—the IPX protocol (MSIPX.COM) and a translation layer (MSIPX.SYS). The translation layer binds the IPX protocol to NDIS to enable mapping of the IPX driver calls to their related NDIS driver calls. The MSIPX transport can be used with Windows for Workgroups transports (for example, NetBEUI) to share the same NDIS-compliant network adapter card driver. This allows both Windows for Workgroups and NetWare networks to share the same network adapter card for both protocol stacks.

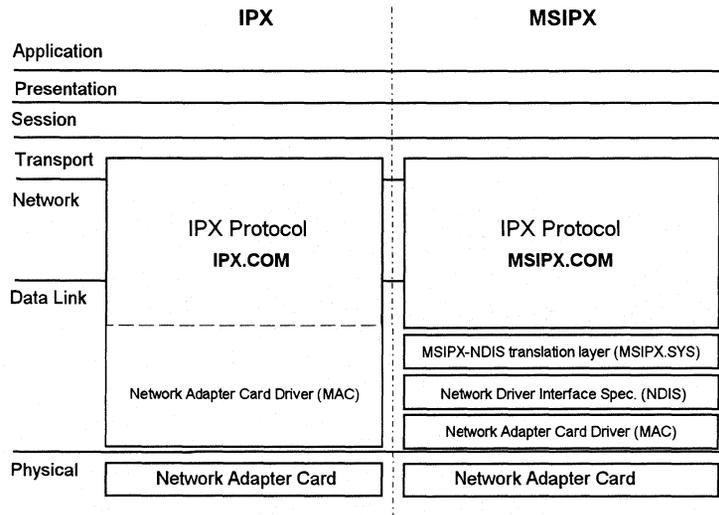
MSIPX.COM is built in the same manner as IPX.COM in that NLINK is used to create MSIPX.COM from IPX.OBJ and driver code to support the IPX transport on top of NDIS. The NDIS driver code, NWNDIS.OBJ, uses the NDIS interface to communicate with a MAC driver for the network adapter card being used.

Note MSIPX.COM is Novell's implementation of IPX. The only additional support provided in MSIPX.COM from IPX.COM is the ability to use NDIS-compliant network adapter drivers.

The MSIPX drivers provided with Windows for Workgroups 3.1 are implemented as real-mode drivers, thus they reside in conventional memory.

Figure 8.7

Relationship of IPX to MSIPX components



When Windows for Workgroups is installed on an existing NetWare workstation, the Setup program will examine the IPX.COM file to obtain the network adapter card settings needed to properly configure Windows for Workgroups. Note, however, that a Windows for Workgroups workstation does not need to be configured with the NetWare workstation software before installation in order to provide connectivity to NetWare servers. All of the necessary components are provided with Windows for Workgroups.

ArcNet Support

The Microsoft version of IPX (MSIPX) supports the Token Ring and Ethernet (both DIX and IEEE 802.2/IEEE 802.3) MAC media header formats. It does not, however, support the ArcNet media header format. Because of the differences in how NetBEUI and MSIPX transmit on ArcNet media, Windows for Workgroups and NetWare client software cannot share the same ArcNet network adapter card. An explanation of the differences is described in the following paragraphs.

When an ArcNet card is used with Windows for Workgroups, the NDIS driver claims to be an Ethernet DIX+802.3 MAC and encapsulates a transport's request to send Ethernet packets in ArcNet packets. The ArcNet NDIS driver presents itself as an Ethernet driver to the transport and actually transmits the Ethernet headers across the network. The driver constructs a fake Ethernet address by putting the ArcNet card's address in the lower byte of the Ethernet address. Ethernet multi-casts are emulated by performing ArcNet broadcasts. This allows Windows for Workgroups to run cleanly over ArcNet.

However, when an ArcNet card is used in a NetWare environment, the IPX.COM driver generates packets in a proprietary form—a general

description of the algorithm used can be found in a supplement to the *Novell LAN Drivers Developers Guide*.

For additional information, consult the *3Com/Microsoft LAN Manager Network Driver Interface Specification Version 2.0.1*, and the *Novell Standard Microsystems ArcNet Supplement* (a supplement to the *Novell LAN Drivers Developers Guide*).

Token Ring and Source Routing

If Windows for Workgroups detects a Token Ring network adapter card in the workstation at the time the user selects Novell NetWare as the compatible network, Windows for Workgroups places a reference to the ROUTE.COM utility in the AUTOEXEC.BAT. ROUTE.COM is used to provide source routing support on a Token Ring network.

There is one situation, however, where ROUTE.COM will need to be removed from the AUTOEXEC.BAT manually by the user. If the version of the network card driver used for the Token Ring card on the NetWare server does not support source routing, then the ROUTE.COM utility must be removed from the AUTOEXEC.BAT file of workstations connected to the server—for example, versions of the IBM Token Ring network adapter driver earlier than 2.4 do not support source routing. (The ROUTE.COM line in the AUTOEXEC.BAT is listed after the MSIPX line, but before the NETX line.)

You can determine the version of the Token Ring driver that is being used on the NetWare server by typing **config** at the server console. This will display the current server configuration and will identify the version number for any drivers used by the NetWare server.

NetWare Drive Assignments

When you set up NetWare support in Windows for Workgroups, the value of the **lastdrive=** command in your CONFIG.SYS file is set to **p** (unless the command already exists and has a value from Q to Z)—when you restart your computer, your logon directory is drive Q. If you have any NetWare login scripts, make sure they reference the correct drive letter for your logon directory.

When the **lastdrive=** command is set to **p**, you can use drive letters from A to P to connect to other Windows for Workgroups computers or Microsoft LAN Manager servers. Letters after P are reserved for connecting to Novell NetWare servers.

Windows for Workgroups uses the **lastdrive=** command to identify the available drive letters to which a user may map remote file shares to. NetWare uses the first drive letter beyond the **lastdrive=** command to begin mapping remote file shares to drive letters (for example, if **lastdrive=p**, then NetWare uses drive **q** as the logon drive). NetWare allows mapping of drives below the drive letter designated by the **lastdrive=** command, but NetWare prompts the user for verification of this action if he or she requests to map a drive less than or equal to the drive specified by the **lastdrive=** command.

Note If you are using any NetWare login scripts that assume a **lastdrive=** setting other than **p**, you may need to modify your scripts to have them work correctly.

Novell ODI Drivers

Novell ODI drivers are not compatible with Windows for Workgroups and should not be used. Windows for Workgroups supports the ability for NetBEUI and IPX to use the same network adapter card only through the NDIS interface. Although some ODI to NDIS mapping drivers may exist that allow ODI drivers to run with NDIS-compliant protocol stacks, Microsoft does not support this configuration for use with Windows for Workgroups.

Novell NetBIOS

If you were running Novell NetBIOS before you set up Windows for Workgroups, the Setup program attempts to remove it when you set up Windows for Workgroups. The following procedure describes how to install Novell NetBIOS after you have set up Windows for Workgroups.

Note This example assumes that you have one network adapter running Microsoft NetBEUI. For information about the settings used in this example and how to change them, see the "SYSTEM.INI: System Initialization File" section in chapter 6 of this manual.

To install Novell NetBIOS after you have set up Windows for Workgroups:

1. Add the following line to the **[network]** section of your SYSTEM.INI file:

exclude=0

This entry will prevent Windows for Workgroups from attempting to run over Novell's NetBIOS.

2. Add the following line to the **[386Enh]** section of your SYSTEM.INI file:

V86ModeLANAs=0

This entry defines the real-mode LANAs that are in use by NetBIOS.

3. If you plan to start NetBEUI before you start Windows (that is, you load the real-mode NetBEUI transport instead of using the VNB.386 VxD), edit this line to read as follows:

V86ModeLANAs=0,1

This entry defines the real-mode LANAs that are in use by NetBIOS.

4. Add the following line to the **[network]** section of your SYSTEM.INI file:

LANAs=0,1

This entry defines the real-mode LANAs that are in use on the workstation.

5. In your AUTOEXEC.BAT file, make sure that the line that loads the Novell NetBIOS.EXE comes after the **net start** command but before any other components of the Windows for Workgroups networking software (for example, before a **net logon** line that loads the networking software, if present in your AUTOEXEC.BAT).

Additional Information About Novell NetWare

The following information will assist you in configuring your Windows for Workgroups workstation to successfully interoperate with Novell NetWare:

- Do not try to log in, log out, attach to, or detach from a Novell server from the MS-DOS Prompt within Windows for Workgroups. You should log in before starting Windows for Workgroups and attach to or detach from servers by using File Manager or Print Manager.
- If you have problems connecting to a Novell server, make sure the Adapter Media Type setting in Control Panel matches your Novell NetWare software configuration. For more information, see Chapter 6, "Troubleshooting," in *Getting Started*.
- The NetWare SHELL.CFG or NET.CFG file must be in the same directory as NETX.COM in order for NETX to use the specified information.

- By default, NetWare allows you access to only 40 files at a time. When you are running applications with Windows, you can exceed this limit quickly. If so, you might see unexpected error messages. To increase the file access limit, add the following line to your SHELL.CFG or NET.CFG file:

file handles = 60

You should also add the following to your CONFIG.SYS file:

files = 60

- If you get unusable output when printing to a NetWare server (such as incorrect fonts, garbled text, invalid page breaks, or blank lines), you may need to edit your print-job configuration. Try using the NetWare PRINTCON utility to set the Auto Endcap and Enable Timeout options to No.
- When running Windows for Workgroups in 386 enhanced mode, swapping files to a drive on a NetWare 286 server might make starting Windows for Workgroups take up to a minute. For information about controlling the location of your swap file, see Chapter 11, "Managing Memory and Performance," in the *Microsoft Windows for Workgroups User's Guide*.
- A NetWare file server does not include the directory entries dot (.) and double dot (..) as MS-DOS does. However, the NetWare shell (version 3.01 or higher) can emulate these entries when applications attempt to list the files in a directory. If you have problems listing files or deleting directories, turning on the Show Dots feature will help. Add the following line to your SHELL.CFG or NET.CFG file:

show dots = on

Turning on Show Dots will cause problems with earlier versions of some 286-based NetWare utilities, such as BINDFIX.EXE and MAKEUSER.EXE. Make sure you upgrade these utilities if you upgrade your NetWare shell. For more information, contact your Novell dealer.

- Redirected drives can look different in Windows for Workgroups from the way they did without Windows for Workgroups. For example, a device mapped as FileServerName/Volume:Directory will appear in Windows for Workgroups as FileServerName/Volume:\, showing the root of the file server's shared volume rather than the subdirectory. You can correct this with the MAP ROOT entry, as explained in the following information.

- In some cases, Windows-based applications may change the default directory on a drive. This might cause problems if, for example, your path includes the current directory on a network drive (that is, your path includes a drive letter without a specific directory, such as z: instead of z:\system).

You can prevent this problem by using MAP.EXE and LOGIN.EXE versions 3.0 and later to set up false roots, or by choosing the MAP ROOT function when mapping a connection in File Manager. This feature simulates the MS-DOS **subst** command, which sets the root of a given drive to a directory designated by the user instead of to the true root of the volume.

For example, suppose you normally mapped drive F to the HOME\TERRI directory on the SERVER\SYS volume, and then included F: in your path. You would do this by including the following command in your AUTOEXEC.BAT file, or by typing the command before starting Windows:

```
map f:=server\sys:home\terri
```

The default directory on drive F would then be HOME\TERRI. To prevent Windows from changing this, you would replace the command above with the following:

```
map root f:=server\sys:home\terri
```

This command would make the directory HOME\TERRI appear to be the root of drive F.

- If you are running Windows for Workgroups in 386 enhanced mode, you can adjust the way Windows handles your network drive mappings by using the Network option in Control Panel.

Usually, when you quit Windows for Workgroups, all of your drive mappings are restored to the way they were before you started Windows for Workgroups, and all changes you made while running Windows are lost. If you clear the Restore Drives option in the settings dialog box for NetWare, the mappings you made in Windows for Workgroups will remain when you quit Windows for Workgroups.

Typically, each instance of MS-DOS Prompt you start from Windows for Workgroups has its own set of drive mappings. Changes you make in one instance do not affect another. If you set the **NWShareHandles=** option, drive mappings will instead be global, and changes to the mappings or current drive made in one instance of MS-DOS Prompt will affect all other applications. If you are running a NetWare 286 server, setting

NWShareHandles= true increases the number of workstations that can be connected to the server before the server runs out of available connections.

See the Help available in the Network dialog box in Control Panel for more information on these options.

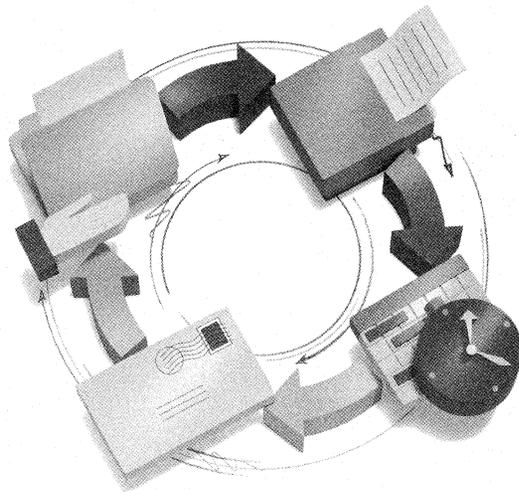
- If you have applications or files that use extended characters (ASCII characters above 127) and you have problems viewing or using them, try adding the following line to your SHELL.CFG or NET.CFG file:

special uppercase = on

- When connecting or disconnecting network resources from File Manager, Print Manager, or Control Panel, pay attention to the state of the Permanent check box. If this option is selected when you make a connection, the connection is automatically made each time you restart Windows for Workgroups. To stop reconnecting, disconnect with the Permanent check box selected. If the check box is not selected when you disconnect, the connection is removed for the current Windows session but is reconnected when you restart Windows for Workgroups.
- After Windows for Workgroups is installed, applications that use named pipes to communicate with NetWare servers may no longer work correctly. Applications that use named pipes to communicate with Microsoft LAN Manager servers will work with Windows for Workgroups.

Many network applications that use named pipes can also be configured to use another method of communicating on the network, such as IPX or SPX. Contact your network-application vendor for additional details.

Configuring Windows for Workgroups



Part

4

Configuring Windows for Workgroups

Chapter 9 Tips for Configuring Windows for Workgroups

9-1

Tips for Optimizing Performance.....	9-2
Optimizing Server/Application Performance.....	9-16

Tips for Configuring Windows for Workgroups

This chapter presents several tips and information for configuring Microsoft Windows for Workgroups 3.1 for gaining optimal performance.

Related information

- *Microsoft Windows for Workgroups User's Guide*: Chapter 9, "Customizing Windows for Workgroups;" Chapter 11, "Managing Memory and Performance."
- *Microsoft Windows for Workgroups Resource Kit*: Chapter 6, "Windows for Workgroups Initialization Files."
- *Microsoft Windows Resource Kit*: Chapter 5, "Windows 3.1 and Memory Management."

Contents of This Chapter

Tips for Optimizing Performance.....	9-2
Checklist for Optimizing Performance.....	9-2
Trading Speed for Hard Disk Space.....	9-7
Optimizing Use of the UMBs	9-8
Adjusting EMM386 for Optimal Performance	9-11
Tips for Using Third-Party Memory Managers	9-14
Tips for Using Disk-Compression Software	9-16
Optimizing Server/Application Performance.....	9-16

Tips for Optimizing Performance

This section presents a summary of tips for achieving the best performance for Windows for Workgroups 3.1, plus specific information about using EMM386.EXE.

Before making changes to your system configuration, create a backup disk, as described in “Creating a ‘Clean Boot’ for Troubleshooting” in Chapter 14, “Troubleshooting Windows for Workgroups.” For information that will help you solve specific performance problems with your system, see the “Troubleshooting Windows for Workgroups Configuration” section in Chapter 14.

Checklist for Optimizing Performance

Here is a summary of items related to optimizing your hardware and software configuration for both Windows-based applications and MS-DOS-based applications running under Windows for Workgroups 3.1.

For information about the parameters for HIMEM.SYS, EMM386.EXE, SMARTDRV.EXE, and RAMDRIVE.SYS discussed in this chapter, see Chapter 11, “Managing Memory and Performance,” in the *Microsoft Windows for Workgroups User’s Guide*.

Tip To see a list of switches that can be used with Microsoft system drivers that have .EXE filename extensions, at the command prompt, type the driver name and the /? switch, then press ENTER. For example, type **emm386 /?** at the command prompt to see a list of the switches you can use with EMM386.EXE.

Configuring Hardware for Optimal Performance

- Install additional extended memory.
- Install the largest hard disk that you can afford, as discussed in “Trading Speed for Hard Disk Space” later in this chapter.
- Set the optimal hard disk interleave for your system, using third-party software such as Steve Gibson's SpinRite. Hard disks are often formatted with the wrong interleave at the dealer or factory. Some utilities can correct your interleave without reformatting the hard disk.
- If your system has a memory expansion board that can be configured as either expanded or extended memory, configure all of this memory as extended memory, then use EMM386.EXE to emulate expanded memory only as needed by MS-DOS-based applications that require such memory

to run *outside* of Windows for Workgroups. In such a case, place the expanded memory driver first in CONFIG.SYS, followed by the lines that load HIMEM.SYS and EMM386.EXE as in this example:

```
device=c:\emmbord.sys
device=c:\dos\himem.sys
device=c:\dos\emm386.exe ram 1024 x=C000-C7FF
```

For information about configuring the memory in your add-on board, see its documentation. Some expanded memory boards may be incompatible with EMM386.EXE.

Configuring MS-DOS for Optimal Performance

- Upgrade to MS-DOS version 5.0 (and above) and load MS-DOS in the high memory area (HMA) by setting **dos=high** or **dos=high,umb** in CONFIG.SYS. For more information, see “Loading MS-DOS in the HMA” later in this chapter.
- Make sure the most recent versions of HIMEM.SYS, EMM386.EXE, RAMDRIVE.SYS, and SMARTDRV.EXE are stored in the directories specified in the command lines that load these drivers in your CONFIG.SYS and AUTOEXEC.BAT files. The versions of these files provided with Windows for Workgroups are the latest, and are placed in the directory where you installed Windows for Workgroups. (Placing any of these drivers in the root directory won't improve performance.)
- Load HIMEM.SYS in CONFIG.SYS and make sure the line that loads HIMEM.SYS comes before commands that load any other applications or drivers that use extended memory.
- Load SMARTDRV.EXE in AUTOEXEC.BAT and allocate the largest amount of memory possible. The SMARTDrive disk-caching driver can produce the best Windows for Workgroups 3.1 performance improvement.
- Set **files=30** in CONFIG.SYS unless you have a software application that requires a higher setting.
- Set **buffers=10** in CONFIG.SYS if you use SMARTDRV.EXE. Using a higher number of buffers with SMARTDrive will decrease efficiency.

If you are not using SMARTDrive, set **buffers=20** in CONFIG.SYS. More buffers may improve disk access times, but will use more conventional memory.

- Install RAMDrive and use the specified drive letter as the value for the TEMP environment variable if you have a diskless workstation or if you have limited hard disk space and ample extended memory (more than 6 MB). Performance improves for Print Manager and for applications that frequently create .TMP files because access is faster for a RAM disk than for a physical disk.

- Load EMM386.EXE if you are running MS-DOS-based applications that require expanded memory or if you want to allocate UMBs for loading memory-resident TSRs and drivers. For more information, see “Optimizing Use of the UMBs” later in this chapter.
- Load only the necessary TSRs, drivers, or software programs. Wherever possible, load TSRs and device drivers into the UMBs.
- If the environment space is set by the **shell** command in CONFIG.SYS, you can specify a smaller environment.
- Remove any commands for mouse drivers in AUTOEXEC.BAT and CONFIG.SYS if you only use the mouse in Windows for Workgroups, and don’t want mouse support while running MS-DOS-based applications in 386 enhanced mode.
- If you are running in 386 enhanced mode and have the MS-DOS **share** TSR loaded in your AUTOEXEC.BAT and you don't need to support file sharing from MS-DOS outside of Windows for Workgroups (that is., you start Windows for Workgroups and only run MS-DOS-based applications from within the Windows for Workgroups environment), you can remove or remark out **share** from your AUTOEXEC.BAT—this will free up some additional conventional memory. The VSHARE.386 virtual device driver provides all of the MS-DOS share functionality to Windows for Workgroups and MS-DOS-based applications executed from within the Windows for Workgroups environment (if you are using DOS 4.x and have a hard disk with a partition defined larger than 32MB, leave the SHARE.EXE TSR in your AUTOEXEC.BAT file to support the large disk partitions).

Maintaining Optimal Hard Disk Performance

- Delete unnecessary application and system files, including .BAK and .TMP files. Make sure that Windows for Workgroups is not running when you delete .TMP files.
- Run **chkdsk** frequently to find lost clusters. If lost chains or clusters are detected, run **chkdsk /f** and delete *.CHK files.
- Use a utility program regularly to optimize or defragment your hard disk. A fragmented hard disk greatly affects Windows for Workgroups performance in a negative way, especially if SMARTDrive is installed or if you are using a temporary swap file. Make sure you do not run a disk optimizing utility from within Windows for Workgroups—run the utility from the MS-DOS command prompt after exiting Windows for Workgroups.

Configuring Windows for Workgroups for Optimal Performance

- Use a color or a pattern instead of wallpaper for the desktop background if you need to free memory for running applications.
- For standard mode Windows for Workgroups, set the application swap file to the fastest hard-disk drive by setting the **SwapDisk=** entry in the **[NonWindowsApp]** section of SYSTEM.INI.
- For 386 enhanced mode, create a permanent swap file on your fastest hard-disk drive by choosing the 386 Enhanced icon in Control Panel and setting the options in the Virtual Memory dialog box.

The only reason not to use a permanent swap file is if hard disk space is at a premium. For details about permanent swap files, see Chapter 5, “Windows and Memory Management” of the *Microsoft Windows Resource Kit*.

- If you don't create a permanent swap file for 386 enhanced mode, set your temporary swap file to the fastest hard-disk drive by setting a value for **PagingDrive=** or **PagingFile=** in the **[386Enh]** section of SYSTEM.INI.
- Run setup and choose the lowest-resolution display driver that will meet your needs. For VGA-compatible displays that can use an extended mode driver, such as Video 7 or 8514, using the standard VGA driver ensures faster display performance (but lower resolution and less color support).
- The use of a screen saver can degrade performance on a workstation that is sharing resources. If you will be sharing resources on your computer and your workstation will be left unattended, do not install a screen saver (or specify a screen saver such as Marquee, which is not CPU intensive) to provide the best resource sharing performance.
- If the workstation you are configuring is only sharing resources (for example, the computer is configured as a “dedicated” file and/or print server), adjust the Performance Priority slider bar as appropriate to provide more CPU time to the sharing of resources. If you are sharing a locally connected printer, Print Manager must be running on the workstation and is considered an application—you may need to adjust the Performance Priority slider bar to provide an optimal performance level depending on the amount of activity on the given workstation.
- If you are using a separator page for print jobs on your computer, choose a less complex separator for faster printing (for example, choose the Simple Separator Page in the Separator Pages dialog box, accessible from the Separator Pages selection on the Options menu in Print Manager).

Optimizing for MS-DOS-Based Applications

- Allocate the expanded and extended memory required by MS-DOS-based applications by using the PIF Editor to create a custom PIF. For more information, see Chapter 8, “PIFs and PIF Editor,” of the *Microsoft Windows Resource Kit*.
- Turn off all entries in the Monitor Ports option in the PIF Editor Advanced dialog box. These options are not required for most MS-DOS-based applications.
- Set **FileSysChange=no** in the **[386Enh]** section of SYSTEM.INI to avoid sending update messages to File Manager from MS-DOS-based applications.
- Set **NoEMMDriver=yes** in the **[386Enh]** section of SYSTEM.INI if your MS-DOS-based applications do not need expanded memory. This ensures that the most possible memory is available in virtual machines.

Optimizing for TSRs

- Include only the TSRs in AUTOEXEC.BAT that are required for the network or that must be present for other purposes.
- If a TSR is required only with a single MS-DOS-based application, start the TSR in the virtual machine after launching that application. For more information about creating batch files for running TSRs from Windows, see Chapter 7, “Setting Up Non-Windows Applications,” of the *Microsoft Windows Resource Kit*.
- For all TSRs that you want to be available in Windows for Workgroups, add their .EXE names to the file WINSTART.BAT, which runs whenever Windows for Workgroups runs in 386 enhanced mode. These TSRs aren't available in the virtual machines (VMs), leaving more room in conventional memory for the MS-DOS-based applications. For more information about WINSTART.BAT, see Chapter 11, “Managing Memory and Performance,” in the *Microsoft Windows for Workgroups User's Guide*.

The following table shows where a TSR is available (and using memory), depending on where it is launched.

<i>Where TSR is launched?</i>	<i>TSR visible in WFW?</i>	<i>TSR visible in virtual machines?</i>
From MS-DOS	Yes	Yes, all virtual machines
From WINSTART.BAT	Yes	No
In a single virtual machine	No	Only that virtual machine

Optimizing for Windows for Workgroups on a Network

- Do not use a network drive for a temporary swap file.
- If a user has problems running Windows for Workgroups or other applications with network software loaded in the upper memory area or in the high memory area, try disabling the network's HMA option. Consult your network documentation for more information.
- Some memory managers, such as EMM386, 386MAX, and QEMM, can load networks and other device drivers in the upper memory area. If the network driver is loaded in the upper memory area and a user has problems running Windows for Workgroups, try loading the network in conventional memory.

Trading Speed for Hard Disk Space

To optimize Windows for Workgroups performance, you must make trade-offs between speed and hard disk space, depending on your system's hardware capabilities. In either case, make sure that as much free memory as possible is available.

To ensure maximum free memory in your system

- Remove unnecessary TSRs from AUTOEXEC.BAT and CONFIG.SYS.
- Load MS-DOS into high memory and use all possible elements of MS-DOS 5.0 (and above) memory management, such as using EMM386.EXE to manage UMBs.

To ensure optimal speed performance

- Free as much conventional and extended memory as possible.
- Install SMARTDrive for disk caching.
- Set the TEMP environment variable to the fastest hard disk or to a RAM drive.
- Create a permanent swap file to use in 386 enhanced mode, and turn on 32-bit disk access if supported on your workstation.
- Set **SwapDisk=** in the **[NonWindowsApp]** section of SYSTEM.INI to the hard disk with the most free space.
- Use third-party software to organize the location of files on your hard disk so that the Windows for Workgroups files are located near the file allocation table (FAT).

To ensure maximum hard disk space

- Create a RAM drive to use as the location specified by the TEMP environment variable.
- Delete unnecessary files after quitting Windows for Workgroups.
- Decrease the size of your permanent swap file, or use a temporary swap file in 386 enhanced mode. You can prevent swapping by typing **none** in the New Settings Type field of the Virtual Memory dialog box, or by setting **paging=no** in the [386Enh] section of SYSTEM.INI.

Optimizing Use of the UMBs

Related information

- *Microsoft MS-DOS User's Guide and Reference*: "Running Programs in the Upper Memory Area" in Chapter 12
- *Microsoft Windows for Workgroups Resource Kit*: "Troubleshooting EMS Memory Problems" in Chapter 14, "Troubleshooting Windows for Workgroups"
- *Microsoft Windows Resource Kit*: "Expanded Memory: A Technical Discussion" in Chapter 5, "Windows 3.1 and Memory Management"

On 80386 and higher PCs that use MS-DOS 5.0 (and above), you can free conventional memory by loading memory-resident programs such as device drivers, TSRs, and network software into upper memory blocks (UMBs). You can optimize the way your system uses UMBs by following the guidelines in this section:

- Load MS-DOS 5.0 (and above) in the HMA with the **umb** switch.
- Load EMM386.EXE with the **noems** or the **ram** switch.
- Find the best order for loading drivers and programs into UMBs.
- Adjust the command line for EMM386.EXE in CONFIG.SYS to set aside fewer UMBs for expanded memory. This provides more room for loading other programs into UMBs. For details about how to do this, see "Adjusting EMM386 for Optimal Performance" later in this chapter.

Loading MS-DOS in the HMA

You must load MS-DOS 5.0 (and above) in the HMA and use the **umb** switch to link UMBs to conventional memory if you want to load programs into UMBs.

To load MS-DOS 5.0 (and above) in the HMA

- Add these lines to CONFIG.SYS, in this order:

```
device=<path>\himem.sys
device=<path>\emm386.exe noems
dos=high,umb
```

The parts of MS-DOS 5.0 (and above) that are loaded in the HMA include most of the kernel and COMMAND.COM, code pages, and the MS-DOS buffers (from the **buffers=** command in CONFIG.SYS). To remain compatible with software created for earlier versions of MS-DOS, about 19K of MS-DOS 5.0 code stays in conventional memory.

You can only load one program at a time in the HMA. So, for example, if you load MS-DOS 5.0 (and above) into the HMA, you can't load a network redirector into it as well. If you must choose what to load in the HMA, always choose to load MS-DOS 5.0 (and above), because it uses the HMA efficiently and leaves little behind in conventional memory.

Note Use the **device** command to load HIMEM.SYS and EMM386.EXE, because loading them with **devicehigh** has no effect.

Loading EMM386.EXE to Use UMBs

After loading HIMEM.SYS in CONFIG.SYS, load EMM386.EXE with the **ram** switch or the **noems** switch, plus any other parameters required for your system. The **ram** switch provides both UMBs and a 64K EMS page frame; **noems** provides UMBs only and no page frame. If none of your applications require EMS, then use **noems**, which will provide an extra 64K in UMBs.

If you load EMM386 with the **noems** switch and find that some MS-DOS-based applications fail to run or display out-of-memory messages, it may be because they require expanded memory to run. If this happens, replace the **noems** switch with the **ram** switch to make expanded memory available again.

Setting the Load Order for Drivers in UMBs

You must experiment to find the best order for loading drivers and TSRs into UMBs.

To find the best order for loading drivers into UMBs:

1. In CONFIG.SYS, load the largest device drivers with the **devicehigh** command. Use the **device** command to start any drivers that are smaller than the largest memory-resident program loaded from AUTOEXEC.BAT.
2. In AUTOEXEC.BAT, start the largest memory-resident programs with the **loadhigh** command.
3. After rebooting, use the MS-DOS **mem /c** command to see what is loaded into UMBs and what is in conventional memory. If everything didn't fit into UMBs, change the order in which the programs are loaded in CONFIG.SYS and AUTOEXEC.BAT.

With **devicehigh** and **loadhigh**, each device and TSR is loaded into the largest available slot, so the order in which programs are loaded can affect how many programs can fit into the UMBs. Also, how and when a program allocates memory determines what size UMB it needs. Some programs need a UMB as large as the program size while running, and others need a UMB as large as the program file size.

The following table lists some Microsoft device drivers and memory-resident programs that can run in UMBs. This table shows whether the program's file size or size while running determines the UMB size the program needs.

Program	UMB size needed	Program	UMB size needed
ANSI.SYS	File size	MOUSE.COM	File size
DISPLAY.SYS	Size while running	MOUSE.SYS	File size
DOSKEY.COM	Size while running	PRINTER.SYS	Size while running
DRIVER.SYS	File size	RAMDRIVE.SYS	Size while running
EGA.SYS	File size	SETVER.EXE	File size
FASTOPEN.EXE	Size while running	SHARE.EXE	Size while running
MIRROR.COM	Size while running		

Note You cannot use **loadhigh** or **devicehigh** to load HIMEM.SYS or EMM386.EXE. SMARTDRV.EXE, provided with Windows for Workgroups for Workgroups, will load itself into high memory automatically if it runs on MS-DOS 5.0 (and above) and if there is enough free space in the upper memory (UMB) area.

If your mouse comes with both MOUSE.SYS and MOUSE.COM files, you can load the mouse driver at any point after HIMEM.SYS and EMM386.EXE. Use **devicehigh** in CONFIG.SYS to load MOUSE.SYS, or use **loadhigh** in AUTOEXEC.BAT to load MOUSE.COM.

For example, suppose you have an 80386 computer with 3 MB of extended memory that you want to configure as follows:

- 512K of expanded memory for a MS-DOS-based CAD application
- As much free conventional memory as possible

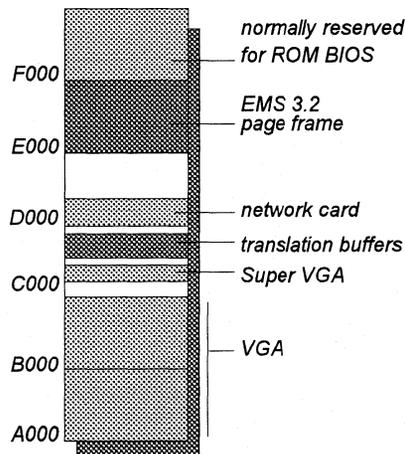
Your CONFIG.SYS file should contain the following commands:

```
device=c:\dos\himem.sys
device=c:\dos\emm386.exe 512 ram
dos=high,umb
devicehigh=c:\dos\mouse.sys
```

Figure 9.1

Example of UMBs in the upper memory area:

A VGA or V7VGA display adapter, a network card at D000-D7FF, and an EMS 3.2 page frame at E000-EFFF



For more information about determining memory requirements and load order for programs, see *Microsoft MS-DOS Getting Started* and the *Microsoft MS-DOS User's Guide and Reference*.

Adjusting EMM386 for Optimal Performance

The EMM386.EXE driver takes a conservative approach when scanning UMBs for free space because it must remain compatible with software that was created for earlier versions of MS-DOS. For example, EMM386.EXE does not map over any unused ROM. If you have space below C800 or above DFFF that you know is unused and you want it used for UMBs, you must explicitly instruct EMM386.EXE to include it by using the `i=` switch to specify a range of segment addresses.

For more information about EMM386, see the *Microsoft MS-DOS User's Guide and Reference*, and see Chapter 11, "Memory Management and Performance," in the *Microsoft Windows for Workgroups User's Guide*.

Adjusting EMM386 to Provide More UMBs

You can adjust EMM386 to allocate space in UMBs, especially if you start EMM386 with the **ram** switch to make expanded memory available to programs. First, follow the steps in Procedure 1 below. If you still can't load programs and device drivers into UMBs, complete Procedure 2 below.

Procedure 1

To allocate additional space for use as UMBs

1. Create a backup boot disk, as described in the "Creating a 'Clean Boot' for Troubleshooting" section in Chapter 14, "Troubleshooting Windows for Workgroups," in this manual.

2. Add the **i** switch in front of the **ram** or the **noems** switch in the **device=emm386.exe** line of CONFIG.SYS.

The **i** switch specifies a range of addresses to be allocated in UMBs. The value you specify for the **i** switch depends on your computer and monitor type.

- If your computer isn't an IBM Personal System/2® (PS/2) and doesn't have a monochrome monitor, add the **i=e000-efff** and the **i=b000-b7ff** switches before the **ram** or **noems** switch—for example:

```
device=c:\dos\emm386.exe i=e000-efff i=b000-b7ff ram
```

- If your computer isn't a PS/2 and has a monochrome monitor, add the **i=e000-efff** switch before the **ram** or **noems** switch—for example:

```
device=c:\dos\emm386.exe i=e000-efff ram
```

- If your computer is an IBM PS/2 and doesn't have a monochrome monitor, add the **i=b000-b7ff** switch before the **ram** or **noems** switch—for example:

```
device=c:\dos\emm386.exe i=b000-b7ff ram
```

If your computer is an IBM PS/2 with a monochrome monitor, see Procedure 2.

3. If your computer fails when you restart it after this change, the memory range you specified for EMM386 is probably being used by hardware or video display drivers. In this case, use your backup boot disk to restart your computer. Then remove the **i** switches you added for EMM386 in CONFIG.SYS.

If you start EMM386 with the **ram** switch, follow Procedure 2.

4. After your computer starts, use the MS-DOS **mem /c** command to check whether the programs were successfully loaded into UMBs.

If you still can't load programs into UMBs, your system might be using much of UMB space for hardware code. Some computers use UMBs for ROM shadowing, which can improve your computer's speed. (In ROM shadowing, the computer copies hardware code from the slower read-only memory into faster RAM.) To find out more about how your computer uses UMBs, see the documentation that came with your system.

If you can't load programs into UMBs, and you are starting EMM386 with the **ram** switch, carry out Procedure 2.

Note Use Procedure 2 only if your MS-DOS-based EMS programs use expanded memory according to LIM EMS version 4.0. Programs that follow LIM EMS 3.2 need the full 64K page frame and are unable to use expanded memory if you follow this procedure.

Procedure 2 **To set aside fewer UMBs for expanded memory**

1. Type **emm386** at the command prompt.

The information that appears shows current memory-management activities. Find the line that reads "Page frame segment" and write down the hexadecimal address that appears on that line—for example:

```
Page frame segment . . . E000 H
```

2. Add the **p0** switch in front of the **ram** switch in the **device=emm386.exe** line of CONFIG.SYS.

The **p0** switch should specify the address you found in Step 1. For example, if the page-frame segment was E000h, the command would be:

```
device=c:\dos\emm386.exe p0=e000 ram
```

3. If your computer fails when you restart it, insert your backup boot disk in drive A and restart your computer. Then remove the **p0** switch you added to CONFIG.SYS in Step 2. Restart your computer and repeat Steps 1 and 2, checking the page-frame segment address carefully.
4. After your computer starts, use the MS-DOS **mem /c** command to check whether the programs were successfully loaded into UMBs.

If your programs still don't run in UMBs, contact Microsoft Product Support Services for assistance.

Freeing Conventional Memory in 386 Enhanced Mode

When you run MS-DOS-based applications in 386 enhanced mode, Windows for Workgroups allocates translation buffers to pass data between the applications and drivers running in real mode. (For more information, see

Chapter 5, "Windows 3.1 and Memory Management," of the *Microsoft Windows Resource Kit*.) On a stand-alone computer, Windows for Workgroups allocates two translation buffers for each virtual machine (8K total); on a networked computer, Windows for Workgroups allocates six buffers (24K total).

If Windows for Workgroups finds space, it places the translation buffers in UMBs; otherwise, they are placed in conventional memory, where they take up space in every virtual machine.

To find out if you still have UMBs free while running EMM386, at the command prompt type `mem /p` to see what segments are free in the hexadecimal addresses between A000 and FFFF. Then add a `x=` switch to the EMM386 line in CONFIG.SYS to exclude the unused UMB ranges.

If you don't have at least 28K of free in the UMB area, you can still partially optimize the use of this area. Because each translation buffer is 4K, each buffer you exclude will save conventional memory. For example, this line will let Windows for Workgroups load one buffer into the UMB area, freeing 4K of conventional memory:

```
device=c:\dos\emm386.exe noems x=df00-dfff
```

Excluding memory, however, can change how MS-DOS 5.0 (and above) loads programs into UMBs, so you may need to experiment to get the optimal configuration.

Tips for Using Third-Party Memory Managers

For complete compatibility, Windows for Workgroups 3.1 and MS-DOS 5.0 (and above) both provide HIMEM.SYS and EMM386.EXE as memory management drivers. The overriding objective Microsoft has for these drivers is to avoid hanging any computer during or after memory configuration. Optimization, while important, is secondary.

You can also use a third-party memory manager to manage UMBs in an 80386 computer. Such memory managers have different objectives, and there are pros and cons to using them in place of the built-in Microsoft memory managers.

To allocate UMBs through a third-party memory manager, you need both a UMB provider and a UMB loader.

- A UMB provider is the memory manager that makes UMBs available. For MS-DOS 5.0 (and above), the UMB provider is the combination of EMM386.EXE plus HIMEM.SYS. In QEMM386 and 386MAX, for example, the UMB providers are QEMM386.SYS and 386MAX.SYS, respectively.
- A UMB loader loads a program into the proper UMB. The UMB loader in MS-DOS 5.0 (and above) is in the MS-DOS kernel, accessed through the **devicehigh** and **loadhigh** commands. The equivalents in 386MAX are 386LOAD.SYS and 386LOAD.COM; in QEMM386, they are ADHI.SYS and LOADHI.COM.

The arguments for using third-party memory managers include these:

- Most third-party memory managers have a wider range of options for loading into UMBs. For example, you can specify a particular UMB for loading a program.
- Third-party memory managers typically require 3K to 5K of conventional memory, while HIMEM.SYS with EMM386.EXE may take up to 10K. However, some memory managers also require overhead in UMBs, but neither HIMEM.SYS nor EMM386.EXE require any.

The arguments against using third-party memory managers include these:

- Third-party memory managers can be more difficult than EMM386.EXE to configure properly, and they are often geared toward more sophisticated users.
- Third-party memory managers are supplemental to HIMEM.SYS and EMM386.EXE, so they require additional costs and resources to support. Unless you absolutely require 100 percent optimization, a third-party memory manager may be superfluous.

If you decide to use a third-party memory manager, follow these guidelines:

- Don't use either HIMEM.SYS or EMM386.EXE.
- Use **dos=high** to load MS-DOS 5.0 (and above) into the HMA.
- Don't use the **dos=umb**, **devicehigh**, or **loadhigh** commands.
- Don't use the third-party commands that relocate the MS-DOS buffers into UMBs. MS-DOS 5.0 (and above) automatically puts its buffers into the HMA, saving UMBs for other programs.
- Use both the third-party UMB provider (such as 386MAX.SYS) and the third-party UMB loader (such as 386LOAD.SYS and 386LOAD.COM).

- Use the third-party commands to relocate MS-DOS file handles and drive space (from the **files** and **lastdrive** commands in your CONFIG.SYS) into UMBs.

You can also use third-party memory drivers as alternatives to HIMEM.SYS, SMARTDRV.EXE, and RAMDRIVE.SYS. However, the memory drivers provided with Windows for Workgroups 3.1 have been specifically designed and tested to provide maximum compatibility between MS-DOS 5.0 (and above) and Windows for Workgroups 3.1.

Tips for Using Disk-Compression Software

Windows for Workgroups will work properly with the popular disk-compression software available on the market (such as STAC Electronics' Stacker). The following information will help to ensure your use of Windows for Workgroups with these products is successful.

Using a Virtual Memory Swap File with Disk-Compression Software

In 386 enhanced mode, the swap file is used to provide support for virtual memory to supplement the physical RAM in the workstation. Some disk compression software may allow a swap file to be installed on a compressed logical disk volume, but this is not recommended. For the best level of performance when running Windows, set the swap file to point to a non-compressed disk volume. Telling Windows to use a swap file on a compressed volume will affect performance (depending on the frequency in which Windows may need to access virtual memory) and may result in the compressed volume becoming fragmented quickly, also affecting performance.

If the disk compression software you are using didn't provide enough free space on the physical drive for uncompressed file storage, check your user's manual for information on reducing the size of the compressed logical drive.

Optimizing Server/Application Performance

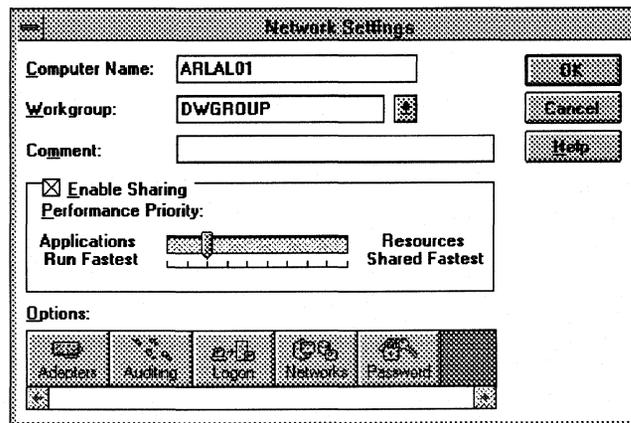
The Network option in Control Panel allows you to change the Performance Priority of your workstation when running in 386 enhanced mode. You can set Performance Priority so applications can run faster on the local workstation or to provide more priority to the components of Windows for Workgroups. Added priority gives better response to remote users when accessing the shared resources on your workstation. The Performance Priority setting only has an effect on the local Windows for Workgroups workstation when another user is connected to it. (That is, when no users are connected to your workstation,

Windows for Workgroups adjusts the performance priority so that applications can run the fastest.) The Performance Priority option isn't available to workstations running in standard mode. Resource sharing is only available to a workstation running in 386 enhanced mode where Windows for Workgroups server components are provided as a series of virtual device drivers (VxDs).

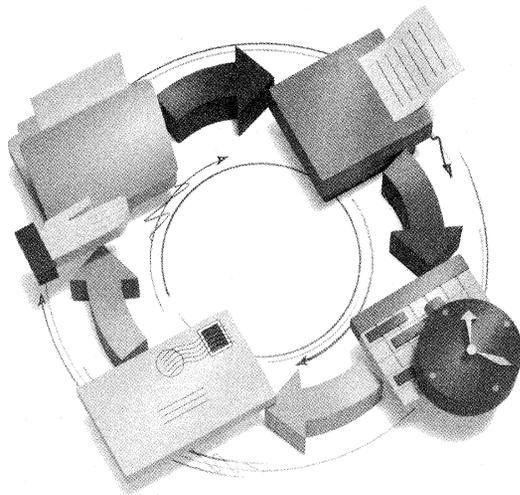
See the discussion of priority values in the "Windows for Workgroups Priority Values" section of Chapter 7 in this manual, for more information.

Figure 9.2

Network Settings
Option from Control
Panel - 386
enhanced mode



Using Windows for Workgroups



Using Windows for Workgroups

Chapter 10 **New and Updated Accessories** **10-1**

Enhancement to Common Dialogs	10-2
ClipBook Viewer	10-6
Net Watcher	10-9
WinMeter	10-12
Hearts	10-14
File Manager	10-14
Print Manager	10-17

Chapter 11 **Network Dynamic Data Exchange** **11-1**

About Network DDE	11-3
Overview of Dynamic Data Exchange (DDE)	11-3
Overview of Network DDE	11-11
Network DDE Security	11-14
Examples of How Network DDE Works.....	11-18
Using Network DDE with Application Macro Languages	11-23
Questions and Answers about Network DDE	11-24

Chapter 12 **Mail** **12-1**

About Mail	12-3
The Mail Client	12-3
The Mail Postoffice.....	12-8
Interface Between the Mail Client and Postoffice	12-11
Customizing the Mail Client.....	12-13
Integrating Mail and Other Windows-Based Applications.....	12-27
Differences Between Windows for Workgroups Mail and Mail 3.x	12-28
Extending the Reach of Mail	12-29
Tips for Using Mail	12-35
Questions and Answers about Mail	12-38

Overview of Schedule+ Architecture.....	13-2
Custom Message Types for Schedule+	13-7
Schedule+ Interoperability.....	13-9
Extending the Reach of Schedule+	13-17
Questions and Answers about Schedule+	13-18

**Chapter
10**

New and Updated Accessories

This chapter provides a discussion of special topics related to some of the new and updated accessories provided with Windows for Workgroups, including ClipBook Viewer, Net Watcher, WinMeter, Hearts, File Manager, and Print Manager. A discussion of the enhancements made to the Windows Common Dialog support library is also provided in this chapter. Information about Chat, a new accessory that permits interactive communication between two Windows for Workgroups workstations, is discussed in the *Windows for Workgroups User's Guide*.

Related information

- *Windows Resource Kit*: Chapter 11, "Compound Documents in Windows 3.1."
- *Windows for Workgroups Resource Kit*: Chapter 11, "Network Dynamic Data Exchange."

Contents of This Chapter

Enhancement to Common Dialogs.....	10-2
ClipBook Viewer	10-6
Net Watcher	10-9
WinMeter	10-12
Hearts.....	10-14
File Manager	10-14
Print Manager	10-17

Enhancement to Common Dialogs

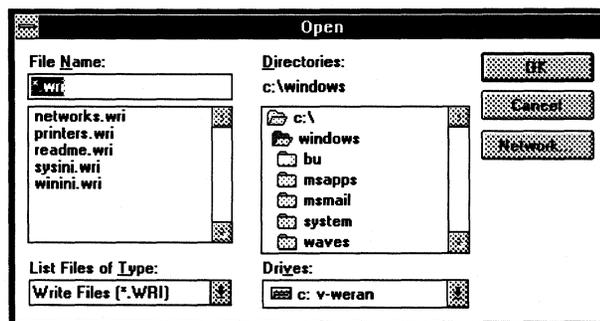
The Common Dialogs are not an application or an accessory, but rather is a dynamic-link library (DLL) consisting of functions that enables developers of Windows-based applications to provide the same look and feel for the handling of interaction with the user for common user interface operations. Common dialog boxes make it easier for an application developer to develop applications for the Microsoft Windows operating system. A common dialog box is a dialog box that an application displays by calling a single function rather than by creating a dialog box procedure and a resource file containing a dialog box template.

The dynamic-link library COMMDLG.DLL provides a default procedure and template for each type of common dialog box. Each default dialog box procedure processes messages and notifications for a common dialog box and its controls. A default dialog box template defines the appearance of a common dialog box and its controls. In addition to simplifying the development of Windows-based applications, a common dialog box assists users by providing a standard set of controls for performing certain operations. As Windows developers begin using the common dialog boxes in their applications, users will find that after they master a common dialog box in one application, they can easily perform the same operations in other applications.

The File Open, File Save As, and Printer Setup common dialog boxes have been extended in the COMMDLG.DLL library provided with Windows for Workgroups to offer additional support for accessing shared resources in a networked environment. This additional support includes displaying a Network button to allow the user to connect to a shared directory or shared printer on another workstation from within the dialog box (instead of requiring the user to switch to File Manager or Print Manager to connect to a remote resource), and providing support for performing universal naming convention (UNC) directory searches, where a user can type a UNC pathname to gain access to the contents of a directory shared on another workstation. Figure 10.1 shows a sample File Open dialog box supported by the Common Dialogs library that has a Network button, as displayed by Write provided with Windows for Workgroups.

Figure 10.1

File Open Common Dialog Box as displayed by Write showing the Network button



Network Button

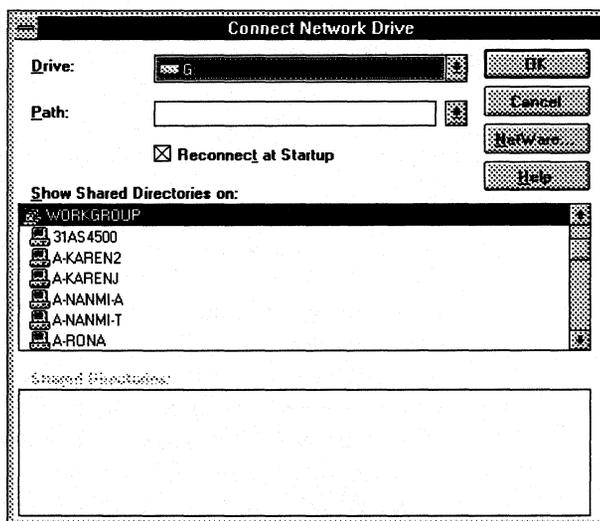
When chosen, the Network button displays the Connect Network Drive dialog box enabling the user to connect to a file share that resides on another workstation. Windows for Workgroups workstation can connect to a shared directory or printer on another Windows for Workgroups workstation or on a server that can interoperate with a Windows for Workgroups workstation directly from within the dialog box. In addition, you can specify the name of a shared resource in the Path field of the connect dialog box (as shown in Figure 10.2) located on a Windows for Workgroups workstation or a secondary network (for example, Novell NetWare) using the naming conventions of the network operating system.

The network drive connection that is made in this dialog box will be active until the user disconnects the network drive by using the Disconnect Network Drive option from the Disk menu in File Manager.

If the user is running Novell NetWare as a secondary network for use with Windows for Workgroups, the Connect Network Drive dialog box that is displayed when the user chooses the Network button may contain a NetWare button, as shown in Figure 10.2.

Figure 10.2

Connect Network Drive dialog box showing the NetWare button



NetWare Button

The NetWare button will be displayed only if Novell NetWare is configured as a secondary compatible network on the Windows for Workgroups workstation. If the NetWare components are installed on the workstation, the NetWare button gives the user access to resources on NetWare file servers in a manner similar to that offered by the Network button for access to SMB-based file

servers (see Chapter 8, “Network Integration with Microsoft LAN Manager and Novell NetWare,” for more information on SMB-server compatibility).

Although the NetWare button is visible in the sample dialog box shown in Figure 10.2, the NetWare button may not always be visible in the Connect Network Drive dialog box for every application that supports the File Open and File Save As common dialog boxes. When the NetWare button is chosen, multiple dialog boxes may need to be displayed to access the NetWare resource. The tracking of the dialog boxes is maintained in the stack space of the application that is calling the common dialog functions. Because of the overhead required to track these additional dialog boxes, the Connect Network Drive dialog box routine will first check to see if the application that is invoking these routines has enough internal stack space to support the requirements for displaying the dialog boxes before the NetWare button is displayed.

If the Connect Network Drive dialog routine determines that there isn't enough internal stack space available for the application to support the displaying of the NetWare connection dialog box (as displayed when the user chooses the NetWare button), the NetWare button won't be displayed. The Connect Network Drive dialog box routine needs at least 7K of free internal stack space in the application in order to support displaying the NetWare button.

UNC Directory Searches

The universal naming convention (UNC) for gaining access to a shared resource as used by Windows for Workgroups is defined as the following format:

`\\servername\sharename\path\filename`

The *servername* identifies the name of the computer where the information resides, and the *sharename* identifies the name of the shared file resource that the information has been shared under. If you know the name of a specific file you want to open, you can append the path and filename of the file to the end of the UNC name. A UNC name may be used with the File Open, File Save As, and Printer Setup common dialog boxes to access a shared resource without first connecting to that shared resource in File Manager, Print Manager, or by using the Network button.

If the resource you are connecting to is protected with a password, you will not be prompted for the password, nor will the password be looked up in your password-list file, if you are using UNC names. Therefore, if the remote resource is password-protected, you will not be able to access the resource using a UNC name unless you are already connected to that resource (for example, using File Manager or Print Manager).

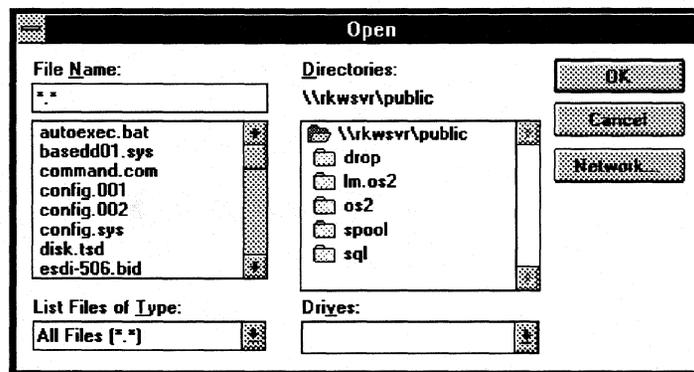
For example, if you want to open a document titled MYDOC.DOC, which is located on a computer named JOHNSMITH in a directory that is shared as DOCUMENT, the user can type “\\JOHNSMITH\DOCUMENT\MYDOC.DOC” when prompted for the name of a file. The File Open and File Save As dialog-box routines will pass the provided filename to the application that displayed the dialog box, and the application can either open the file or save the file depending on the requested action.

Note Some applications assume that file operations are being performed on a logical drive using MS-DOS filename conventions, in which a drive letter and path are used, and may not be able to use a file referenced by a UNC path. If the application you are using doesn't support the use of UNC names, the Connect Network Drive dialog box can be used to associate a MS-DOS drive letter with a given network drive connection.

If the application accepts the UNC name, the network components will establish a temporary connection to the requested computer and perform the requested operations on the file. The defined network connection will remain active until the user closes the given file, and the connection “times out,” or closes, after a period of inactivity as specified by the `keepconn=` entry in the `[network]` section of SYSTEM.INI (see the “SYSTEM.INI: System Initialization File” in Chapter 6 for more information). Figure 10.3 shows a sample UNC name provided for accessing a remote file share. The Directories list box in Figure 10.3 shows the path that the UNC name represents, which in this case is “\\rkwsvr\public”.

Figure 10.3

File Open Common Dialog Box as called from Write, showing the interpretation of a UNC name



ClipBook Viewer



ClipBook
Viewer

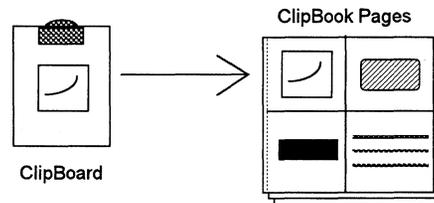
ClipBook Viewer is a new accessory provided with Windows for Workgroups that extends the Clipboard application provided with Windows 3.1 to the network environment. ClipBook Viewer allows you to store multiple pages of information and to share stored information with other workgroup members on the network.

Information that has been copied to the Clipboard by performing a Cut or Copy operation in an application may be pasted into ClipBook Viewer to create a new ClipBook *page*. A ClipBook page represents the contents of a single Copy or Cut operation and is stored in a Clipboard file (with an extension of .CLP). The ClipBook Viewer can hold as many as 127 pages of information, where each page is identified by a name of up to 47 characters and limited in size only by the amount of available disk space.

Figure 10.4 illustrates the relationship between items copied to the Clipboard and items pasted into pages in the ClipBook.

Figure 10.4

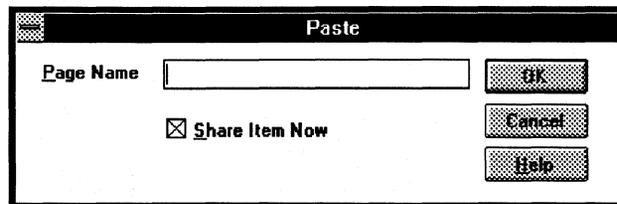
ClipBook Viewer stores multiple pages of Clipboard information.



When information is pasted into ClipBook Viewer, ClipBook Viewer displays the Paste dialog box, as shown in Figure 10.5, which prompts the user for the name to be assigned to the ClipBook page. This dialog box also enables the user to specify whether the information should be shared with other Windows for Workgroups users at the time the ClipBook page is created. Once the user types a name to assign to the ClipBook page and chooses the OK button, a new ClipBook page will be created.

Figure 10.5

Paste dialog box as displayed by ClipBook Viewer when a new ClipBook page is created



Sharing a ClipBook Page

Sharing a ClipBook page enables a user of another Windows for Workgroups workstation to remotely access the information stored in the ClipBook page on

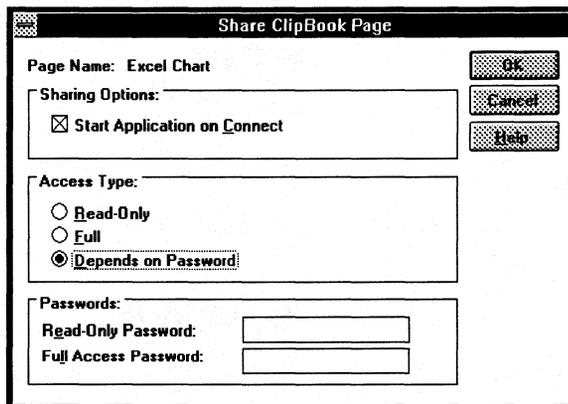
your workstation. Windows for Workgroups uses Network DDE, as discussed in Chapter 11 of this manual, to exchange information between two workstations on the network. ClipBook Viewer serves as the user interface for performing Network DDE operations that involve creating links between applications such as word processors or spreadsheets for information stored in a ClipBook page.

The ClipBook Server application (CLIPSRV.EXE) is an invisible Windows-based application that is loaded by the Windows for Workgroups Winnet driver (WFWNET.DRV) and runs in the background on the workstation. (The ClipBook Server application won't appear in the Windows task list when it is running.) The ClipBook Server is responsible for handling Network DDE requests to ClipBook pages issued by ClipBook Viewer running on other Windows for Workgroups workstations.

When a new ClipBook page is created, if the user has selected the Share Item Now check box, as shown in Figure 10.5, ClipBook Viewer will display the Share ClipBook Page dialog box, as shown in Figure 10.6. The Share ClipBook Page dialog box is used to specify the options related to how your local Windows for Workgroups workstation will behave when a remote user accesses the shared ClipBook page. The Start Application On Connect check box is used to automatically start the application that was used to create the information in the shared page when the shared page is accessed. If this option isn't selected, other Windows for Workgroups users won't be able to establish a link to the information unless the application users used to create the information is already running.

Figure 10.6

The Share ClipBook Page dialog box is used for sharing ClipBook pages with other Windows for Workgroups workstations.



The Access Type option buttons are used to define the permission level for the operations that another Windows for Workgroups workstation can perform on the shared information. A Read-Only access type prevents other Windows for Workgroups users from editing or changing the contents of the shared ClipBook page. The Full access type allows other Windows for Workgroups users to view and edit the information on the shared ClipBook page. Each access type can be accompanied by a password defined by the person sharing

the ClipBook page, granting access of ClipBook page only to users who have the authorized password. The Depends on Password access type lets you choose which users have which access-permission level based on the password they supply when accessing the shared ClipBook page. A password can be up to 14 characters in length and is not case sensitive.

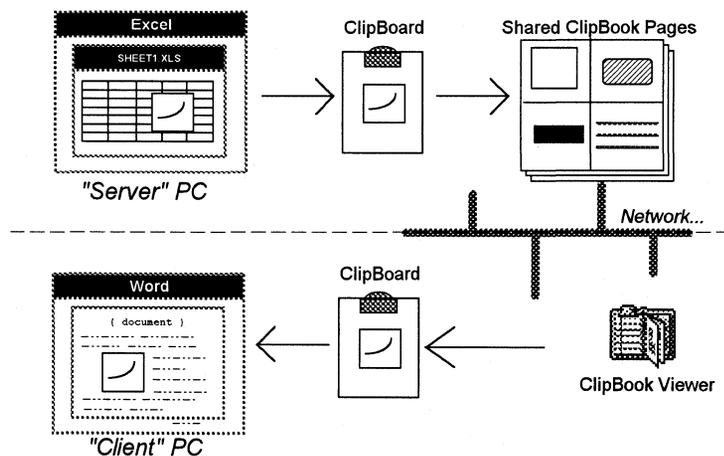
Technically, the Read-Only access type defined in ClipBook Viewer tells the ClipBook Server to respond to only the Initiate, Request, Advise, and Terminate DDE messages. This combination of DDE messages doesn't allow the remote Windows for Workgroups workstation to modify the information in the shared ClipBook page. The Full access type defined in ClipBook Viewer tells the ClipBook Server to respond to all DDE requests, and thus provides full access to information shared in a ClipBook page.

Accessing a Shared ClipBook Page

A user who wants to access the shared ClipBook information uses ClipBook Viewer on the local workstation to connect to the ClipBook stored on the remote workstation. Once the user has connected to the remote ClipBook, the shared pages are displayed, enabling the user to select the desired page of information. A ClipBook page is selected by the client workstation by copying the information to the local Clipboard. If a password has been assigned to that page, the user is prompted for the password to gain access to that page. Figure 10.7 illustrates the flow of information between an application that has shared information through a ClipBook page and an application that has requested the information through ClipBook Viewer.

Figure 10.7

Using ClipBook Viewer to share information between two workstations



Once the user has established a link through ClipBook Viewer to the remote workstation's ClipBook page, the user can copy the information from the remote ClipBook and paste this information into a Windows-based application on the local workstation. If the source and client applications support dynamic

data exchange (DDE) or OLE links, then a Network DDE connection is established between the local and the remote workstation. The selected information may then be dynamically exchanged between the two workstations in a manner similar to using DDE between two applications on the same workstation. See the Chapter 11, "Network Dynamic Data Exchange," for a discussion of the functionality supported by Network DDE.

Net Watcher



Net Watcher is a new accessory provided with Windows for Workgroups 3.1 that enables you to view the users that are connected to your workstation and list the resources they are using. Net Watcher relies on the server functionality supported by a Windows for Workgroups workstation running in 386 enhanced mode. It won't run in standard mode. If your workstation is running in standard mode, or if you disabled sharing (with the Network option in Control Panel), Net Watcher displays a dialog box telling you that it requires the VSERVER.386 virtual device driver.

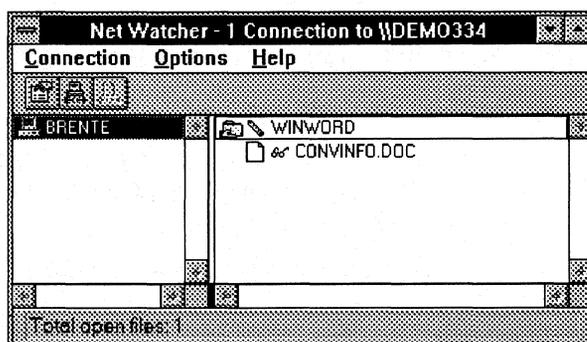
The Net Watcher application window identifies workstations that are connected to your Windows for Workgroups in three different scenarios:

- Workstations accessing a shared printer connected to your workstation
- Workstations accessing a shared directory on the disk drive of your workstation
- Workstations that have established (or have requested) a Network DDE conversation with your workstation

Figure 10.8 shows a sample Net Watcher application window illustrating a workstation with the computer name "BRENTTE" connected to the local Windows for Workgroups workstation with the computer name "Demo334". In this example, the "BRENTTE" computer is connected to a shared directory with the name "WINWORD", and has opened the file "CONVINFO.DOC".

Figure 10.8

Net Watcher application window showing one computer connected to the local Windows for Workgroups workstation

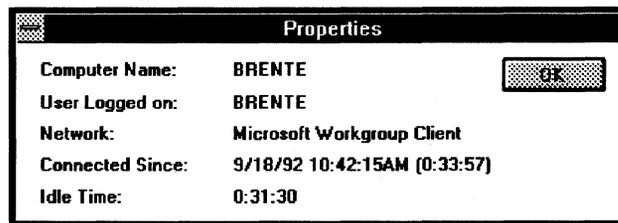


The server process running on the Windows for Workgroups workstation in 386 enhanced mode (VSERVER.386) keeps track of the connection information for a given remote workstation. By double-clicking on the computer name identified in the left window pane, the properties for the given computer connection can be displayed if the remote workstation is accessing a shared printer or a shared directory. If the remote workstation is connected to your workstation but is only engaged in a Network DDE conversation (that is, the remote workstation is not accessing any shared resources on your computer), no properties about the connected workstation are available because VSERVER.386 is not involved with the computer connection (the connection is handled solely by Network DDE).

Figure 10.9 shows a sample Properties dialog box for a remote workstation connected to a shared resource on the local computer. The Computer Name field identifies the name of the remote workstation. The User Logged On field identifies the name of the user that has logged on to the given computer and is connected to a resource shared on your workstation. The Network field identifies the type of network client software that the remote workstation is running—for example, this could be a Microsoft Workgroup Client, or a Microsoft LAN Manager Client, and so on.

Figure 10.9

Sample Properties dialog box showing information about a given computer connected to your workstation



The Connected Since field identifies the time when the remote workstation first connected to the shared resource on your workstation. The elapsed time shown in parentheses identifies the amount of time that has passed since the remote workstation first made this connection. The Idle Time field identifies the amount of time that has passed since the remote workstation accessed the shared resource. This field can be used to see whether a remote workstation is actively accessing a shared resource.

Disconnecting Connected Workstations

The Disconnect option on the Connection menu of Net Watcher can be used to terminate the connection between your computer and a remote workstation. There are three scenarios in which a remote workstation may be disconnected from your computer

- When a remote workstation is connected to a shared printer
- When a remote workstation is connected to a shared directory

- When a remote workstation is engaged in a Network DDE conversation

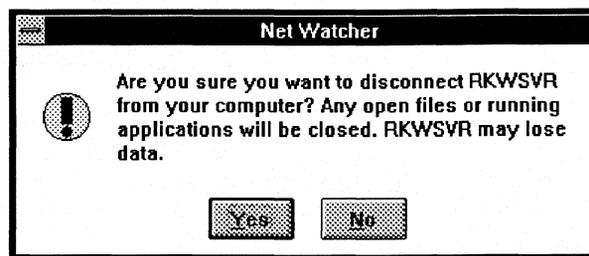
If the remote workstation is accessing a shared resource on your computer, disconnecting that workstation from your computer means that any resources currently in use by the remote workstation will be closed and the connection between the workstations will be temporarily terminated until the remote workstation tries to access the shared resources in the future. Disconnecting a workstation from your computer won't prohibit that remote workstation from accessing your shared resources in the future; only changing the password for a shared resource will allow you to control who has access to a given resource.

For example, suppose the remote workstation is accessing files in a shared directory on your computer when it becomes disconnected. This action closes the files that the remote workstation currently has open, and removes the remote workstation from the Net Watcher application window. Later, if the remote workstation accesses the resource that was in use before its disconnection, a connection between the remote workstation and your computer is re-established. This is also the case for a remote workstation accessing a shared printer—the remote workstation will be re-connected to your shared printer when the remote computer user begins a new print job to the redirected printer port.

Before Net Watcher allows you to disconnect a remote workstation from your computer, a dialog box informs you that if the remote user has made any changes to files that are open on your workstation, the remote user may lose any changes made to the files when you disconnect the remote workstation. This is illustrated in Figure 10.10. Be sure to check with the remote workstation user before disconnecting the remote computer if the remote user has any open files on your computer.

Figure 10.10

Net Watcher Disconnect Warning Dialog Box shows a warning about potential loss of data when disconnecting workstations.



If the remote workstation is engaged in a Network DDE conversation, disconnecting the remote workstation will end the Network DDE conversation. For example, if Network DDE is being used for an interactive session using Chat running on two workstations, disconnecting the remote workstation from the Net Watcher application window will terminate the Network DDE conversation, and Chat will report that the remote computer has hung up.

WinMeter



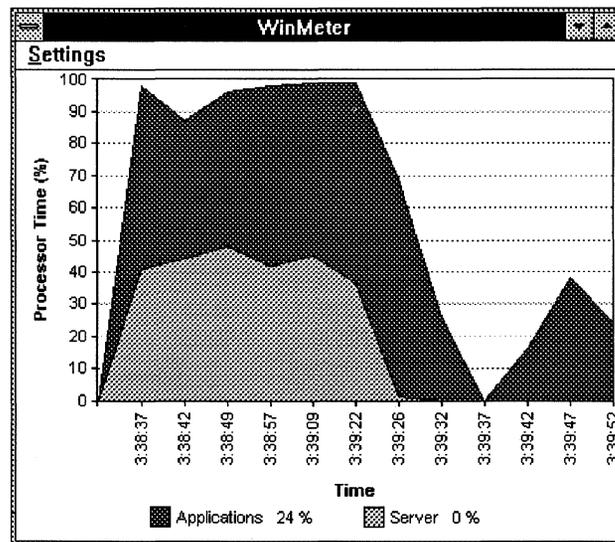
WinMeter

WinMeter is a new accessory provided with Windows for Workgroups 3.1 that allows you to track how much use of computer processor time the server (implemented as VSERVER.386) is using on your local workstation. WinMeter relies on the server functionality supported by a Windows for Workgroups workstation running in 386 enhanced mode, and thus will not run in standard mode—WinMeter will display a dialog box telling the user that it requires the VSERVER.386 virtual device driver present to run if the workstation is not running in 386 enhanced mode, or if the user has disabled sharing in the Control Panel Networks option.

WinMeter graphically displays the ratio between the amount of processing time the server is using, the amount of time the workstation is idle (that is, the amount of time the system is waiting for input or interaction from the user), and the amount of processing time that is being used for applications running on the workstation. The purpose of WinMeter is to give the user an idea of how much processing time the server is using to share resources. Figure 10.11 shows a sample WinMeter display.

Figure 10.11

In this sample WinMeter display, the sampling interval set to every 5 seconds.



The amount of time the workstation spends supporting the server functionality is calculated internally in the server process. The amount of time WinMeter calculates and displays as idle time and application usage time is based on some assumptions about the Windows environment and is designed to give a rough approximation of how the workstation is being used. Although application usage and idle time may not be calculated entirely accurately (depending on a number of factors such as the other tasks running), the server process will accurately calculate the amount of processor time that is being allocated to it and the server utilization display will properly reflect this.

When a print job is being printed and managed by Print Manager for a shared printer on the server, the processor time allocated to this task is attributed to the number reflecting usage by applications rather than by the server process.

MS-DOS-Based Applications

Because MS-DOS-based applications use a variety of means to create an “idle” loop in which the application is waiting for a response from the user, the WinMeter display may show a high amount of processor time being used for applications running in the system when a DOS virtual machine is running.

Although an MS-DOS-based application may cause WinMeter to show high application utilization, the ratio of processing time shown for use by the server will be accurate.

Windows-Based Applications

Some Windows-based applications that use the Windows PeekMessage API function may register high utilization (100%) since they consume all the idle processor time.

Advanced Power Management (APM) Computers

WinMeter will not accurately show the processing time used for applications running on a computer on which Advanced Power Management (APM) functionality is enabled. Microsoft Windows operating system version 3.1 includes support for APM. This enables Windows to extend battery life, display power-status information, and work together with the suspend features of battery-powered personal computers.

All major BIOS and PC vendors will likely provide APM support for existing and new notebook models in 1992. To find out if your computer supports APM, contact your computer manufacturer.

On a computer that has APM enabled, WinMeter can't detect when the workstation is idle, and thus the amount of utilization for applications will read 100 percent at all times. Although the application utilization time (and thus the idle time) will not be accurately displayed, the processor time for the server usage will be correctly calculated.

Hearts



Hearts is a new game provided with Windows for Workgroups. Network DDE is used to allow as many as four Windows for Workgroups workstations to participate in a game of Hearts. If fewer than four Windows for Workgroups workstations are participating in the game, the computer can play the remaining hands. For specific rules for the Hearts game, see the on-line help information.

Changing the Names of Hands Played by the Computer

When the computer is playing additional hands in Hearts, some predefined names are assigned to those hands and are displayed in the Hearts application window. To change the predefined names for the hands the computer plays, add or change the following `p#name=` entries in the `[Hearts]` section of the WIN.INI file:

```
[Hearts]
p1name= name
p2name= name
p3name= name
```

The specified names can be up to 14 characters in length.

File Manager



File Manager, as provided with Windows for Workgroups 3.1, is used to share files and directories with a workgroup or to connect to directories shared on another Windows for Workgroups workstation (or compatible file server as discussed in Chapter 8).

Toolbar

File Manager provided with Windows for Workgroups 3.1 features a toolbar that associates buttons with operations defined in the File Manager menus. File Manager extensions, as documented in the *Windows Software Developer's Kit*, can create new icons to add to the toolbar that are associated with tasks supported by the extension. There is no mechanism to enable a user to change the picture associated with an icon on the toolbar. The tasks present on the toolbar can be changed by selecting *Customize Toolbar* on the *Options* menu.

Disabling File Sharing User Interface

By default, File Manager displays the Share As and Stop Sharing options in the Disk menu. The Share As option enables the user to share a file directory with other members of his or her workgroup. The Stop Sharing option enables the user to stop sharing a file directory that was previously shared.

File Manager provided with Windows for Workgroups recognizes a special restriction switch in the WINFILE.INI that will tell File Manager not to display the file sharing options in the Disk menu. This hides the user interface from the user, and the user will not be able to share any file directories with other users (the user will also not be able to stop sharing any file directories if a directory was shared before the state of the special switch was changed).

The `NoShareCommands=` switch in the **[Restrictions]** section of the WINFILE.INI is used to hide the file sharing options. A value of **1** for the `NoShareCommands=` entry will hide the file sharing options, and a value of **0** (the default) tells File Manager to display the file sharing options on the Disk menu.

Previous Network Drive Connections

Note The list of previous network drive connections is actually maintained by the Windows for Workgroups Winnet driver. File Manager serves as the user interface for connecting and disconnecting network drives.

The Windows for Workgroups File Manager maintains a list of the *most recently used* (MRU) network drive connections. This list provides easy access to the network drives that a user most commonly connects to. File Manager maintains a list of up to 12 of the most recently used network drive connections. The network drive connection information is stored in the **[MRU_Files]** section of the WIN.INI file.

From File Manager, you can review the network drive connections list by choosing Connect Network Drive from the Disk menu. In the subsequent dialog box, use the Path drop-down combo box to see the list of recent network drive connections.

As the phrase “most recently used” implies, this list identifies only the previous 12 most recently used network drive connections. If a connection to a network drive is made that isn't on the MRU list, the least recently used file share connection item is removed from the list and is replaced with the new connection.

The following is an illustration of how the **[MRU_Files]** section is updated in the WIN.INI file when a new entry is added to a full MRU list. In this example, the **[MRU_Files]** section of the WIN.INI file looks something like this:

```
[MRU_Files]
Order=gdfjhcklaei
a=\\SERVER1\PSSB
b=\\SERVER2\PUBLIC
c=\\SERVER3\PUBLIC
d=\\SERVER4\SCRATCH
e=\\SERVER5\USER
f=\\SERVER6\PUBLIC
g=\\SERVER7\USER
h=\\SERVER8\WRKGRP
i=\\SERVER9\OCSE
j=\\SERVER10\PUBLIC
k=\\SERVER11\TOOLS
l=\\SERVER12\ARCHIVE
```

Each drive connection in the **[MRU_Files]** section is identified by a unique letter from A to L. These letters represent the 12 entries in the MRU list, as shown in the preceding example. The exact sequence of the drive connection entries (where the name of the server and share are identified) is unimportant as long as they are identified with unique letters.

The **order=** keyname is used to maintain the drive connection list in the most recently used order. The connection identified on the left side of the list represents the most recently used network drive connection. In this example, “g” is left-most and is thus the most recently used connection. The connection on the right side of the list (“i” in this case) is the least recently used.

If a new network drive connection is made that isn't listed in a full **[MRU_Files]** section, the least recently used network drive is replaced by the new entry, and the **order=** line is updated. If the **[MRU_Files]** section isn't full, the new network drive connection is added, and the drive connection entry is prepended to the beginning of the **order=** entry line.

For example, if a new network drive connection is made to `\\OCOFFSRV\PUBLIC`, then the least recently used network drive connection (“i”, representing `\\SERVER9\OCSE`) is replaced with the new entry. The `order=` entry line is updated by removing “i” from the end of the list and adding a new “i” to the beginning of the list. The resulting `[MRU_Files]` section of `WIN.INI` would now look like this:

```
[MRU_Files]
Order=igdfjhbcklae
a=\\SERVER1\PSSB
b=\\SERVER2\PUBLIC
c=\\SERVER3\PUBLIC
d=\\SERVER4\SCRATCH
e=\\SERVER5\USER
f=\\SERVER6\PUBLIC
g=\\SERVER7\USER
h=\\SERVER8\WRKGRP
i=\\OCOFFSRV\PUBLIC
j=\\SERVER10\PUBLIC
k=\\SERVER11\TOOLS
l=\\SERVER12\ARCHIVE
```

Print Manager



Print Manager

Print Manager, as provided with Windows for Workgroups 3.1, is used to share printers with other members of your workgroup, or to connect to a printer shared on another Windows for Workgroups workstation (or on a compatible server as discussed in Chapter 8).

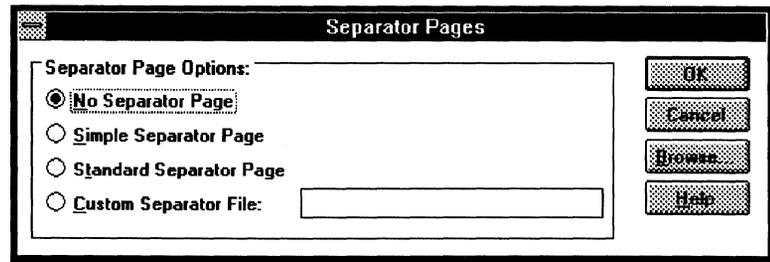
Separator Pages

The Windows for Workgroups Print Manager provides for a separator page to be used to differentiate print jobs that will be printed on a printer connected to the local computer—this printer may or may not be shared with other users of your workgroup.

The use of a separator page may be specified in Print Manager by choosing Separator Pages in the Options menu. This option displays the Separator Pages dialog box as shown in Figure 10.12. The Separator Pages dialog box is used to select one of the Separator Page Options to use when Print Manager prints a document to a shared printer connected to your computer. The selected Separator Page Option will affect output to all shared printers connected to the local workstation. Print Manager doesn't allow individual Separator Page Options to be assigned to each shared printer (the options selected will be used for all printers connected to the local workstation).

Figure 10.12

Separator Pages dialog box from Print Manager



The No Separator Page option button is selected as the default and tells Print Manager not to use a separator page with each document that prints on a locally shared printer.

Note The use of a separator page may be useful if the shared printer is used to print output generated from an MS-DOS-based application that does not generate a form-feed at the end of the print job. In this case, the separator page will ensure that the previous print job has been cleared from the printer before beginning the new print job.

The Simple Separator Page option button tells Print Manager to use a simple format separator page with a typewriter-style font and no graphics. Use this option for dot-matrix printers, daisy-wheel printers, or any printer that doesn't accept downloaded fonts or graphics, or to support faster printing and least printer memory usage.

The Standard Separator Page option button tells Print Manager to use a separator page with a large font. This option is designed for use with laser printers.

The Custom Separator File option button tells Print Manager to use a custom file to define a separator page to use with each document that prints on a locally shared printer.

Custom Separator Pages

A Custom Separator File may be used to personalize your separator page by using your own art. The Custom Separator File option accepts Windows metafiles as the format for the specified file—a Clipboard file may also be used to define a custom separator page, if it includes data in the Picture format.

A *metafile* is a mechanism for storing a graphics device interface (GDI) “picture”—a series of GDI functions that are used to draw an image. A metafile consists of a series of records, each representing a GDI function. When the metafile is played back, each stored function is performed using its recorded parameters. In effect, a metafile is a journal of GDI operations, and because all GDI primitives can be recorded, any image that can be drawn can be stored in a

metafile. Because of their device independence and scaling abilities, metafiles are useful for transferring images between applications, and most applications support the Clipboard format associated with metafiles (CF_METAFILEPICT). When treated as a single graphics primitive, a metafile is easy to paste into an application without that application needing to recognize the specific content of the picture.

Many Windows-based applications that manipulate graphics allow the saving of an image to a file in the Windows metafile format. Information stored on the Clipboard may be saved to a file which includes Windows metafile format—the Clipboard can be used as an intermediary storage location for those graphic applications that don't support the Windows metafile as a file format option. From the File menu in ClipBook Viewer, choose Save As to save the contents of the Clipboard to a file. The resulting file can be specified for the Custom Separator File option in the Separator Pages dialog box.

Note The Microsoft Draw OLE application (provided with Microsoft Word for Windows 2.0 and Microsoft PowerPoint 3.0, for example) may also be used to create the metafile image for use as a Custom Separator File. Since the Microsoft Draw application doesn't support saving the resulting image to a file, the image may be pasted into ClipBook Viewer and then saved. An existing Clipboard file can be read into the Clipboard of ClipBook Viewer, which can then be pasted into the Microsoft Draw application for further modification.

Print Manager recognizes three special metafile strings when interpreting the Windows metafile at the time the Custom Separator File is being sent to the printer. These special strings are used to represent the name of the document being printed, the name of the person printing the document, and the date and time the document is being printed—these strings are “%1%”, “%2%”, and “%3%”, respectively (without the quotation marks).

The custom separator file that is sent to the printer is automatically re-scaled (stretched) to fit the paper size on the target printer. This is a consideration that you should keep in mind when creating a custom separator file, as the generated output on the printer may not be the desired appearance and may need to be adjusted.

For example, a program that manipulates graphic primitives (such as the Microsoft Draw OLE application provided with Microsoft Word for Windows) can be used to create a Custom Separator File as illustrated in Figure 10.13. In this example, the metafile image contains four literal metafile strings (“My Custom Separator Page”, “Document Name:”, “Printed By:”, and “Date and Time:”), a graphic bitmap (pasted from the CARS.BMP file provided with Windows for Workgroups), and three metafile strings to represent the name of the document being printed, the name of the person printing the document, and the date and time the document is being printed (“%1%”, “%2%”, “%3%”,

respectively—the special metafile strings are underlined in Figure 10.13 to highlight them).

Figure 10.13

A sample Print Manager
Custom Separator File
metafile

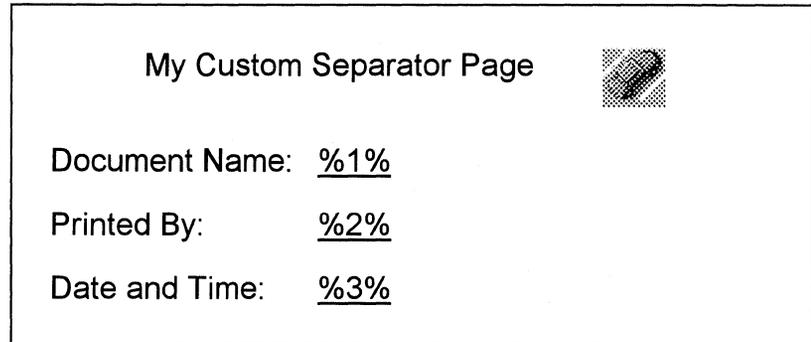
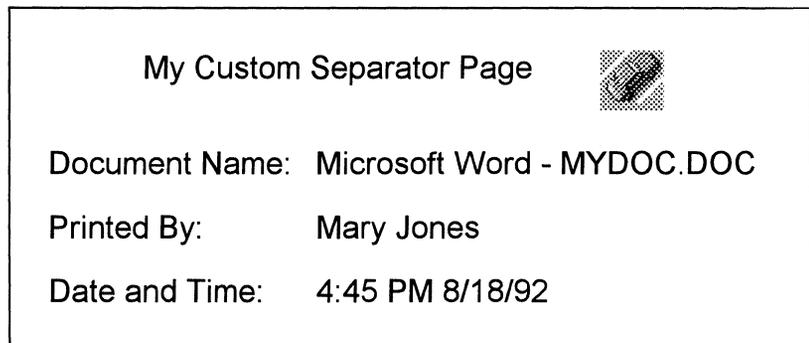


Figure 10.14 illustrates how the Custom Separator Page will look when it is printed preceding a document called “MYDOC.DOC”, printed from Microsoft Word by Mary Jones, at 4:45 P.M. on August 18, 1992.

Figure 10.14

The resulting custom
separator page



Disabling Printer Sharing User Interface

By default, Print Manager displays the Share As and Stop Sharing options in the Printer menu. The Share As option enables the user to share a printer connected to the local workstation with other members of his or her workgroup. The Stop Sharing option enables the user to stop sharing a printer that was previously shared.

Print Manager provided with Windows for Workgroups recognizes a special restriction switch in the WIN.INI that will tell Print Manager to not display the printer sharing options in the Printer menu. This hides the user interface from the user, and the user will not be able to share any locally attached printers with other users. (The user will also not be able to stop sharing any printers if a printer was shared before the state of the special switch was changed.)

The **NoShareCommands=** switch in the **[spooler]** section of WIN.INI is used to hide the printer sharing user interface. A value of **1** for the **NoShareCommands=** entry will hide the printer sharing options, and a value of **0** (the default) tells Print Manager to display the Printer Sharing options on the Printer menu.

Previous Network Printer Connections

Note The list of previous printer connections is actually maintained by the Windows for Workgroups Winnet driver. Print Manager and Control Panel serve as the user interface for connecting and disconnecting network printers.

The Windows for Workgroups Print Manager maintains a list of the most recently used (MRU) network printer connections. This list provides easy access to the network printers that a user most commonly connects to. Print Manager maintains a list of up to 12 of the most recently used network printer connections. The network printer connection information is stored in the **[MRU_Printers]** section of the WIN.INI file.

The network printer connections list is displayed by selecting Connect Network Printer from the Printer menu. Choose the down arrow on the Path line (that is, the drop-down combo box for specifying the network printer path) to see the list of maintained network printer connections. (A network printer connection can also be made from the Printers option in Control Panel).

This list identifies only the previous 12 most recently used network printer connections. If a connection to a network printer is made that is not a member of the MRU list, the least recently used (LRU) printer share connection item is removed from the list and is replaced with the new connection. See the "Previous Network Drive Connections" discussion in the "File Manager" section earlier in this chapter for more information.

Network Dynamic Data Exchange

This chapter provides an overview of the functionality offered by Network Dynamic Data Exchange (DDE) in Windows for Workgroups. Before explaining Network DDE, a thorough discussion of DDE is presented to familiarize the reader with the concepts and terms used by DDE. Network DDE extends the capabilities of Dynamic Data Exchange from the local workstation to the workgroup environment. Windows for Workgroups includes applications that use Network DDE to communicate between workstations—for example, ClipBook Viewer, Hearts, and Chat.

Related information

- *Windows Resource Kit: Chapter 11, “Compound Documents in Windows 3.1”*
- *Microsoft Windows 3.1 Software Developer's Kit (SDK)*

Contents of This Chapter

About Network DDE	11-3
Overview of Dynamic Data Exchange (DDE)	11-3
Dynamic Data Exchange Management Library (DDEML)	11-3
Exchanging Data with Windows-Based Applications	11-4
Uses for Windows Dynamic Data Exchange	11-5
Dynamic Data Exchange from the User's Point of View	11-5
Dynamic-Data-Exchange Concepts	11-6
Dynamic-Data-Exchange Messages	11-8
Dynamic-Data-Exchange Message Flow	11-9
Relation of DDE to Object Linking and Embedding (OLE)	11-10
Overview of Network DDE	11-11
Network DDE Components	11-12
The Role of ClipBook Viewer	11-13
Network DDE Security	11-14
DDE Shares	11-14
How Network DDE Controls Access to a DDE Share	11-15
Network DDE, DDE Shares, and ClipBook	11-16
Examples of How Network DDE Works	11-18
ClipBook Viewer Example	11-18
Direct Network DDE Example	11-22

Using Network DDE with Application Macro Languages	11-23
Network DDE API.....	11-24
Questions and Answers about Network DDE	11-24

About Network DDE

The Dynamic Data Exchange (DDE) mechanism has been supported since the first release of Windows. DDE provides the ability for Windows-based applications to dynamically exchange information. For example, a spreadsheet application can dynamically exchange information with a word processor to ensure that a word processing document contains up-to-date information. Network DDE extends the DDE mechanism from the local workstation to the workgroup environment.

Before we begin discussing the capabilities of Network DDE, a thorough review of DDE is presented in the following section.

Overview of Dynamic Data Exchange (DDE)

The Microsoft Windows operating system provides several methods for transferring data between applications. One way to transfer data is to use Windows dynamic data exchange (DDE). DDE is a message protocol that developers can use for exchanging data between Windows-based applications. When used in an application, DDE offers the user a more integrated Windows work environment.

Dynamic Data Exchange Management Library (DDEML)

Both Windows version 3.1 and Windows for Workgroups version 3.1 include the Dynamic Data Exchange Management Library (DDEML). The DDEML is a dynamic-link library (DLL) that applications running with Windows can use to share data. The DDEML provides an application programming interface (API) that simplifies the task of adding DDE capability to a Windows-based application. Instead of sending, posting, and processing DDE messages directly, an application uses the functions provided by the DDEML to manage DDE *conversations*—a DDE conversation is the interaction between client and server applications. The DDEML also provides a facility for managing the strings and data that are shared among DDE applications. Instead of using atoms and pointers to shared-memory objects, DDE applications create and exchange string handles, which identify strings, and data handles, which identify global-memory objects.

DDEML provides a service that makes it possible for a server application to register the service names that it supports. The names are broadcast to other applications in the system, which can then use the names to connect to the server. The DDEML also ensures compatibility among DDE applications by forcing them to implement the DDE protocol in a consistent manner.

Existing applications that use the message-based DDE protocol are fully compatible with those that use the DDEML. That is, an application that uses message-based DDE can establish conversations and perform transactions with applications that use the DDEML. Because of the many advantages of the DDEML, new applications should use it rather than DDE messages. For more information about the DDEML, see the *Microsoft Windows Programmer's Reference, Volume 1*.

Exchanging Data with Windows-Based Applications

In general, Windows supports two mechanisms that users can use to exchange data between Windows-based applications:

- Clipboard transfers
- Dynamic data exchange

Clipboard Transfers

With the Windows Clipboard, a user can transfer data between applications in the system. In an application, the user issues a command to copy selected data onto the Clipboard. Then, in another application, the user issues a command to paste the data from the Clipboard into the second application's workspace. In general, the Clipboard is a temporary repository of information that requires the direct involvement of the user to initiate and complete the transfer.

Dynamic Data Exchange

The Windows DDE protocol is a standard for cooperating applications. By using the protocol, applications exchange data and carry out remote commands by means of Windows messages.

Because Windows has a message-based architecture, passing messages is the most appropriate method for automatically transferring information between applications. However, Windows messages contain only two parameters for passing data. As a result, these parameters must refer indirectly to other pieces of data if more than a few words of information are to be passed between applications. The DDE protocol defines exactly how the two parameters are used to pass larger pieces of data by means of *global atoms* and *global shared-memory handles*.

A global atom is a reference to a character string. In the DDE protocol, atoms identify the applications exchanging data, the nature of the data being exchanged, and the data items themselves.

A global shared-memory handle is a handle of a memory object allocated by the Windows **GlobalAlloc** function, using the **GMEM_DDESHARE** option. In the DDE protocol, global shared-memory objects store data items passed between applications, protocol options, and remote command execution strings.

The DDE protocol has specific rules for applications involved in a DDE exchange. These rules apply to allocating and deleting global atoms and shared-memory objects.

Uses for Windows Dynamic Data Exchange

DDE is most appropriate for data exchanges that do not require ongoing user interaction. Usually, an application provides a method for the user to establish the link between the applications exchanging data. Once that link is established, however, the applications exchange data without further user involvement.

DDE can be used to implement a broad range of application features:

- Linking to real-time data, such as stock market updates, scientific instruments, or process control.
- Creating compound documents, such as a word-processing document that includes a chart produced by a graphics program. By using DDE, the chart will change when the source data is changed, while the rest of the document remains the same.
- Performing data queries between applications, such as a spreadsheet querying a database application for accounts past due.

Dynamic Data Exchange from the User's Point of View

The following example illustrates two cooperating Windows-based DDE applications, as seen from the user's point of view.

A Microsoft Excel spreadsheet user wants to track the price of a particular stock on the New York Stock Exchange. The user has a Windows-based application called Quote, which in turn has access to NYSE data. The DDE conversation between Microsoft Excel and Quote takes place as follows:

- The user initiates the conversation by supplying the name of the application (Quote) that will supply the data and the particular topic of interest (NYSE). The resulting DDE conversation is used to request quotes on specific stocks.

- Microsoft Excel broadcasts the application and topic names to all DDE applications currently running in the system. Quote responds, establishing a conversation with Excel about the NYSE topic.
- The user can then create a spreadsheet formula in a cell that requests that the spreadsheet be automatically updated whenever a particular stock quotation changes. For example, the user could request an automatic update whenever a change in the selling price of ZAXX stock occurs by specifying the following Microsoft Excel formula:

```
='Quote!'NYSE!ZAXX
```

- The user can end the automatic updating of the ZAXX stock quotation at any time. Other data links that were established separately (such as for quotations for other stocks) still will remain active under the same NYSE conversation.
- The user can also end the entire conversation between Excel and Quote on the NYSE topic so that no specific data links may be subsequently established on that topic without initiating a new conversation.

Dynamic-Data-Exchange Concepts

Certain concepts and terminology are key to understanding dynamic data exchange. The following sections explain the most important of these.

Client, Server, and Conversation

Two applications participating in dynamic data exchange are engaged in a DDE *conversation*. The application that initiates the conversation is the *client* application; the application responding to the client is the *server* application. An application can be engaged in several conversations at the same time, acting as the client in some and as the server in others.

A DDE conversation takes place between two windows, one for each of the participating applications. The window may be the main window of the application; a window associated with a specific document, as in a multiple document interface (MDI) application; or a hidden (invisible) window whose only purpose is to process DDE messages.

Because a DDE conversation is identified by the pair of handles of the windows engaged in the conversation, no window should be engaged in more than one conversation with another window. Either the client application or the server application must provide a different window for each of its conversations with a particular server or client application.

An application can ensure that a pair of client and server windows is never involved in more than one conversation by creating a hidden window for each conversation. The sole purpose of this window is to process DDE messages.

Application, Topic, and Item Names

DDE identifies the units of data passed between the client and server by using a three-level hierarchy of *application*, *topic*, and *item* names.

Each DDE conversation is uniquely defined by the application name and topic. At the beginning of a DDE conversation, the client and server determine the application name and topic. The application name is usually the name of the server application. For example, in a conversation in which Microsoft Excel acts as the server, the application name is Excel.

The DDE topic is a general classification of data within which multiple data items may be discussed (exchanged) during the conversation. For applications that operate on file-based documents, the topic is usually a filename. For other applications, the topic is an application-specific name.

Because the client and server window handles together identify a DDE conversation, the application name and topic that define a conversation cannot be changed during the course of the conversation.

A DDE data item is information related to the conversation topic that is exchanged between the applications. Values for the data item can be passed from the server to the client, or from the client to the server. Data can be passed with any of the standard clipboard formats or with a registered clipboard format. A special, registered format named Link is used to identify an item in a DDE conversation.

Figure 11.1 illustrates a sample link between a Microsoft Excel worksheet and a Microsoft Word document using DDE. In this example, the application name is *Excel*, the topic name is the name of the worksheet, *sheet1.xls*, and the data item that the Word document is requesting is the range of cells *r3c2:r4c4*.

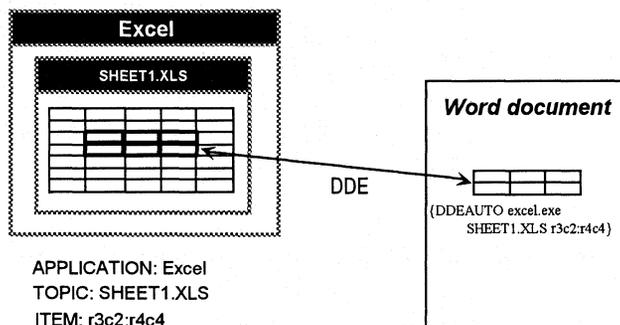


Figure 11.1

Sample DDE link between an Excel worksheet and a Word document

Permanent Data Links

Once a DDE conversation has begun, the client can establish one or more permanent *data links* with the server. A data link is a communication mechanism by which the server notifies the client whenever the value of a given data item changes. The data link is permanent in the sense that this notification process continues until the data link or the DDE conversation itself is ended.

There are two kinds of permanent DDE data links: *warm* and *hot*. In a warm data link, the server notifies the client that the value of the data item has changed, but the server does not send the data value to the client until the client requests it. In a hot data link, the server immediately sends the changed data value to the client.

Applications that support hot or warm data links typically provide a Copy or Paste Link command in their Edit menu to permit the user to establish links between applications. These applications typically support a registered clipboard format named *Link*. When associated with the application's Copy and Paste Link commands, this clipboard format enables the user to establish DDE conversations between applications simply by copying a data item in the server application and pasting it into the client application.

A server application supports the Link clipboard format by placing in the Clipboard a string containing the application, topic, and item names when the user chooses the Copy command from the Edit menu. Following is the standard Link format:

```
application\Otopic\Oitem\O\O
```

A single null character separates the names, and two null characters end the entire string. Both the client and server applications must register the Link clipboard format in order to support the establishment of a link.

A client application supports the Link clipboard format by means of the Paste Link command on its Edit menu. When the user chooses this command, the client application parses the application, topic, and item names from the Link-format Clipboard data. Using these names, the client application initiates a conversation for the application and topic, if such a conversation does not already exist. The client application then sends a WM_DDE_ADVISE message to the server application, specifying the item name contained in the Link-format Clipboard data.

Dynamic-Data-Exchange Messages

Because DDE is a message-based protocol, it employs no special Windows functions or libraries. All DDE transactions are conducted by passing certain

defined DDE messages between the client and server windows. There are nine DDE messages.

The nine DDE messages are summarized as follows. For a detailed description of each DDE message, see the *Microsoft Windows Programmer's Reference, Volume 3*.

Message	Description
WM_DDE_ACK	Acknowledges receiving or not receiving a message.
WM_DDE_ADVISE	Requests the server application to supply an update or notification for a data item whenever it changes. This establishes a permanent data link.
WM_DDE_DATA	Sends a data-item value to the client application.
WM_DDE_EXECUTE	Sends a string to the server application, which is expected to process it as a series of commands.
WM_DDE_INITIATE	Initiates a conversation between the client and server applications.
WM_DDE_POKE	Sends a data-item value to the server application.
WM_DDE_REQUEST	Requests the server application to provide the value of a data item.
WM_DDE_TERMINATE	Ends a conversation.
WM_DDE_UNADVISE	Ends a permanent data link.

An application calls the Windows SendMessage function to issue the WM_DDE_INITIATE message or a WM_DDE_ACK message sent in response to WM_DDE_INITIATE. All other messages are sent by the Windows PostMessage function. The window handle of the receiving window appears as the first parameter of these calls. The second parameter contains the message to be sent, the third parameter identifies the sending window, and the fourth parameter contains the message-specific arguments.

Dynamic-Data-Exchange Message Flow

A typical DDE conversation consists of the following events:

1. The client application initiates the conversation and the server application responds.

2. The applications exchange data by any or all of the following methods:
 - The server application sends data to the client at the client's request.
 - The client application sends unsolicited data to the server application.
 - The client application requests the server application to send data whenever the data changes (hot data link).
 - The client application requests the server application to notify the client whenever a data item changes (warm data link).
 - The server application carries out a command at the client's request.
3. Either the client or server application ends the conversation.

An application window that processes requests from a client or server must process them strictly in the order in which they are received.

A client can establish conversations with more than one server; a server can have conversations with more than one client. When handling messages from more than one source, a client or server must process the messages of a given conversation synchronously, but need not process all messages synchronously. In other words, a client or server can shift from one conversation to another as needed.

Relation of DDE to Object Linking and Embedding (OLE)

Note Chapter 11 of the *Windows Resource Kit* presents an overview of how Object Linking and Embedding (OLE) works when a Windows-based application is designed to take advantage of the OLE features of Microsoft Windows 3.1.

At the Windows operating system level, both object linking and embedding (OLE) and dynamic data exchange (DDE) are protocols for exchanging data and sending commands between applications. For Windows and Windows for Workgroups, neither the OLE libraries nor the Dynamic Data Exchange Management Library (DDEML) depend on special interprocess communications (IPC) mechanisms provided by the operating system. Rather, both OLE and DDE depend on the Windows messaging facility.

Object Linking and Embedding (OLE) is made up of two components: object linking and object embedding. Object linking is analogous to using DDE. It allows you to paste information as a link (for example, data from a spreadsheet into a word processing document) and have the destination document dynamically updated to reflect changes made to the information in the source document.

Object embedding provides a mechanism in which information from one application can be embedded (or directly inserted) in the destination document created by another application. Object embedding is different from object linking in two key ways. First, the information embedded in the destination document consists of not only a representation of the data (for example, a drawing or a chart), but also consists of all of the data that was used to create the object (for example, spreadsheet values). Second, the embedded object is not updated dynamically but can be updated at the request of the user by double-clicking on the object. Double-clicking will activate the object and launch the application that created the object to allow the user to update the data.

At the Windows operating system level, the OLE protocol is built on top of the DDE protocol. More specifically, the OLE protocol is largely a set of DDE Execute command strings that follow a conventional syntax. At the application level, a function-based application program interface (API) is available for both OLE and DDE. For OLE, this API is provided to clients by the OLECLI.DLL file and to servers by the OLESVR.DLL file. For DDE, this API is provided by the DDEML.DLL file.

Overview of Network DDE

Network DDE takes all of the capabilities of DDE as explained earlier in this chapter and extends this powerful capability across the network to allow applications on two or more workstations to dynamically share information. Network DDE is not a special form of DDE; rather it is a redirector that runs in the background on a Windows for Workgroups workstation searching for particular information contained in a DDE conversation. The advantage to the implementation of Network DDE in Windows for Workgroups is that it requires no changes to any DDE-aware applications—all DDE-aware applications can take advantage of Network DDE.

In order for an application on a Windows for Workgroups workstation to use Network DDE to communicate with an application running on another workstation, a DDE share must first be created. A DDE share is analogous to a file share or a printer share as defined in Windows for Workgroups and is used to specify the security permission-levels that grant remote users access to DDE information on the local workstation. The DDE share is used to prevent applications that support DDE from gaining unauthorized access to the local workstation.

A DDE share may be established by ClipBook Viewer for sharing a ClipBook page (see “ClipBook Viewer” in Chapter 10, “New and Updated Accessories”), or may be a defined share name that is used to establish a Network DDE conversation between two DDE-aware applications running on different

workstations. The “Network DDE Security” section later in this chapter describes the security mechanisms used by Network DDE.

Network DDE Components

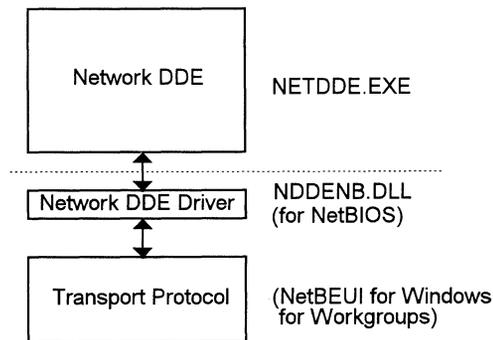
The Network DDE components that are provided with Windows for Workgroups include the Network DDE application, the Network DDE driver, the Network DDE API library, ClipBook Viewer, and ClipBook Server.

Network DDE is implemented as a Windows-based application (NETDDE.EXE) that runs in the background of the Windows for Workgroups workstation, and is loaded by the Windows for Workgroups network driver (WFUNET.DRV) at startup. Because Network DDE is an application that is not visible to the user, Network DDE doesn't appear in the Windows Task List while it is running.

NETDDE.EXE uses the Network DDE driver (NDDENB.DLL) to use NetBIOS to communicate with a Windows for Workgroups workstation over a given transport protocol (for example, NetBEUI) in order to establish a Network DDE conversation. This is illustrated in Figure 11.2.

Figure 11.2

Relationship of Network DDE to the Network DDE driver and transport protocol



The Network DDE API library (NDDEAPI.DLL) is used to provide a programming interface for use by application developers to support the creation and manipulation of DDE shares. The Network DDE API library can be used to create DDE shares enabling applications that support DDE to communicate directly with one another using Network DDE, without having to go through ClipBook Viewer.

ClipBook Viewer (CLIPBRD.EXE) serves as the user interface for accessing information stored in ClipBook pages and sharing stored information with other members of your workgroup. (ClipBook Viewer is discussed in detail in the “ClipBook Viewer” section of Chapter 10, “New and Updated Accessories.”) ClipBook Server (CLIPSRV.EXE) is a Windows-based application that runs in the background on the Windows for Workgroups workstation like

NETDDE.EXE and doesn't appear in the Windows Task List while it is running. ClipBook Server manages local ClipBook data and provides it to ClipBook Viewer running on local and/or remote workstations. CLIPSRV.EXE is loaded by the Windows for Workgroups network driver (WFWNET.DRV) at startup.

The Role of ClipBook Viewer

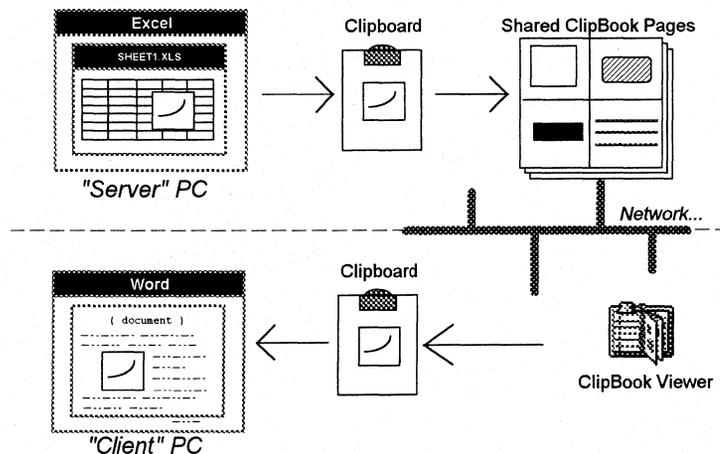
For Network DDE, ClipBook Viewer (CLIPBRD.EXE) application serves as the user interface for establishing Network DDE conversations between Windows-based applications on different workstations that support DDE or OLE links.

Information in a shared ClipBook page can be accessed from the local workstation by using ClipBook Viewer. ClipBook Viewer is used to establish the connection to the remote workstation. Once the user has established a link through ClipBook Viewer to the remote workstation's ClipBook, the user can copy the information from the remote ClipBook and paste this information into a Windows-based application on the local workstation. If the source and client applications support DDE or OLE links, then Network DDE is used to establish a dynamic connection between the local and the remote workstation. Once a Network DDE connection is established, the selected information may be dynamically exchanged between the two workstations in a manner similar to using DDE between two applications residing on the same workstation.

Figure 11.3 illustrates how information pasted into a ClipBook page is shared by a server application (in this case, Excel), and then how the information is accessed by a client application (in this case, Word) by copying information from the remote ClipBook page by using ClipBook Viewer.

Figure 11.3

Using ClipBook Viewer to share information between two Windows for Workgroups workstations



More information about the use of ClipBook Viewer in establishing Network DDE conversations is discussed later in this chapter.

Network DDE Security

Because Network DDE enables for direct communication with other workstations, certain security mechanisms have been put in place to prevent unauthorized access to DDE information on a remote Windows for Workgroups workstation.

DDE Shares

Just as Windows for Workgroups provides the ability to control security for users access to file shares and printer shares, Network DDE uses the concept of a DDE share to control access to DDE information. A DDE share may be created in one of two ways: by using ClipBook Viewer share a ClipBook page with other workstations, or by an application that calls the Network DDE API functions present in NDDEAPI.DLL.

DDE shares are defined in the DDE shares database. The DDE shares database defines the name of the DDE shares that can participate in a Network DDE conversation. An entry in the DDE shares database identifies the name used in the DDE initiate message and the DDE topic that the Network DDE conversation can be established with. The DDE shares database also identifies the security permission levels for a given DDE share for controlling access to that share.

A DDE share can be created for each DDE topic supported by a DDE-aware application that you want to communicate with, or a DDE share that specifies "*" as the topic name can be used to enable access through Network DDE to all topics supported by the given application name. To provide compatibility with future versions of Windows, the **DDEShareAdd** API function present in NDDEAPI.DLL should be used to add new DDE shares to the DDE shares database. Developer information for the Network DDE API functions will be incorporated into a future edition of the Windows 3.1 SDK.

Important Currently the DDE shares database is located in the **[DDEShares]** section of the SYSTEM.INI file. However, the location and format used by the DDE shares database may change in a future version of Windows. The information present in this section should not be edited by hand or directly by an application—the API calls present in the Network DDE API library should be used to manipulate the DDE shares database.

How Network DDE Controls Access to a DDE Share

All Network DDE communication between an application on the local workstation and an application on a remote Windows for Workgroups workstation is routed through the Network DDE application. Network DDE uses the reserved DDE application name “NDDE\$” to establish a connection to a DDE share on a remote Windows for Workgroups workstation. You will recall from our discussion of DDE earlier in this chapter that a DDE conversation is established for a particular application and topic. Once a DDE conversation has been established, an application can request information about a given data item.

In order to establish a Network DDE conversation, a connection must first be made between an application on the local workstation and an application identified by a DDE share defined on a remote workstation. A DDE conversation is established by passing the name of the application and the desired topic that the local application wants to communicate with. An example of the syntax used to connect to a DDE share is defined as the following:

application: \\ComputerName\NDDE\$ topic: DDEShare name

In this example, *ComputerName* represents the defined name of the remote Windows for Workgroups workstation and *DDEShare name* represents the name of the defined DDE share. Once a Network DDE conversation has been established between the remote DDE application and the specified DDE topic, an application on the local workstation can request information about a given data item.

When an application on a local workstation requests a Network DDE conversation to be established with a remote workstation, the following operations are performed:

- DDE is used by the application on the local workstation to establish a conversation with NETDDE.EXE running on the local workstation.
- NETDDE.EXE on the local workstation establishes a connection to NETDDE.EXE on the specified remote workstation using NetBIOS.
- NETDDE.EXE on the remote workstation checks the DDE shares database residing on the workstation to see if the specified DDE share is valid.
- If the DDE share is defined and the proper password (if requested) is supplied, then a Network DDE conversation between the local and remote workstation is established. If the DDE share does not exist, or the requested password is not supplied, then Network DDE will not establish a conversation between the local and remote workstations.

Because of the security mechanisms built into Network DDE, all Network DDE conversations are first passed through the reserved “NDDE\$” application name

to validate the DDE conversation with a defined DDE share. An application on the local workstation that wants to support communication with other applications running on remote Windows for Workgroups workstations can create a DDE share by calling the appropriate DDE share manipulation functions present in the Network DDE API dynamic-link library (NDDEAPI.DLL).

Network DDE, DDE Shares, and ClipBook

In the context of using ClipBook Viewer, a DDE share is used for access to information stored in a ClipBook page—ClipBook page may reside in the local ClipBook or in a ClipBook on a remote workstation. Information stored in a ClipBook page supports at least one of three properties: static, link, and object link. A DDE share is created for each given property of ClipBook page supported by the information.

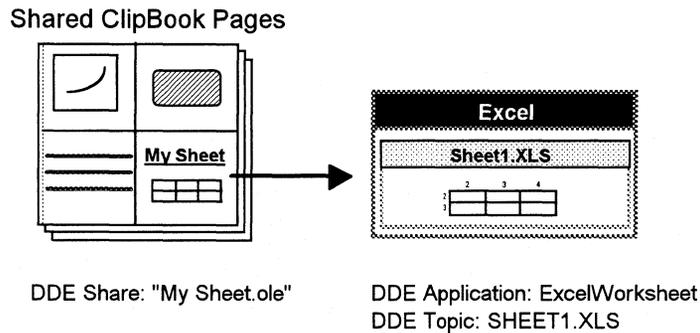
Static information in a ClipBook page is used to render a view of the data as it was defined at the time the information was pasted into a ClipBook page—this information may be stored in a variety of clipboard formats, including as text, as a Windows bitmap, or as a Windows metafile.

Link information is used by some applications to establish a DDE connection to the source application when the Paste Link operation is specified. ClipBook Viewer creates an additional share with a .dde suffix to permit this connection. The .dde suffix is used to identify that the DDE conversation is to be performed by using the DDE link protocol. Similarly, ClipBook Viewer may also create an additional share with an .ole suffix to permit a DDE connection. The .ole suffix is used to identify that the DDE conversation is to be performed by using the OLE link protocol.

Figure 11.4 illustrates the relationship between a DDE share and a shared ClipBook page reference that identifies an object link to data that resides in a Microsoft Excel worksheet. In this example, copying the information from the remote ClipBook page and pasting the data as a link into an application residing on the local workstation that supports OLE links establishes a Network DDE conversation between the SHEET1.XLS Excel worksheet and the client application.

Figure 11.4

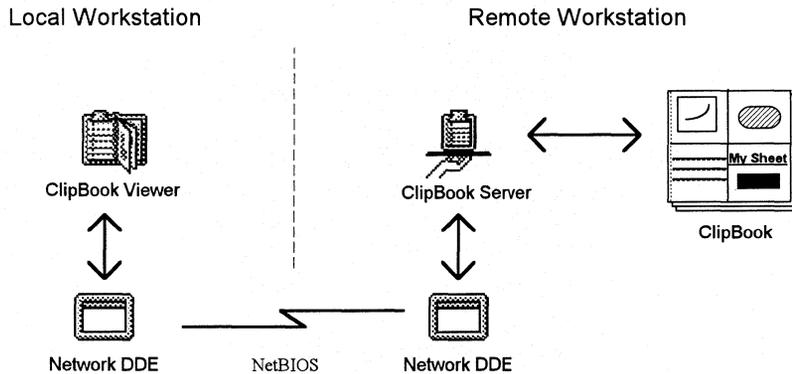
Relationship of DDE share to DDE application and topic as created by ClipBook Viewer for an object link



ClipBook Viewer communicates with ClipBook Server application for access to the desired information stored in ClipBook. ClipBook Viewer is used to request a list of the available pages defined in ClipBook (for a remote workstation, the available pages are only the shared pages) and static data stored in a ClipBook page. If the requested ClipBook information resides on a remote workstation, Network DDE is used to obtain the information. Figure 11.5 illustrates the role of ClipBook Viewer in retrieving information from ClipBook through ClipBook Server.

Figure 11.5

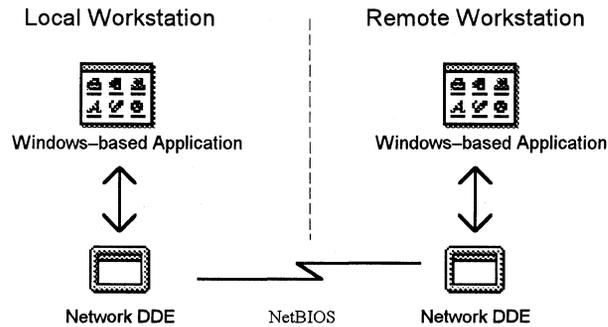
ClipBook Viewer establishes the Network DDE conversation with the remote workstation.



If the information in the remote ClipBook supports the object link property and the client application supports DDE or OLE links, the user on the local workstation can select Paste Special from the Edit menu in the Windows-based application and perform a Paste Link operation. This will establish a dynamic Network DDE link between the client application running on the local workstation and the server application running on the remote workstation, as illustrated in Figure 11.6. The types of operations that the client application can perform on the data shared in the remote ClipBook is dependent on the access type permissions defined by the remote user for the given ClipBook page. (See the "ClipBook Viewer" section in Chapter 10, "New and Updated Accessories," for more information on the supported permission options.)

Figure 11.6

Once a Network DDE conversation is established, Windows-based applications can dynamically exchange data.



Examples of How Network DDE Works

As described earlier in this chapter, Network DDE is a redirector that runs in the background of the Windows for Workgroups workstation. It analyzes DDE Initiate requests to detect the use of a special DDE application name, “NDDE\$”, as reserved by Network DDE, preceded by the name of the remote Windows for Workgroups workstation. The NDDE\$ application name is used to establish a Network DDE session between an application running on a local workstation and a DDE share residing on a remote workstation. NDDE\$ is used to validate the DDE request for the DDE share defined by the remote user.

This section describes the Network DDE process in more detail by examining two sample scenarios. The first scenario involves using ClipBook Viewer to establish a link with data residing in a ClipBook page on a remote Windows for Workgroups workstation. The second scenario involves using a client application on a local Windows for Workgroups workstation to communicate directly with a server application running on a remote Windows for Workgroups workstation.

ClipBook Viewer Example

Let's examine how Network DDE works by looking at an example scenario involving ClipBook Viewer and a shared ClipBook page residing on a remote Windows for Workgroups workstation. In this scenario, a remote Windows for Workgroups workstation defined as “JohnSmith” has shared a ClipBook page named “May Sales Report,” which is a Microsoft Excel worksheet. Given the caveat that the information present in the DDE shares database should not be modified directly by the user, let's assume the following DDE share names are mapped to the following application and topic name entries present in the DDE shares database (more information than that listed in the following example is present in the DDE shares database, but it isn't included here for this example).

DDE Share Name	Application Name	Topic Name
CLPBK\$	clipsrv	system
\$May Sales Report	clipsrv	\$May Sales Report
\$May Sales Report.ole	ExcelWorksheet	MAYSALES.XLS

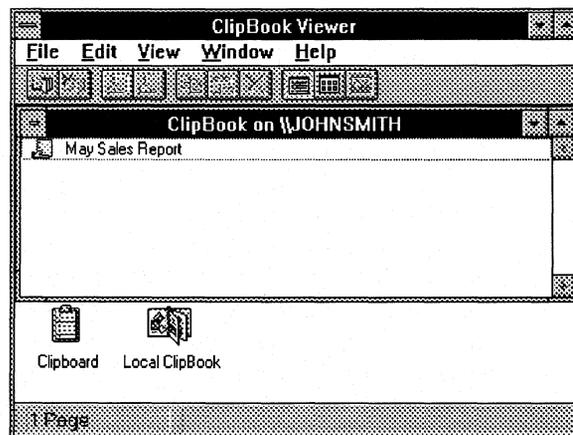
The DDE Share Name column represents the names of the DDE shares defined in the DDE shares database. The Application Name column represents the application names that are used in a DDE conversation. The Topic Name column represents the names of the topics that are supported in a DDE conversation with the given application name.

For this example, the “CLPBK\$” entry identifies the DDE share that is used to communicate with ClipBook Server application. The “\$May Sales Report” DDE share is used to reference the static representation of the “May Sales Report” ClipBook page. The “\$May Sales Report.ole” DDE share is used to establish an OLE link with the MAYSALES.XLS Excel worksheet on which the data resides.

In order to access the information in the shared ClipBook page on the remote workstation, the local user invokes ClipBook Viewer. ClipBook Viewer establishes a Network DDE conversation with ClipBook Server running on the remote workstation and requests information about the shared ClipBook pages defined on the “JohnSmith” workstation (using “\\JOHNSMITHNDDE\$” as the application name and “CLPBK\$” as the topic name). ClipBook Server on the remote workstation checks ClipBook pages defined on the remote workstation and returns the list of shared ClipBook pages to ClipBook Viewer on the local workstation. Figure 11.5 shows a sample ClipBook Viewer screen showing the shared ClipBook pages on the remote “\\JOHNSMITH” workstation.

Figure 11.5

Sample ClipBook Viewer display showing shared ClipBook pages on the remote \\JOHNSMITH workstation



The user on the local workstation selects the desired ClipBook page item from the remote workstation displayed in ClipBook Viewer and chooses Copy on the Edit menu of ClipBook Viewer to gain access to the desired information. If a password is assigned to ClipBook page, the local workstation user is prompted for the authorized password, and must supply the proper password before access to the shared ClipBook page will be given. The Copy operation places information about the shared ClipBook page into the system Clipboard of the local workstation—the information includes a static representation of the data and any link information if supported by ClipBook page item.

Once the information has been copied to the Clipboard, the user can then paste the information into the client application. The information may either be pasted as a static representation of the information or, if supported, the user can perform a Paste Link operation to establish a link from the client application on the local workstation to the application that created the data on the remote workstation. If the user performs a Paste Link operation, the client application attempts to make a connection to the object link information for the DDE share defined on the remote workstation (for this example, the DDE share for the link information is defined as "\$May Sales Report.ole"). For an OLE link, the OLE client library (OLECLI.DLL) performs the DDE initiate request on behalf of the client application. The client application generates a DDE initiate message with an application name defined as "\\JOHNSMITH\NDDE\$" and a topic name defined as "\$May Sales Report.ole" as identified in the link information that was present for the data residing in the Clipboard.

NETDDE.EXE on the local workstation will intercept the DDE request and establish a NetBIOS communication session with NETDDE.EXE on the remote workstation. NETDDE.EXE on the remote workstation will check the DDE shares database to determine if the specified DDE share name is valid. In this case, NETDDE.EXE on the remote workstation determines that the specified DDE share is valid.

NETDDE.EXE on the remote workstation will attempt to establish a DDE conversation with the application name and topic name denoted by the DDE shares database entry—" \$May Sales Report.ole" is the database entry in this case. For the "\$May Sales Report.ole" entry, the DDE application name in the DDE shares database identifies "ExcelWorksheet" as a class name and "MAYSALES.XLS" as the supported topic name.

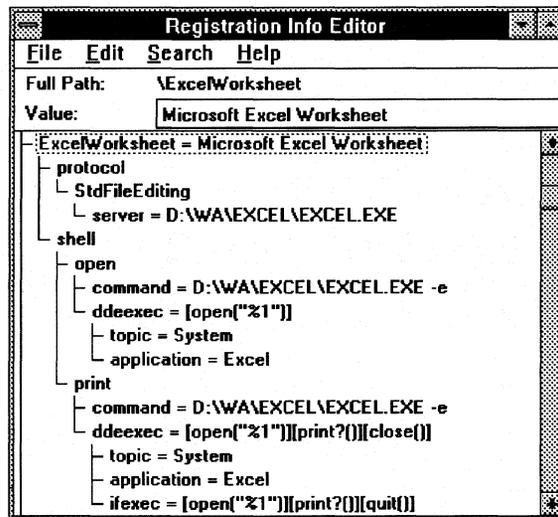
If the DDE initiate message sent by the NETDDE.EXE application on the remote workstation is not acknowledged by the specified server application, the remote workstation may attempt to start the specified server application depending on how the Start Application On Connection option is set on the remote workstation. (See Figure 10.6 for an example of the options present in the Share ClipBook Page dialog box used for sharing ClipBook pages.)

If the requested application isn't running, NETDDE.EXE on the remote workstation will examine the value of the Start Application On Connection option. If permission is given for the server application to be started, NETDDE.EXE will pass the name of the requested server application (in this case, "ExcelWorksheet") to the Windows WinExec function to attempt to start it. If the WinExec function call is successful, NETDDE.EXE on the remote workstation retries the DDE initiate request, but in this case the application name doesn't match the executable name, so WinExec returns an error indicating that the path or filename wasn't found.

When the WinExec function call isn't successful, NETDDE.EXE checks the Windows registration database to see if the requested application name is registered as a defined class name. In this example, a registration database key of the form "ExcelWorksheet\protocol\StdFileEditing" is used to check the registration database. Figure 11.6 shows some sample entries of the registration database for the ExcelWorksheet section—the first section listed under the ExcelWorksheet heading in this figure identifies the path for the server application executable file (in this example it is D:\WA\EXCEL\EXCEL.EXE).

Figure 11.6

Sample display of registration database information for Microsoft ExcelWorksheet section



Note To view the contents of the registration database on your workstation, choose Run from the File menu of Program Manager or File Manager, and type "REGEDIT /V" for the Command Line entry.

If the specified registration database key is opened successfully, NETDDE.EXE tries to get the value of the "server" entry for the "ExcelWorksheet\protocol\StdFileEditing" key. If a value is obtained, NETDDE.EXE passes this value to the WinExec function. If the WinExec function call succeeds, Network DDE repeats the DDE initiate request. In our scenario, this succeeds. If this failed, NETDDE.EXE ends the Network DDE

conversation that was requested between the client application and the remote workstation.

Once a successful DDE conversation is established between the local client application and the server application on remote workstations using Network DDE, data items may be exchanged until either side ends the conversation. For example, when a change occurs to the MAYSALLES.XLS worksheet on the “\\JOHNSMITH” computer, the changes will be dynamically reflected in the client application.

Direct Network DDE Example

In another example for explaining how Network DDE works, let's assume a user wants to enable an application to establish a direct Network DDE connection between a client application running on a local workstation and an Excel worksheet residing on a remote Windows for Workgroups workstation (considered to be the server application). A direct Network DDE conversation would be used to establish a connection between the client application and the server application without going through ClipBook Viewer—in this case, the client application needs to know the specific information available from the server application to request it. An example of a client application could be Microsoft Excel, Microsoft Word for Windows, or Microsoft Visual Basic—all of which support a macro language that enables the use of DDE under programmatic control.

Given the caveat that information present in the DDE shares database should not be modified directly by the user, let's assume the “MY DATA” DDE share name is defined on the remote Windows for Workgroups workstation (“\\JOHNSMITH”), which maps to the following application and topic name entries present in the DDE shares database (more information than that listed in the following example may be present in the DDE shares database, but it is not included here for this example):

DDE Share Name	Application Name	Topic Name
MY DATA	Excel	MYSHEET.XLS

As in the previous example, the DDE Share Name column represents the names of the DDE shares in the DDE shares database, the Application Name column represents the application names that are used in a DDE conversation, and the Topic Name column represents the names of the topics that are used in a DDE conversation with the given application name.

The “MY DATA” entry identifies the DDE share that is used to establish a Network DDE communication link with Excel, specifying MYSHEET.XLS

present on the remote Windows for Workgroups workstation as the desired topic name. In this example, we'll assume that the defined DDE share authorizes full access to the information (that is, the DDE share enables the information it references to be updated remotely).

In order to gain access to the information in the defined DDE share on the remote workstation, Excel on the local workstation needs to first establish a DDE conversation with the DDE share, passing “\JOHNSMITH\NDDE\$” as the application name, and “MY DATA” as the topic name. NETDDE.EXE on the local workstation will intercept the DDE request and establish a NetBIOS communication session with NETDDE.EXE on the remote workstation. NETDDE.EXE on the remote workstation will check the DDE shares database to determine if the specified DDE share name (in this case, “MY DATA”) is valid. In this case, NETDDE.EXE on the remote workstation determines that the specified DDE share is valid.

NETDDE.EXE on the remote workstation will attempt to establish a DDE conversation with the application name and topic name denoted by the DDE shares database entry—“MY DATA” in this example. For the “MY DATA” entry, the DDE application name in the DDE shares database identifies “Excel” as a class name and “MYSHEET.XLS” as the supported topic name.

If the DDE initiate message sent by the NETDDE.EXE application on the remote workstation is not acknowledged by the specified server application, the remote workstation may attempt to start the specified server application as described in the previous example. For this example, we'll assume that Microsoft Excel is presently running. Excel will receive the DDE initiate message and will attempt to open the worksheet as identified by the provided topic name, which in this case is MYSHEET.XLS.

Once the MYSHEET.XLS worksheet is opened successfully by Excel, the client application on the local worksheet will be engaged in a Network DDE conversation with Excel running on the remote workstation. The client application can then request information from the MYSHEET.XLS worksheet, or poke information into the MYSHEET.XLS worksheet as desired for a specified DDE item name. In the case of an Excel worksheet, the DDE item name can be a worksheet cell, a range of cells, or a name defined in the Excel worksheet to represent a cell or a range of cells.

When the client application is finished using the remote MYSHEET.XLS worksheet, the client application can end the Network DDE conversation.

Using Network DDE with Application Macro Languages

Applications such as Microsoft Excel for Windows and Microsoft Word for Windows feature built-in macro languages that can be used to automate tasks

and operations within these applications. Both Excel and Word provide functionality for performing DDE requests from within their respective macro languages. Microsoft Visual Basic for Windows also provides support for implementing DDE in a Visual Basic application.

In order to enable an application on a local workstation to communicate directly with an application on remote workstation, a DDE share must be created to define the supported application name and topic(s) that the local application can access. As described earlier in this chapter, functions present in the Network DDE API library can be used to enable application developers to create DDE shares.

Network DDE API

The Network DDE API supports routines that, for example, create and delete a DDE share are provided in the NDDEAPI.DLL. The documentation for the macro language supported by your application provides information on how you may call a function present in a DLL from within your application.

The developer information for Windows for Workgroups, including information on the Network DDE API library, will be incorporated into a future edition of the Windows 3.1 SDK. Information may be available before this update; check the Microsoft Download Service (MSDL), or the appropriate forum on CompuServe for up-to-date information.

Note The *Windows for Workgroups Resource Kit* provides a utility uses the NDDEAPI.DLL Network DDE support routines for managing DDE shares on the local workstation. You can use this utility to define DDE shares that support a Network DDE conversation.

Questions and Answers about Network DDE

This section answers some common questions about Network DDE.

Q I don't see an application called Network DDE in one of my program groups. How does Network DDE get loaded?

A NETDDE.EXE is loaded automatically by the Windows for Workgroups Winnet driver (WFUNET.DRV) during system initialization.

Q Can I use Network DDE with my current version of Windows 3.1?

A No. Network DDE works with the integrated networking functionality of Windows for Workgroups and will not work without the other networking components found only in Windows for Workgroups.

Q How do I enable an application that I wrote to use Network DDE to get information from an application that supports DDE running on a remote workstation?

A If your application supports DDE or OLE links, all you need to do is to create an entry in the DDE shares database on the remote workstation that specifies the DDE application name and topic that the remote application supports. If the remote application supports multiple topics, the value for the topic for the DDE shares database entry can be set to "*" to enable all supported topics to be accessed. The utility provided with the *Windows for Workgroups Resource Kit* can be used to create a DDE share that can be used with your application.

Windows for Workgroups includes an electronic mail application, Mail, that can be used to exchange information with other Windows for Workgroups workstations. Mail also can work interactively with many other Windows-based applications. This chapter describes the components that make up Mail and explains how to use mail functionality from within other Windows-based applications such as Microsoft Word for Windows and Microsoft Excel for Windows. This chapter also describes four customization features you can use to tailor Mail for your office—custom commands, custom message types, the Messaging Application Program Interface (MAPI), and user information templates.

Related information

- *Microsoft Mail User's Guide (Version 3.1 for the Windows for Workgroups Operating System):* Part 2, "Managing the Workgroup Postoffice."
- *Windows for Workgroups Resource Kit:* Chapter 6, "Windows for Workgroups Initialization Files;" Appendix B, "Microsoft Mail Configuration Guide."

Contents of This Chapter

About Mail	12-3
The Mail Client	12-3
Rich Message Content	12-4
Easy Addressing	12-5
Flexible Message Organization	12-6
Working Online or Offline.....	12-7
The Mail Postoffice.....	12-8
Workgroup Postoffice Administration	12-10
Interface Between the Mail Client and Postoffice	12-11
Customizing the Mail Client.....	12-13
Custom Commands.....	12-13
Custom Message Types.....	12-19
Messaging Application Program Interface (MAPI).....	12-26
Integrating Mail and Other Windows-Based Applications.....	12-27
Differences Between Windows for Workgroups Mail and Mail 3.x	12-28

Extending the Reach of Mail	12-29
Extensions Software	12-29
Connecting Multiple Workgroups	12-30
Connecting to Other Mail and Scheduling Systems with Gateways	12-30
Connecting to Mail from a Remote Location.....	12-31
Upgrading Connectivity.....	12-31
Tips for Using Mail.....	12-35
Recreating the Mail Initialization Procedure.....	12-36
Changing the Name of the Postoffice Administrator	12-36
Packaging Objects with UNC Pointers	12-37
Questions and Answers about Mail.....	12-38

About Mail

The Mail application provided with Windows for Workgroups has a client side, a mail-server side, and an interface between them. The client side includes a visual user interface, made up of viewers for messages, folders, and address lists. The server simply contains a directory structure known as the *postoffice* and has no programmatic components. The interface between client and server manages message storage and retrieval, name validation, and directory access.

Mail uses a “store and forward” design. A user sends mail to, and receives mail from, a *message store* on his or her own computer. As a user sends a message, it is forwarded from the local computer's message store to the *postoffice* located on the mail server. The postoffice has a mailbox for each user, giving users access to the messages they've received when they log on to Mail.

Mail is based on the “shared file system,” meaning the postoffice can reside on any computer in the workgroup running Windows for Workgroups in 386 enhanced mode. As mentioned before, the postoffice requires no extra programmatic components. Instead, all postoffice file manipulation is handled by the Mail client.

The postoffice is a directory structure in which the main directory is called WGPO. The structure of the postoffice is efficient because Mail only stores one copy of each mail message, even when a message is addressed to multiple recipients. For example, if you sent a 75K message to 10 people, Mail stores only 75K of information on the postoffice, not 750K. The postoffice is a temporary message store, holding the message until the recipient's workstation retrieves it. When it is retrieved, the message is removed from the postoffice.

The following sections describe the features of the mail client, the server, and the interface between them.

The Mail Client

Mail is a small but robust tool. Its easy-to-use and versatile user interface includes these features:

- Rich message content supported by the mail editor
- Easy addressing supported by Mail's address books
- Flexible message organization supported by the folder filing system
- The ability to work online or offline

These features are readily available to users so they can customize their own work environment. Later in this chapter, we'll discuss some other advanced techniques that you might use as the workgroup administrator.

This section describes each feature in detail and shows how you can use these features to make your job as workgroup administrator easier.

Rich Message Content

Mail includes a text editor that lets you create messages using a variety of typefaces. It also features automatic word wrapping and tab support. The best part of Mail messages is that you can include the contents of other files—formatting and all—in your message. You can exploit any or all of these options when you use custom message types to create special message forms for your workgroup. (See the “Custom Message Types” section later in this chapter for more information.)

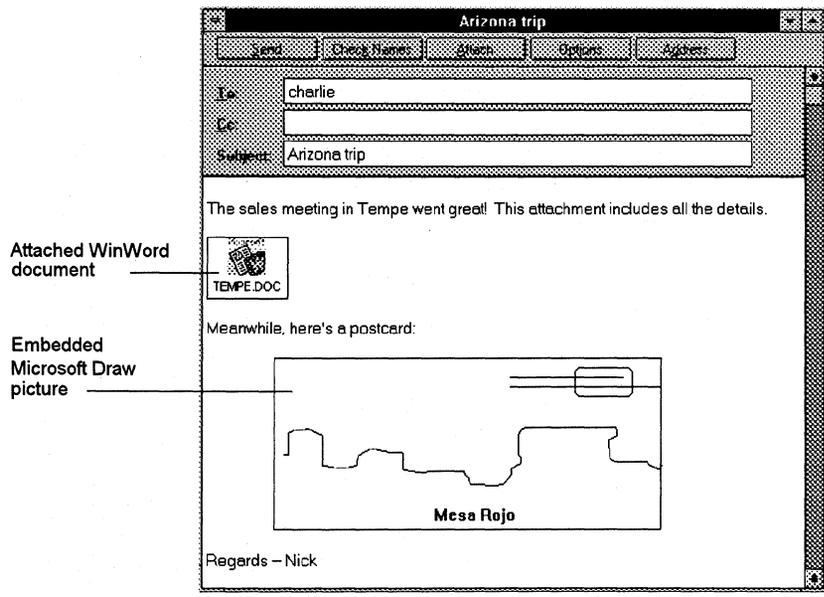
You can use all of these techniques to create a message:

- Type text directly. Mail includes a text editor with automatic word wrapping, a proportional font, and tab support.
- Embed objects, such as pictures or spreadsheets, from other applications.
- Attach documents to appear as icons in the message.

Figure 12.1 shows a message with text, an attachment, and an embedded object.

Figure 12.1

You can attach documents and embed objects in your mail messages



You can have multiple compose forms open at the same time, making it simple to share text, objects, and file attachments between multiple messages. Mail also supports cutting and pasting between messages and from other Windows-based applications.

Attached files are represented in the message by their icons. Recipients using the new Mail client can launch attachments by double-clicking the file attachment icon (assuming the recipient has the appropriate application associated to the attachment). Besides launching attachments from the mail message, the recipient can save them to disk.

Because Mail supports object linking and embedding (OLE) you can embed objects in mail messages. Typical objects might include part of an Excel spreadsheet, part of a richly-formatted Word for Windows document, or a vibrant graphics object. Objects can be pasted into any location in the message and can easily be moved to another location and/or re-sized. Message recipients can manipulate and edit the embedded object without leaving Mail so long as they have the appropriate associated application for that object installed on their systems.

Easy Addressing

Mail users don't have to memorize the sometimes cryptic e-mail aliases for everyone in your workgroup. Instead, they can look up a person's real name in Mail's address book to identify the recipient for a message. The address book also enables users to track other information about workgroup members, such as office number and phone number.

The address book is provided by the address book library, AB.DLL, the software component used to manage the names of the users on the workgroup.

The address book also has a name-resolution feature. Here are two examples of how the name-resolution feature works:

- Suppose you are not sure about how to spell the name of the person to whom you want to send a mail message. You can type the name (taking your best guess at its spelling) into the address fields, then click the Check Names command button. Mail will compare the name to your default address list. If you typed the name correctly, it will display as a resolved name. Resolved names are underlined in the message header and are objects that you can delete, or copy and paste into other messages. If you typed the name incorrectly, an error message is displayed.
- Suppose you want to send a memo to the new guy named Dave or David Something. You can type “Dav” and choose Check Names to see a list of all names beginning with those letters. Then you can choose the correct name from that list.

Flexible Message Organization

Mail uses the traditional folder metaphor for organizing messages. Users can create their own folder organization to suit individual work styles.

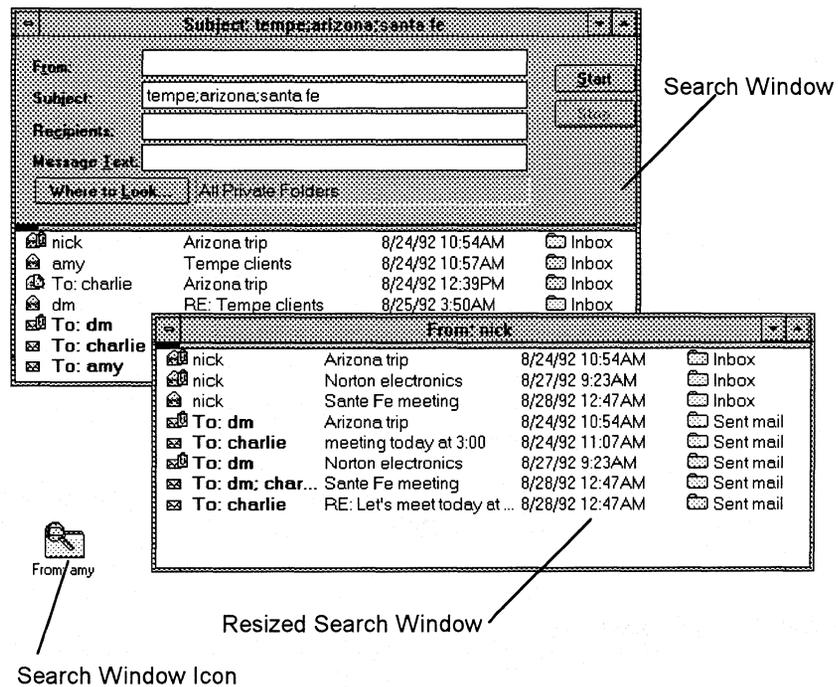
Mail supports shared folders, which are folders shared by all members of a postoffice. These can be used as topical bulletin boards for workgroup members. By choosing New Window from the Windows menu, a user can view the contents of multiple private and shared folders simultaneously.

Mail's search viewers give users the ability to view a cross-section of their folder organization by searching for messages that match any of a number of criteria. Users can have multiple folder viewers open simultaneously. Each of these windows displays both the hierarchy and the list of messages in a folder. Notification of changes to the database keeps all the multiple viewers in sync.

By using the Message Finder command on the File menu, users can also create search viewer windows. Figure 12.2 shows an example of two open search-viewer windows—one showing the search information and one resized to display only the search results. A search window icon is also shown.

Figure 12.2

You can create multiple windows for search viewers as well as folder viewers



The search-viewer windows are dynamically updated. A search runs in the background when first invoked and can be paused or resumed at any time. Once it has made a full scan of its domain, notification from the Notification engine keeps it up to date. This dynamically-updated search information can be maintained as permanent as a folder and will remain in tact even when you stop and restart Mail.

This is a handy feature for workgroup administrators, for example, who might want to track workgroup-related help requests in a separate folder. You might even keep a separate file for workgroup printer issues or upgrade information.

A particular message may be listed in several folders, but actually is stored in only one location. What appear to be duplicates are merely pointers to the original message.

Working Online or Offline

Both Mail and Schedule+ are designed to be used at the office and at home or on the road. When your computer is connected to the mail server and a session is active between them, you can work "online." If the mail server is temporarily unavailable, you can still work offline to compose outgoing messages or respond to received messages without interruption. You can also work offline when using a laptop or home computer to compose mail.

Mail keeps the messages each user creates offline in the local outbox until the user once again is working online. At that time, the user's outbound messages are transported to the postoffice on the mail server until the recipients collect them.

You can also export your message file (with an .MMF extension) to work away from the office. To do this, you can use the Export File command from the File menu. You can then select which folders you want to copy or move to an export message file (usually EXPORT.MMF). You can even specify a range of dates for the messages you want to export. Your personal address book is also copied with the messages to the export message file.

When you take messages back to the office, you can use the Import File command from the File menu.

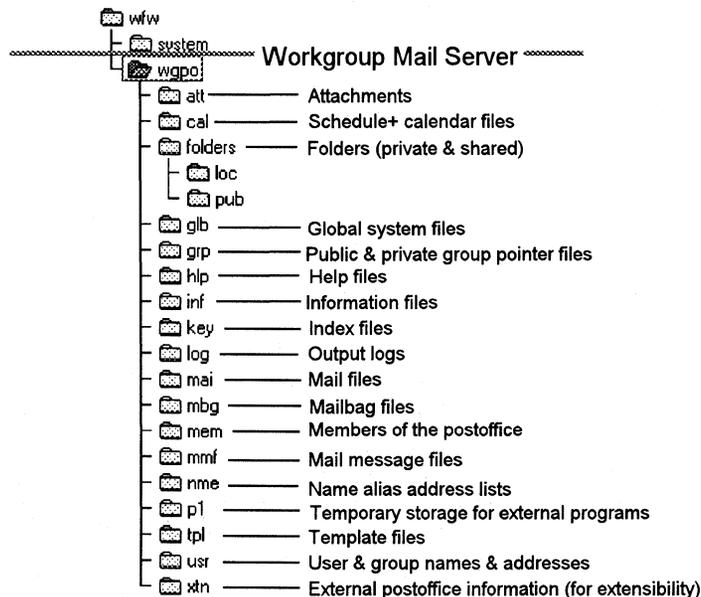
For more information about how to work offline, see the Mail Help topic called "Working Offline."

The Mail Postoffice

The postoffice, as we mentioned before, is no more than a directory structure. Figure 12.3 shows the postoffice directory structure:

Figure 12.3

Postoffice directory structure



All subdirectories must be present for the Mail system to function correctly. The following is a list of what's in each directory of the Mail postoffice:

- The ATT subdirectory contains encrypted file attachments.
- The FOLDERS subdirectory contains shared and private folders (with a filename extension of .FLD) for use by MS-DOS workstations. (Windows for Workgroups workstations' folders are in their .MMF files on the workstations.)
- The GLB subdirectory contains global system files for Mail. These files contain local user logon information and control files to generate mail files.
- The INF and TPL subdirectories contain file information relating to postoffice-defined templates. INF contains information files and TPL contains templates. ADMIN.INF and ADMIN.TPL contain template information for local postoffice users. (For more information about templates, see "User Information Templates" later in this chapter.)
- The KEY subdirectory includes index files that contain pointers to header records in the mailbag (.MBG) files.
- The MAI subdirectory stores mail messages in encrypted form until the recipients' workstations retrieve them.
- The MBG subdirectory contains mail headers that point to the mail (.MAI) files. For each file in this directory, there is a matching index (.KEY) file.
- The MEM subdirectory contains a list of the postoffice's members.
- The NME subdirectory contains pointer files for the name alias address lists. ADMIN.NME and ADMINSHD.NME list members of the postoffice address list.
- The GRP, LOG, MMF, USR, and XTN subdirectories are useful only if you decide to upgrade to a multi-postoffice configuration. This can be done with the Microsoft Mail and Schedule+ Extensions for Windows for Workgroups. In that case, USR is used to list user names and group names for the other network and XTN is used to list other external information.

Workgroup Postoffice Administration

The workgroup postoffice administrator is responsible for creating and managing the postoffice. Administering a workgroup postoffice is easy. The only difference between an administrator and other Mail users is that you can perform the following tasks from the mail server:

- Backup the postoffice (good to do on a regular basis)
- Add users to the postoffice
- Change user information, including passwords, if someone should forget his or her current one
- Check the status of shared folders

The workgroup postoffice manager library, WGPOMGR.DLL, is the software component on the postoffice that supports administrative functions such as adding or deleting users and changing passwords.

For more information about performing administrative tasks, see the *Mail User's Guide (Version 3.1 for the Windows for Workgroups Operating System)*.

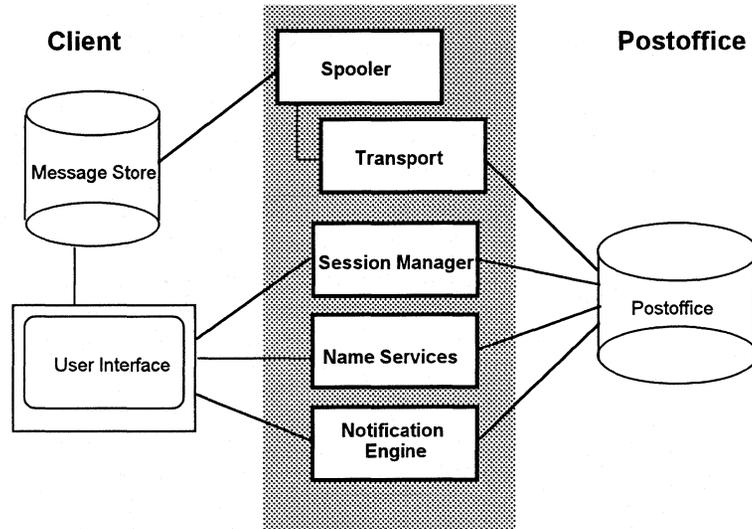
In the previous sections, we've discussed the key features of the user interface and the directory structure that make up the Mail server postoffice. In the next section, we describe how the client and server sides of Mail interact and also explore the components that make up the interface between the client and server sides.

Interface Between the Mail Client and Postoffice

Mail has a modular architecture. While some of the modules comprise the user interface and postoffice, most make up the interface between the Mail client and the Mail postoffice. Several powerful features are included in the interface between the user tools and the postoffice. Figure 12.4 shows four key components that make up this interface:

Figure 12.4

Components that make up the interface between client and server sides of Mail



This section describes each of these components.

The Mail Spooler and the Mail Transport

When you have a message to send, the mail spooler (MAILSPL.EXE) tells the mail transport (MSSFS.DLL) to move it from your outbox to the postoffice. When a message arrives for you at the postoffice, the mail spooler tells the mail transport to deliver it to your inbox.

The mail spooler's primary job is to parcel out system idle time to the mail transport. This enables the mail transport to transfer messages in a timely way but without interfering with the work you really want to do in the foreground. The spooler also provides a safety net for the transport by retrying any operations that fail.

The spooler also resolves address book entries, adds message recipients to the personal address book, and generates non-delivery reports. It checks for new mail and deletes mail from the server.

The Mail Session Manager

Once a message is received at the postoffice, the mail session manager (MAILMGR.DLL) logs it on the server. The mail session manager also validates users' identities and manages connections with the message store, directory, and transport.

The mail session manager is the component that enables you to work even when the mail server is unavailable (for example, when you work offline). Mail continues to work with the resources that are available at your computer and automatically connects to the server's resources when the server is again available. Maintaining the security of messages across transitions from offline to online and back is another feature of the mail session manager.

The Name Service

The name service (PABNSP.DLL) manages functions related to browsing and filtering lists of names. At the user interface, the name service is apparent whenever you browse file folders, specify search criteria, or ask for a recipient's name to be checked.

With the exception of the personal address book, the name service treats the directories it uses as read-only lists. Addresses as used in Mail consist of three parts:

- Display name
- Mail address type
- Mail address

The display name is usually the full descriptive name of a person, group, or resource. This is the name you most often see in the user interface.

The mail address type helps the transport route mail and provide the syntax of the mail address to the Mail client.

The mail address is the part actually used to identify the routing destination for a message.

The Notification Engine

When you receive new mail, the notification engine (STORE.DLL) lets you know it has arrived.

The notification engine also works with the session manager when you browse or filter messages. Folder management and message searches use the services of the notification engine. The message finder and other folders detect

notification of every change to a message. As the message changes, it is reflected in the appropriate folders and search windows.

For example, Mail enables you to have multiple windows viewing the same folder. If you delete a message from the folder in one of the viewers, it automatically disappears from the other because each viewer listens for notification of events like messages being deleted.

When a new message arrives and is written to the store by the mail spooler, the Inbox viewer is updated by the notification engine. When you place a message in your outbox, the notification engine alerts the spooler that there is a message to deliver to the postoffice.

So far we've described the standard features of Mail. In the following section, we'll show you ways you can customize Mail to suit the specific needs of your workgroup.

Customizing the Mail Client

You can use any of three Mail features to customize the way Mail works on a Windows for Workgroups workstation:

- *Custom commands* are commands you add to Mail menus. Once installed, custom commands appear to be built-in Mail features. You can install custom commands on an individual computer for use by a single Mail user or on a network file server to be shared by many Mail users.
- *Custom message types* enable you to define and install mail forms customized for your own requirements. You can install custom message types on an individual computer for use by a single Mail user or on a network file server to be shared by many Mail users.
- The *Messaging Application Program Interface (MAPI)* is a set of calls you can use to easily add mail-enabled features to other Windows-based applications.

Custom Commands

You can add custom functions to your Mail menus with custom commands. A custom command is one you add to an existing Mail menu to perform a unique function tailored for your needs. There is a second type of custom command not associated with any menu. Instead, this type of custom command can be set to run whenever the user starts Mail, ends Mail, or receives a new message.

A custom command is implemented as a dynamic-link library (DLL). Any executable code—including batch files or applications—can be run from a custom command. Here are some examples of how custom commands can be used to tailor Mail for specific needs:

- Displaying a window that contains information specific to the place the user works, such as a parts list or a list of stock quotes.
- Enabling the user to perform a query on a database without leaving Mail.
- Enabling the user to launch another application and open a specific file. For example, users can create a command to open their AUTOEXEC.BAT file in the Windows Notepad application.

The *Windows for Workgroups Resource Kit* disk includes the files needed to install custom commands described in this section. For more information about these files, see the Readme file on the disk.

Installing Custom Commands

You can install custom commands on the local computer for use by a single user or centrally on the postoffice to be shared by many users. Installing the command on the postoffice is handy because it eases installation and administrative tasks, and saves space on other users' hard disks.

Once installed, these custom commands can appear to users as built-in features. You can install up to 100 custom commands (but not new menus) to Mail. If you install several custom commands, you can add separator bars between groups of commands to help organize the menu's appearance.

To install custom commands on an individual workstation, you need to modify the user's MSMAIL.INI file for each custom command and add the appropriate DLLs to the user's computer. To install custom commands one time on the postoffice is easier than installing on several workstations. It requires modifying each user's MSMAIL.INI file only once, then adding the DLLs to the postoffice and modifying the postoffice's SHARED.INI file. The following sections explain how to install custom commands both ways.

Note Typically, the program that creates the custom commands automatically modifies lines in MSMAIL.INI and SHARED.INI so you don't have to. The information in the following sections is for your reference if you need to modify these files manually.

Starting Separate Applications

Mail expects that custom commands will be implemented as dynamic-link libraries (DLLs). However, APPEXEC.DLL, supplied on the *Windows for Workgroups Resource Kit* disk, can be used to call an application's executable file (.EXE) and pass a command line from the custom message type entry to it. The DLL can also pass information about the message to the application when it starts.

Although Mail always calls a DLL, the DLL can pass information to an application by using Windows dynamic data exchange (DDE), command line parameters, or by using a disk file. If the called application is already running, the DLL transfers information most efficiently using DDE. If a disk file is used, care should be taken so that multiple temporary files are not added to the user's disk. The application that implements the custom message type runs independently of Mail.

Installing Custom Commands on a Single Workstation

To install a custom command on an individual workstation, follow these steps:

1. Copy the DLL for the custom command to the WINDOWS directory of your hard drive.
2. Add a custom command entry to the **[Custom Commands]** section of the MSMAIL.INI file on your workstation. Use this syntax:

```
tag=Mail version; menu name; position; DLL name; command;  
event map; status text; Help file; Help context
```

(Note that this is a single command line in MSMAIL.INI. Don't add any carriage returns even if line-wrapping occurs. For more information about this MSMAIL.INI entry, see Chapter 6, "Windows for Workgroups Initialization File.")

3. Quit Mail if it is running, and then restart it. Mail rereads your MSMAIL.INI file and adds the custom command.

The Edit MSMAIL.INI Custom Command

This *Windows for Workgroups Resource Kit* includes a sample custom command entry suited to install at an individual workstation. This custom command adds an Edit MSMAIL.INI command to the Mail File menu. When you select this command from Mail, it displays the contents of your MSMAIL.INI file by opening it in the Windows Notepad.

To install this custom command, follow these steps:

1. Copy APPEXEC.DLL from the *Windows for Workgroups Resource Kit* disk to your WINDOWS directory.
2. Assuming that NOTEPAD.EXE and MSMAIL.INI are both located in C:\WINDOWS, add this entry to the **[Custom Commands]** section of the MSMAIL.INI file:

```
Sample1=3.0;File;Edit MSMAIL.&INI...; 8;  
APPEXEC.DLL; C:\WINDOWS\NOTEPAD.EXE  
C:\WINDOWS\MSMAIL.INI; ; Displays MSMAIL.INI; ;  
;
```

(Note that this is a single command line in MSMAIL.INI. Don't add any carriage returns even if line-wrapping occurs.)

3. Quit Mail if it is running, and then restart it. Mail rereads your MSMAIL.INI file and adds the custom command to the File menu.

You should now see the Edit MSMAIL.INI command on the File menu. (If Edit MSMAIL.INI is not listed on the File menu or an error message appears when you choose the command, check that you have followed each of these steps accurately.)

Note The average user has no need to edit the MSMAIL.INI file manually. This custom command is useful only to administrators or advanced users who regularly make customizations to MSMAIL.INI. See Chapter 6, "Windows for Workgroups Initialization Files," for more information about customizing this file.

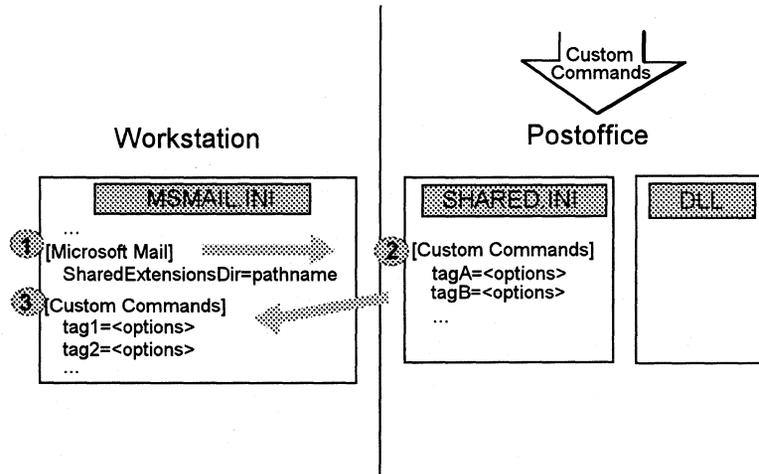
Installing a Custom Command on the Postoffice

If you have a custom command that you want to make available to multiple users, you can install the command on the postoffice instead of on each user's computer. When you create a new custom command, you must include a **SharedExtensionsDir=** entry in the **[Microsoft Mail]** section in each user's MSMAIL.INI file. This entry instructs Mail to check the server's SHARED.INI file for custom command entries.

Mail finds the **SharedExtensionsDir=** entry in MSMAIL.INI, then reads entries for custom commands in SHARED.INI before returning to MSMAIL.INI to read any custom command entries there. For example in Figure 12.5, Mail reads the **SharedExtensionsDir=** entry in MSMAIL.INI first. Then it reads the entries in the **[Custom Commands]** section of SHARED.INI (in this case, the lines labeled tagA= and tagB=). After reading all of the custom command entries in SHARED.INI, Mail reads entries in the **[Custom Command]** section of MSMAIL.INI (in this case, the lines labeled tag1= and tag2=).

Figure 12.5

File interaction with shared custom commands



The directory on the postoffice specified by **SharedExtensionsDir=** contains the SHARED.INI file as well as the DLLs that implement shared custom commands.

Mail can check a server for custom commands and custom messages in one of two ways:

- When you use a dynamic connection to the postoffice
- When you manually connect to the postoffice before starting Mail

The syntax you use with **SharedExtensionsDir=** reflects which way you choose to connect. The **SharedExtensionsDir=** entry uses the following syntax:

SharedExtensionsDir= [*pathname*] | [\\server\share\pathname] [*password*]

When you want to dynamically connect to the server's shared disk, use the \\server\share\pathname option with **SharedExtensionsDir=**. Use the *pathname* option with **SharedExtensionsDir=** to point to a manual connection to the server's shared disk.

To install a shared custom command, follow these steps:

1. Create a shared Mail extensions directory on the postoffice.
2. Copy the custom command DLL to the shared extensions directory.
3. A sample SHARED.INI file is included on the *Windows for Workgroups Resource Kit* disk. Copy this file to the shared extensions directory.

4. Add custom command entries to the **[Custom Commands]** section of the SHARED.INI file on the postoffice, using this syntax:

*tag=Mail version; menu name; position; DLL name; command;
event map; status text; Help file; Help context*

(Note that this is a single command line in SHARED.INI. Don't add any carriage returns even if line-wrapping occurs. For more information about this and other MSMAIL.INI and SHARED.INI entries, see Chapter 6, "Windows for Workgroups Initialization File.")

5. Add a **SharedExtensionsDir=** entry in MSMAIL.INI to reference the shared extensions directory. Use this syntax:

SharedExtensionsDir= *[pathname] | [\\server\share\pathname]
[password]*

(Note that this is a single command line in MSMAIL.INI. Don't add any carriage returns even if line-wrapping occurs.)

6. If you have chosen the manual connection method, make the necessary network connection using Windows File Manager.
7. Quit Mail if it is running, and then restart it.

Mail rereads your MSMAIL.INI file and adds the shared custom command to the specified menu.

8. Test the shared custom command.
9. When the custom command works successfully on the postoffice, modify each user's MSMAIL.INI file to have access to the shared extensions directory. (See steps 5 and 6.)

Empty Wastebasket Custom Command

The *Windows for Workgroups Resource Kit* disk also includes an Empty Wastebasket custom command. This command modifies your File menu in Mail. With it, you can delete the contents of the Deleted mail folder without having to quit Mail. This custom command is suitable to install as a shared custom command on the postoffice.

To install this custom command, follow these steps:

1. Copy the files SHARED.INI and EMPTYWB.DLL from the *Windows for Workgroups Resource Kit* disk to the shared extensions directory on the postoffice.

- Using the sample SHARED.INI file, remove the initial semicolon from these lines:

```
;Separator=3.0; File;; 7,,,,;
;SharedSample1=3.0; File; &Empty Wastebasket; 8;
  <ExtsDir>EMPTYWB.DLL;1;Deletes the contents of the Wastebasket;;
```

(Note that these are two individual command lines. Don't add any extra carriage returns in these lines even if line-wrapping occurs.)

- Add the following line in the **[Microsoft Mail]** section of the MSMAIL.INI file on the local computer:

```
SharedExtensionsDir=\\SERVER1\SHARE1\MAILEXTS
```

Replace the network path shown above with the appropriate path for your shared custom commands.

- Restart Mail and look at your File menu. The Empty Wastebasket command should be displayed there. If not, recheck your work, making sure you followed each of the preceding steps correctly.

For more information about the MSMAIL.INI and SHARED.INI entries, see Chapter 6, "Windows for Workgroups Initialization File."

Custom Message Types

A custom message type is a special type (or class) of message for delivery between two or more Mail recipients or mail-enabled applications. A custom message type can define a particular way to perform standard Mail operations such as composing, replying to, and forwarding messages. Custom message types can define a message's appearance and content, or the behavior of the forms displayed to the user when sending, reading, and replying to messages of that type.

Most custom message types are received in your inbox just like standard Mail messages. A second kind of custom message type doesn't appear in the inbox when it arrives, and yet is available to other mail-enabled applications. This allows mail-enabled applications to define their own message classes.

Custom message types have many similarities to custom commands. Like custom commands, a command for composing a message of a custom type can be added to an existing Mail menu. When the Mail user chooses this command, the custom message type DLL displays a dialog box or form that enables the user to compose a message of the corresponding type. These dialog boxes or forms can include features specific to the message type.

You can install up to 100 custom message types, but you can't add a new menu to Mail.

When a message of a custom type arrives in a Mail folder, it can appear in the message list the same as other standard Mail messages. But when read or replied to, the custom dialog boxes or forms associated with the message type can be displayed in place of the standard read or reply form. You can also design a custom message type that doesn't appear in Mail's inbox when it arrives, and yet is available to other mail-enabled applications.

Note If a custom message type is delivered to a user who does not have that custom message type installed, the message is treated as a standard message type.

Custom message types can provide special messaging functionality such as the following:

- Messages that are composed or read using a special form.
- Messages accessed by the user through an application other than Mail.
- Messages that are pre-addressed to a particular recipient.
- Messages that are used to order parts or services or to describe an event. The dialog boxes displayed when the user composes these types of messages can include special structured fields specific to the purpose of the message.
- Messages that help route work-flow events
- Messages that are specific to group scheduling
- Messages that enable some type of game, such as chess, to be played between two mail users.

Schedule+ is an example of an application that defines its own custom message types. Schedule+ uses the following custom message types:

- IPM.Microsoft Schedule.MtgReq—used to generate the Meeting Request form
- IPM.Microsoft Schedule.MtgRespP—used to generate the Positive Meeting Response form
- IPM.Microsoft Schedule.MtgRespN—used to generate the Negative Meeting Response form
- IPM.Microsoft Schedule.MtgRespA—used to generate the Tentative Meeting Response form
- IPM.Microsoft Schedule.MtgCncl—used to generate a Meeting Cancellation message

For more information about these custom message types, see Chapter 13, “Schedule+.”

The *Windows for Workgroups Resource Kit* disk includes the files needed to install custom message types described in this section. For more information about these files, see the Readme file on the disk.

Installing Custom Message Types

Installing a custom message type is similar to installing a custom command. Just like custom commands, you can install custom message types on an individual workstation or on the postoffice as shared custom message types. Typically, though, you'll install them as shared custom message types because you'll want both sending and receiving parties to take advantage of the custom type.

When you install custom message types on an individual workstation, you must modify the MSMAIL.INI file by adding a custom message type entry in the **[Custom Messages]** section. Each custom message type entry uses this syntax:

```
MessageClassName= Mail version; menu name; command name;  
command position; ExtsDir DLL name; ExtsDir command string;  
operation map; status line; ExtsDir Help file name; Help context;
```

(Note that this is a single command line in MSMAIL.INI. Don't add any carriage returns even if line-wrapping occurs.)

For shared custom message types, you must add the custom message type entry to the postoffice's SHARED.INI file instead of the workstation's MSMAIL.INI

file. Then, on each workstation using the shared custom message type, you must add a **SharedExtensionsDir=** entry. As with custom commands, when Mail sees this entry in MSMAIL.INI, it reads SHARED.INI for custom message type entries before reading custom message type entries in MSMAIL.INI.

For more information about the [Custom Messages] entries, see Chapter 6, "Windows for Workgroups Initialization File."

Installing Custom Message Types on the Postoffice

To install a shared custom message type on the postoffice, follow these steps:

1. Create a shared Mail extensions directory on the postoffice.
2. Copy the SHARED.INI file and custom message type DLL (and/or .EXE) to the shared extensions directory.
3. Add custom message type declarations to the [Custom Messages] section of the SHARED.INI file on the postoffice, using this syntax:

*MessageClassName= Mail version; menu name; command name;
command position; ExtsDir DLL name; ExtsDir command string;
operation map; status line; ExtsDir Help file name; Help context;*

(Note that this is a single command line in SHARED.INI. Don't add any carriage returns even if line-wrapping occurs. For more information about this and other MSMAIL.INI and SHARED.INI entries, see Chapter 6, "Windows for Workgroups Initialization File.")

4. Add a **SharedExtensionsDir=** entry in MSMAIL.INI to reference the shared extensions directory. Use this syntax:

SharedExtensionsDir= [pathname] | [[\server\share\pathname]
[password]]

(Note that this is a single command line in SHARED.INI. Don't add any carriage returns even if line-wrapping occurs.)

5. If you have chosen the manual connection method, make the necessary network connection using Windows File Manager.
6. Quit Mail if it is running, and then restart it.

Mail rereads your MSMAIL.INI file and adds the shared custom command to the specified menu.

7. Restart Mail and test the shared custom message type.

8. When the custom message type works successfully on your computer, modify each user's MSMAIL.INI file to have access to the shared extensions directory. (See steps 4 and 5.)

The Help Request Custom Message Type

The SHARED.INI file on the *Windows for Workgroups Resource Kit* disk includes a sample message type declaration that adds a Help Request command to the Mail menu and enables you to send help request messages using special forms.

This custom message type enables users to send help requests to the Mail administrator. The message type is defined in SHARED.INI. Any user's MSMAIL.INI that references the shared extensions directory can send and read the help request message. (If you don't want all users to use this type of message, install it only in the MSMAIL.INI file of the users who will send and read messages of this type.)

Figure 12.6 shows the Send Help Request form:

Figure 12.6

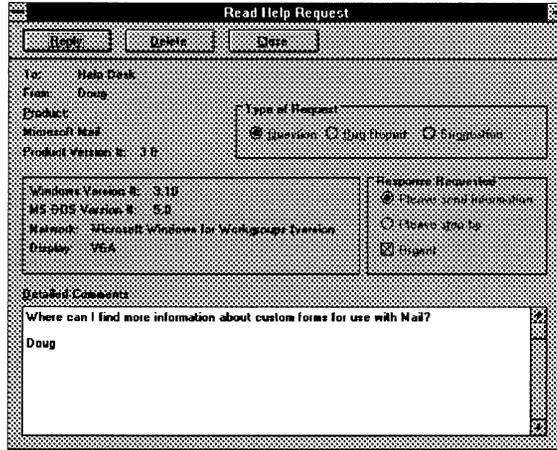
The Send Help Request form

This form appears when the user chooses the Send Help Request command in Mail. The Send Help Request form is automatically addressed to the recipient specified on the **DeliverTo=** entry in the HELPREQ.INI file. A drop-down selection box enables users to choose the product for which they need help. If the user closes the message without sending it, a dialog box appears with the prompt: "Do you want to save changes to *subject name* to 'Inbox'?"

The Read Help Request form in Figure 12.7 appears when the recipient of the help request opens the message:

Figure 12.7

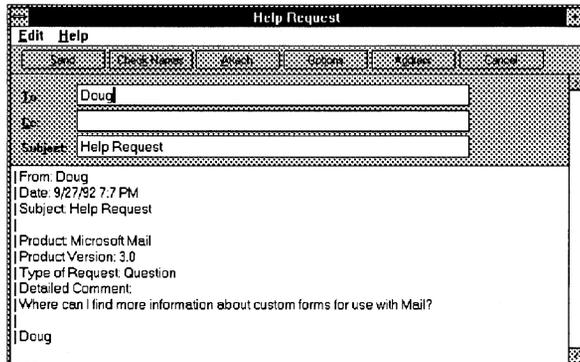
The Read Help Request form



The button controls for the reader differ from those on the Send Help Request form. Also, the Product and Product Version # fields appear as uneditable text. If the reader modifies and tries to close the form, the prompt “Save changes to message before closing?” appears. If the reader chooses yes, the Reply form in Figure 12.8 appears:

Figure 12.8

The Reply Help Request form



Installing the Help Request Forms

To install the help request forms described in the preceding section, follow these steps:

1. Create a shared Mail extensions directory on the postoffice.

2. Copy the files SHARED.INI, APPEXEC.DLL, HELPREQ.EXE, and HELPREQ.INI from the *Windows for Workgroups Resource Kit* disk to the shared extensions directory.
3. In the **[Custom Messages]** section of SHARED.INI, remove the semicolon at the beginning of the following line:

```
;IPM.Microsoft.SampleHelpRequest=3.0;Mail;&Help
Request...;2;<ExtsDir>APPEXEC.DLL;1;1100000000000000;Send Help Request;;;
```

(Note that this is a single command line in SHARED.INI. Don't add any carriage returns even if line-wrapping occurs. For more information about this and other MSMAIL.INI and SHARED.INI entries, see Chapter 6, "Windows for Workgroups Initialization File.")

4. Add this line to the **[Microsoft Mail]** section of MSMAIL.INI:

```
SharedExtensionsDir=\\SERVER1\SHARE1\MAILEXTS
```

Replace the network path shown above with the appropriate path for your shared custom message types.

5. Edit the supplied HELPREQ.INI file to specify the user to receive help request messages as follows:

```
DeliverTo=user_id
```

Replace *user_id* with the name of the Mail administrator who should receive help requests.

Note that the HELPREQ.INI file shows how an .INI file can be used to supply startup settings to your custom message type application. Not all applications require .INI files; your application's design will dictate whether an .INI file is needed or not.

6. Restart Mail and test the shared custom message type.

Mail rereads your MSMAIL.INI file and adds the Help Request compose command to the Mail menu. By choosing this command, a user can send a help request form to the Mail administrator specified in the HELPREQ.INI file.

7. Modify each user's MSMAIL.INI file to have access to the shared extensions directory. (See step 4.)

Messaging Application Program Interface (MAPI)

The Messaging Application Program Interface (MAPI) is a set of functions that developers can use to create mail-enabled applications. Mail includes a subset of 12 functions called Simple MAPI. Simple MAPI functions enable developers to send, address, and receive messages from within Windows-based application.

With Simple MAPI functions, developers can easily add the power of messaging to any Windows-based application. Simple MAPI supports the standard interface for simple integration of a Windows-based application with Mail.

All of the Simple MAPI functions are designed to be called from C or C++ programs, but they can also be called from high-level languages, such as Visual Basic, Actor, Smalltalk, and Object Vision. Simple MAPI functions can also be called from applications with macro languages that can call a DLL. Two examples of these are Microsoft Excel or Microsoft Word.

Some Simple MAPI functions include a user interface (a dialog box) but can also be called without generating an interface. The seamless integration with Mail is convenient for applications such as word processors and spreadsheets, which manipulate files that users may want to exchange through mail. The style of the user interface is not defined by Simple MAPI, so you can design your own.

Here's an example of how you can use Simple MAPI functions in your workgroup:

An application developer can incorporate mail functionality into his or her application by calling the MAPI functions. For example, if an application creates data files that need to be distributed to other users in a workgroup, the application developer can use the MapiSendDocuments MAPI function to create a mail messages and send the data files as an attachment to the mail message. Sending mail message is fully controlled from the application and the MAPI support library (MAPI.DLL). The user doesn't need to have the Mail program running to do this.

In this case, one function call is all that is required. The MapiSendDocuments function creates and initializes a message and supplies all the standard Mail dialog boxes for the user to send messages. The function can be compiled into the native code of the application, or if the application includes a macro facility that can dynamically link to a code library (a DLL), the developer can integrate the DLL as an added macro command.

Simple MAPI consists of these 12 functions:

<i>Function</i>	<i>Description</i>
MapiAddress	Addresses a Mail message
MapiDeleteMail	Deletes a message
MapiDetails	Displays a recipient details dialog box
MapiFindNext	Returns the ID of the next (or first) Mail message of a specified type
MapiFreeBuffer	Frees memory allocated by the messaging system
MapiLogoff	Ends a session with the messaging system
MapiLogon	Begins a session with the messaging system
MapiReadMail	Reads a Mail message
MapiResolveName	Displays a dialog box to resolve an ambiguous recipient name
MapiSaveMail	Saves a Mail message
MapiSendDocuments	Sends a standard Mail message
MapiSendMail	Sends a Mail message

Integrating Mail and Other Windows-Based Applications

Some Windows provide macros and support functions to enable them to use the capabilities of Mail directly from within the application. For example, the latest releases of both Microsoft Excel for Windows and Microsoft Word for Windows provide macros to send worksheets and documents, respectively, directly from within these applications.

As an example of how Mail can be integrated with applications, Windows for Workgroups provides a File Manager extension to add a Send Mail option to the File Manager menu and toolbar.

The Send Mail command displays all the user interface necessary for the user to send the currently selected files or executable files as file attachments to a message. The user can add message text around the file attachment, change the attached files, and address the message as usual. This reflects Microsoft's belief that messaging should be tightly integrated into the operating system.

Differences Between Windows for Workgroups Mail and Mail 3.x

The Workgroup postoffice architecture is similar to the architecture for Microsoft Mail for PC LANs version 3.x. This is so users with connectivity needs beyond their workgroup can upgrade easily.

The postoffices are identical with the following exceptions:

- No executable files or Help files are in the workgroup postoffice structure. Mail 3.x does include these files in its postoffice.
- The postoffice in Windows for Workgroups doesn't support connections to external postoffices or gateways.
- Default ADMIN.TPL and ADMIN.INF files are created to add the predefined extended user information in the workgroup postoffice structure.
- Mail 3.x includes an ADMIN account not created in Windows for Workgroups Mail.
- Mail 3.x includes an administration utility that is used to administer and configure the post office. The ADMIN.EXE program can be used to configure and administer a post office from any workstation on the network.
- Mail 3.x includes support files for external postoffice mail transfer through a network or modem. Windows for Workgroups Mail does not.

Functionally, Microsoft Mail 3.x is broader than Mail in Windows for Workgroups. Microsoft Mail 3.x includes online demonstrations, an extended Help index, a spell checker, and online tutorials not included with Windows for Workgroups Mail. The *Mail and Schedule+ Extensions for Windows for Workgroups* product can be used to upgrade the Mail application provided with Windows for Workgroups to the functionality supported by Microsoft Mail 3.x.

In summary, with Microsoft Mail 3.x you can connect multiple postoffices together, use more administrative tools, and connect to external postoffices and gateways. A Windows for Workgroups post office can be upgraded to a Mail 3.x post office by using the *Mail and Schedule+ Extensions for Windows for Workgroups* product as discussed in the following section.

Extending the Reach of Mail

The *Mail and Schedule+ Extensions for Windows for Workgroups* product (also called *Extensions*) provides enhanced connectivity and advanced administrative features for your Windows for Workgroups postoffice. It enables you to extend your mail and scheduling functionality in three main ways:

- By allowing multiple postoffices to be connected together
- By connecting to other mail and scheduling systems
- By providing remote dial-in support for Mail

This section provides more information about using the Extensions product. In addition to this information, consult Appendix B, "Microsoft Mail Configuration Guide," for more information about extending the reach of Mail.

Extensions Software

The Extensions package includes software, documentation, and licensing to extend the connectivity of your workgroup. It includes the following software components:

- Software to upgrade your workgroup postoffice to a full Microsoft Mail version 3.x postoffice.
- Additional server software:
 - o Advanced administration tools (including routing, directory synchronization, network group names, user access privileges, mail log files, delete old mail, delete old Mail/Schedule+ accounts).
 - o The Message Transfer Agent (External) component provides the process to connect postoffices (with physical or asynchronous link) and the platform for remote access.
 - o The Schedule Distribution component sends schedules to other postoffices.
- Microsoft Windows Mail client enhancements (including spell-checking, CBT, and advanced online Help).
- MS-DOS, Macintosh®, and Operating System/2® (OS/2®) Mail client software
- Schedule+ for Windows 1.0 client software

Connecting Multiple Workgroups

Extensions enables you to connect multiple workgroups together so that you can send Mail messages and Schedule+ meeting requests between many postoffices. To do that, you can use the software components in Extensions to upgrade a Windows for Workgroups postoffice to a full Microsoft Mail 3.x postoffice.

Another component, called the *Microsoft Mail Message Transfer Agent* (or *External*), is also included in the Extensions product. External acts as the postal carrier in your mail system, carrying messages between the multiple postoffices configured in your company.

Once upgraded, postoffices can be connected in two convenient ways: by direct cable connection (for example, across the network); or, where workgroups are not physically located next to each other, over a regular phone line. No matter how large your extended Mail network becomes, users will still have the same easy-to-use workstation software and the same easy access to a list of everyone accessible through Mail and Schedule+.

An advanced Microsoft Mail administration program is also included. This simple yet comprehensive tool enables the administrator to manage and control communications across the whole organization. Just as with Windows for Workgroups, administrators using Microsoft Mail can create and delete user accounts. Microsoft Mail also offers centralized control over who connects to a given postoffice, enabling administrators to track mail use, disk storage space, and, where long distance charges apply, the costs of sending mail to other postoffices. Performance can be enhanced with Extensions because you can also delete old mail stored on the server and recover disk storage space.

The Extensions product also lets you most efficiently configure and manage a multi-workgroup system over a local or wide area network. It adds features such as the global address list and automatic fault-tolerant directory synchronization to assist administrators and control maintenance costs. For example, you can set up postoffices so they regularly exchange information about changes made to their directories. This ensures that any changes made at a workgroup or departmental level will be reflected in the address list viewed in another part of the organization, even if it is in another part of the world.

Connecting to Other Mail and Scheduling Systems with Gateways

You may need to communicate with workgroups or organizations using electronic mail and scheduling systems other than Microsoft Mail and Schedule+. Microsoft Mail and Schedule+ Extensions help build organization-wide information sharing by connecting with other systems using products

from the Microsoft Mail suite of gateways. Microsoft provides a range of gateway products to connect you to most host-based, public, and LAN-based electronic mail systems and services.

The gateways are built to ensure that messages always get across to their intended recipients. Addressing messages remains an easy process because people using foreign mail systems will be listed in your Microsoft Mail global address list—all you need to know is the name of the person you need to communicate with.

Microsoft also enables users of other desktop systems to communicate with Windows for Workgroups users because the Microsoft Mail Extensions provide electronic mail software for MS-DOS, Macintosh, and OS/2 Presentation Manager (client licenses available separately). Mail users on these other desktop platforms will also have access to the easy addressing provided in Microsoft Mail.

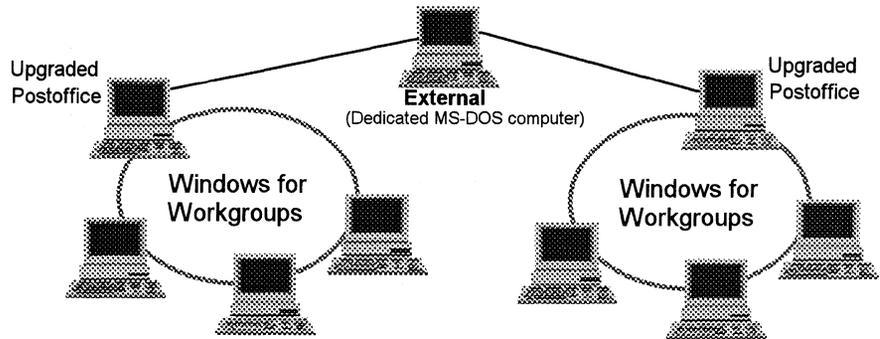
Connecting to Mail from a Remote Location

Extensions also includes software for dial-in access to Mail from anywhere that has a standard phone connection. Each user who wants to retrieve their mail messages remotely must install Microsoft Mail Remote (available separately) on the remote workstation or have a remote network connection.

Upgrading Connectivity

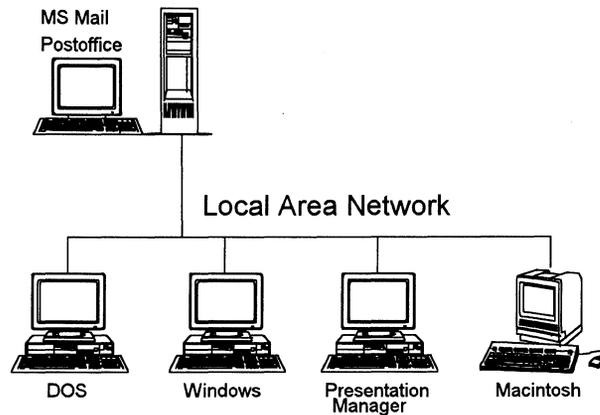
This section includes several upgrade scenarios for Windows for Workgroups postoffices, explaining what you need to install in each case to extend the functionality of your postoffice.

Connecting multiple Windows for Workgroups workgroups



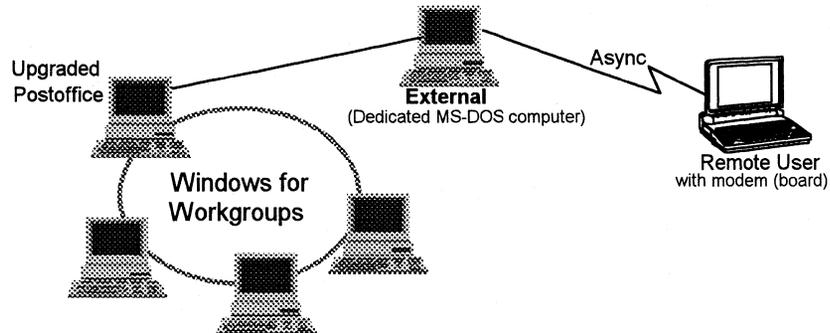
- Install Microsoft Mail and Schedule+ Extensions to upgrade the postoffice to a Mail 3.x postoffice.
- Install and configure a dedicated External computer to connect the upgraded postoffices.

Adding Macintosh and MS-DOS workstations to my workgroup



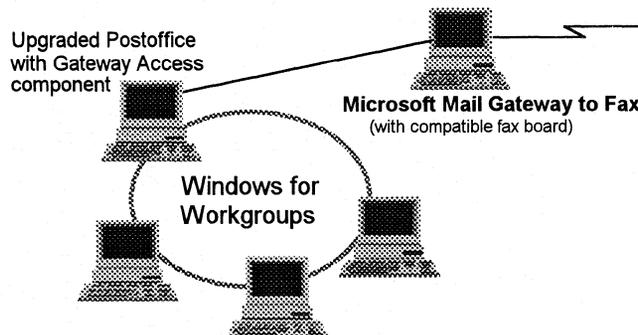
- Install Microsoft Mail and Schedule+ Extensions to upgrade the postoffice to a Mail 3.x postoffice.
- Install Mail client software on the new workstations.

Enabling remote access of Mail for Windows for Workgroups users



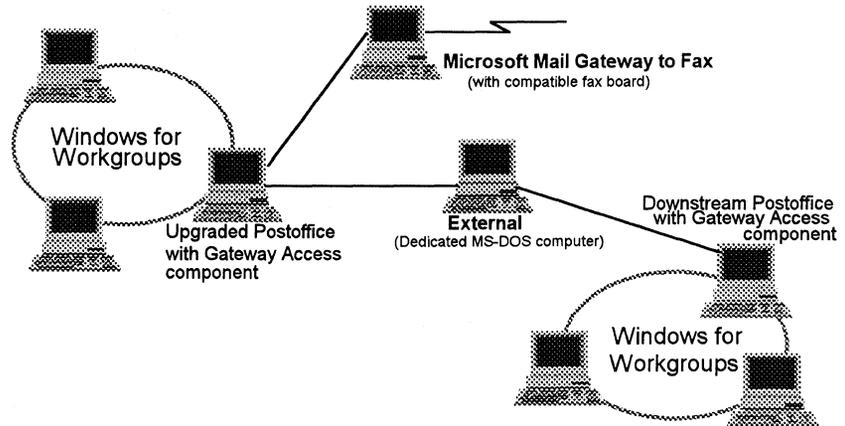
- Install Microsoft Mail and Schedule+ Extensions to upgrade the postoffice to a Mail 3.x postoffice.
- Install Microsoft Mail Remote and a modem (board) in each remote workstation.

Sending Mail messages and other documents via fax

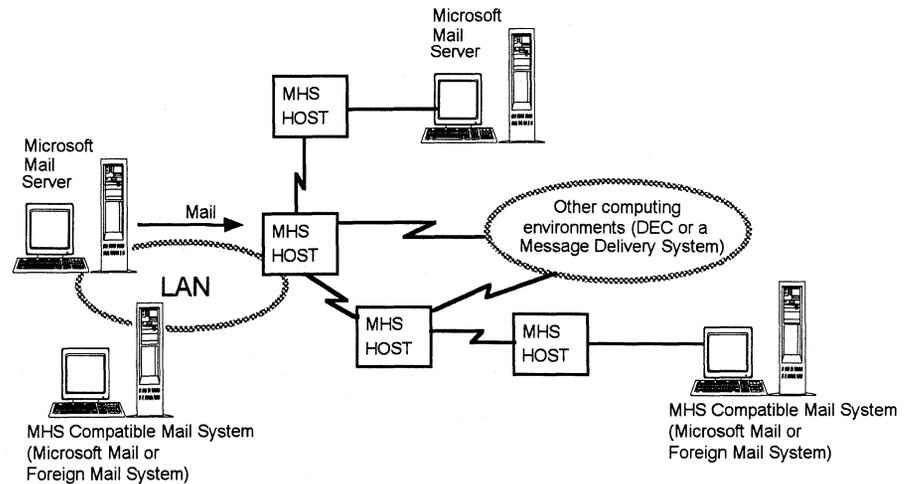


- Upgrade the postoffice with Gateway Access component.
- On a dedicated computer, install Microsoft Mail Gateway to Fax and a compatible fax board and connect to the postoffice.

If you have multiple postoffices (as in the following illustration), upgrade them with the Gateway Access component, then connect them with a dedicated External computer. Only one of your postoffices will be connected to Microsoft Mail Gateway to Fax computer.



Connecting another LAN-based mail system through Message Handling Service (MHS)



- Add the Microsoft Mail postoffice to the LAN of the MHS host.
- Install MHS Connectivity Manager to the postoffice computer (or to a dedicated computer connected to that postoffice).

Note Each client must be able to connect to the upgraded postoffice. MS-DOS clients will be able to connect to a Windows for Workgroups LAN using the Workgroup Connection. For Macintosh connectivity, you must move the postoffice to a server with Macintosh services such as LAN Manager or Novell NetWare.

Additional Workstation Licenses

Once you have upgraded your Workgroup postoffice, you can add new Mail and Schedule+ accounts by purchasing additional workstation licenses:

- Windows for Workgroups
- Workgroup Connection (MS-DOS).

Note These products include additional Mail client software for Windows and MS-DOS, respectively. The Schedule+ product is available only for use with Windows and Windows for Workgroups.

- Microsoft Mail Add Paks (for Macintosh and OS/2 workstations). Sold in units of 5, 20, 100, and 500 workstations, these Add Paks can also be used to license Windows and MS-DOS workstations that don't need the additional workgroup features in Windows for Workgroups or the Workgroup Connection.
- Schedule+ Add Paks (Windows workstations only). Sold in units of 5, 20, 100, and 500 workstations, these Add Paks can be used to license Windows workstations that don't require the additional workgroup features in Windows for Workgroups.
- Microsoft Office (for Windows and Macintosh). Each unit of Office contains one license for a Mail client .

Tips for Using Mail

This section offers tips for running the Mail application that ships with Windows for Workgroups.

Recreating the Mail Initialization Procedure

When you first run the Mail application, it asks whether to use an existing postoffice on the workgroup or to create the postoffice on your computer. If you accidentally select the incorrect option, you can use the following procedure to reinitialize Mail so that these options are available again:

1. Open MSMAIL.INI in a text editor such as NOTEPAD.EXE, then disable, or “comment out,” the **ServerPath=** and **Login=** entries by typing a semicolon (;) at the beginning of each of these lines:

```
; ServerPath=<something>  
; Login=<something>
```

2. Add or edit the **CustomInitHandler=** entry to read as follows:
CustomInitHandler=WGPOMGR.DLL, 10
3. Run Mail. The initialization process will begin and will once again prompt you for the location of the workgroup postoffice.

Changing the Name of the Postoffice Administrator

If you need to assign someone new as the postoffice administrator, use the following procedure. This procedure includes two parts—one for the future administrator and one for the former administrator. As noted below, each person must wait for the other before continuing to certain steps. These procedures must be done together.

On the future administrator's workstation

1. Copy these lines from the [**Custom Commands**] section of the former administrator's MSMAIL.INI file:

```
WGPOMgr1=3.0;Mail;;13  
WGPOMgr2=3.0;Mail;&Postoffice Manager...;14;WGPOMGR.DLL;0;;Administer  
Workgroup
```

2. Make a backup copy of your .CAL file from the postoffice. (This is the calendar file used for Schedule+.)
3. Use the Export Folder command from the File menu to move all Mail messages for the future administrator to a floppy disk.
4. After the former administrator is through with steps 1 through 3 described below, log on to Mail using the former administrator's account. This is the first account that was created on the postoffice.

5. From the Mail menu, choose Postoffice Manager. Then, change the name and password for this account to reflect the information for the future administrator.
6. If the former administrator is still part of the workgroup, create a new account for that person.
7. Replace the .CAL files for your account on the postoffice with the backup file from step 2.
8. Use the Import Folder command from the File menu to move your Mail messages from export folder files created in step 3 to the .MMF files on your workstation.

On the former administrator's workstation

1. After the future administrator has copied them, remove these lines from the [Custom Commands] section of your MSMAIL.INI file:

```
WGPOMgr1=3.0;Mail;;13
WGPOMgr2=3.0;Mail;&Postoffice Manager...;14;WGPOMGR.DLL;0;;Administer
Workgroup
```
2. Make a backup copy of your .CAL file from the postoffice.
3. Use the Export Folder command from the File menu to move all Mail messages for the future administrator to a floppy disk.
4. If you are remaining as part of this workgroup, log on to your new account (created by the future administrator in step 6 above). Change the password for the account.
5. Replace the .CAL files for your account on the postoffice with the backup file from step 2.
6. Use the Import Folder command from the File menu to move your Mail messages from export folder files created in step 3 to the .MMF files on your workstation.

Packaging Objects with UNC Pointers

Windows for Workgroups supports the use of the universal naming convention (UNC) inside packaged objects. This means that you can create an object with the Object Packager utility that includes a pointer to a file located on a network file share.

For example, instead of embedding a 1 MB Word for Windows document into a mail message, you can insert a packaged object that contains a UNC pointer

to the document on the network share. When the message is received, the recipient can double-click on the icon created by Object Packer to connect to the share and load the Word for Windows document.

Here's how to create a packaged object containing a UNC pointer to a Word document on a network share:

1. In Mail, choose the Insert Object command from the Edit menu.
2. Choose Package from the Insert Objects list. This starts Object Packager.
3. In Object Packager, choose the Command Line command from the Edit menu.
4. Type the UNC path and filename of the Word for Windows document file in the command box. For example, you can include something like the following:

```
\\COMPUTER2\WORDDOCS\BUDGET.DOC
```

(The network share should not have a password.)

5. Select the Insert Icon button and choose one of the available icons for this object. Choose OK.
6. From the Edit menu, choose Label, and then type a descriptive label for the icon.
7. From the Edit menu, choose Exit. When asked if you want to update the Mail message, choose Yes.
8. Send the Mail message.

Questions and Answers about Mail

This section answers some common questions about Mail.

Q How much disk space does Mail's postoffice require?

A You should allow approximately 2 MB of storage on the mail server to start. As the mail system is used, the amount of space it requires grows based on the number of users and the size of the messages and attachments being stored.

Q Is there a limit on the size or number of attachments in Mail?

- A** There is a limit to the *number* of attachments, not the size. The constraint is the size of a single mail message (not including the size of any attachments). The sum (size of the message text in bytes) + (the number of attachments) must always be less than 32K—allowing for many more attachments than most users need to get their message across.
- Q** Will the message finder search attachments to my mail messages?
- A** Not currently.
- Q** Can I retrieve deleted messages?
- A** Mail doesn't actually purge deleted files; it moves them to a Deleted mail folder. Deleted messages are not deleted until you empty the Deleted mail folder or quit the program. You can also configure mail not to delete messages when you quit the program.
- Q** How do I change my Postoffice location for the Mail client?
- A** In the MSMAIL.INI file, use the **ServerPath=** entry to redirect where Mail looks for the postoffice.
- Q** Can I use the spell checking libraries provided with other Microsoft Windows-based applications with Mail provided as part of Windows for Workgroups?
- A** The *Microsoft Mail and Schedule+ Extensions* product provides the spell checking libraries for use with Windows for Workgroups, and the accompanying software license allows the spell checking libraries to be used on all workstations that are a member of the workgroup. The software license covering the spell checking library provided with other Microsoft Windows-based applications such as Microsoft Word for Windows does not allow the spell checking libraries to be used with applications other than the one the libraries were provided with.

Schedule+

Schedule+ is an application that lets you plan and schedule meetings and appointments with others in your workgroup. As a mail-enabled application, Schedule+ works together with Mail to perform key functions such as sending meeting-request messages to other workgroup members.

This chapter describes the components that make up Schedule+, and shows how the Schedule+ application works with Mail. It also describes the custom message types used for Schedule+ and includes a section that answers common questions about Schedule+.

Related information

- *Schedule+ User's Guide*

Contents of This Chapter

Overview of Schedule+ Architecture.....	13-2
Initialization Files.....	13-3
Calendar Files.....	13-3
User Accounts for People and Resources	13-6
Custom Message Types for Schedule+	13-7
Schedule+ Interoperability	13-9
The Microsoft Mail File Format API (FFAPI).....	13-10
Schedule+ Interchange Format Files	13-10
Scheduling for the MS-DOS User	13-17
Extending the Reach of Schedule+.....	13-17
Questions and Answers about Schedule+	13-18

Overview of Schedule+ Architecture

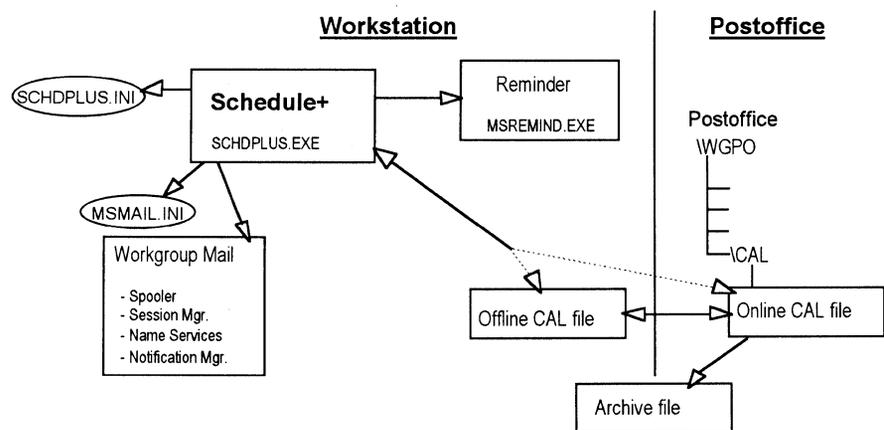
Schedule+ is an example of a mail-enabled application. It relies on Mail for certain functions, including support for logging on, accessing Mail's address book, sending messages, and receiving messages.

Because Schedule+ relies on Mail for key functionality such as sending and receiving messages, there is no special Schedule+ server. User account information for Mail is automatically translated to Schedule+ accounts.

Figure 13.1 illustrates the key components that make up Schedule+:

Figure 13.1

Overview of Schedule+



As shown in Figure 13.1, Schedule+ maintains both an online calendar and an offline calendar. This enables you to use Schedule+ as a stand-alone application or as a networked application. When you start Schedule+ offline, the local calendar file on your hard disk is read to the program. As you make changes to your calendar, those changes are written to disk immediately.

When you use Schedule+ online, it uses the online calendar file located in the **WGPO\CAL** directory of the workgroup postoffice. When run online, Schedule+ uses Mail to send meeting messages and to provide name service and logon support.

A separate terminate-and-stay-resident (TSR) program, **Reminder (MSREMIND.EXE)**, is an application used to notify the user of Schedule+ appointments. This application is typically installed in the Windows Startup group and runs in the background. (It can alternately be launched by Schedule+.) When it is included in the Startup group, as soon as you start Windows you are prompted to supply your logon name and password. Reminder, like Schedule+, uses Mail's logon functionality. The same logon is shared by Reminder, Schedule+, and Mail. Once you log on to one of these, you are automatically logged on to the others. You can log off from all programs at once by choosing **Exit and Sign Out** from either Mail or Schedule+.

The next few sections discuss the components shown in Figure 13.1 that make up Schedule+.

Initialization Files

Schedule+ uses two initialization (.INI) files—SCHDPLUS.INI and MSMAIL.INI. The SCHDPLUS.INI file stores the user's preferences, such as colors and the general options. It also stores the location of your calendar files and archives. MSMAIL.INI is the initialization file for Mail. Schedule+ uses this file to identify logon information and custom message types used for scheduling.

The access privileges you set for Schedule+ are not stored in an .INI file. They are stored in your calendar (.CAL) file on the postoffice so that Schedule+ can check for the following information:

- When a person wants to look at or modify your calendar—what privileges have you assigned for that person to access your calendar?
- When someone is inviting you to a meeting—do you have an assistant, and what is the assistant's name?

Schedule+ also uses MSMAIL.INI to identify your service provider and the path of your server. SCHEDMSG.DLL refers to MSMAIL.INI to find out how to launch Schedule+ when you choose the View Schedule button from within the Microsoft Mail client. (This button is displayed when you read received meeting requests.)

For more information about SCHDPLUS.INI and MSMAIL.INI entries, see Chapter 6, “Windows for Workgroups Initialization Files.”

Calendar Files

The main file used by Schedule+ is a calendar file, which contains scheduling information.

Each user has an offline calendar file (named with the person's logon and a .CAL filename extension) and an online calendar file (also having a .CAL filename extension, but with a random filename). The offline file resides anywhere you want, usually on your local hard disk in the SCHDPLUS directory. The online file location is determined by the Mail transport, and resides in the CAL directory on the postoffice.

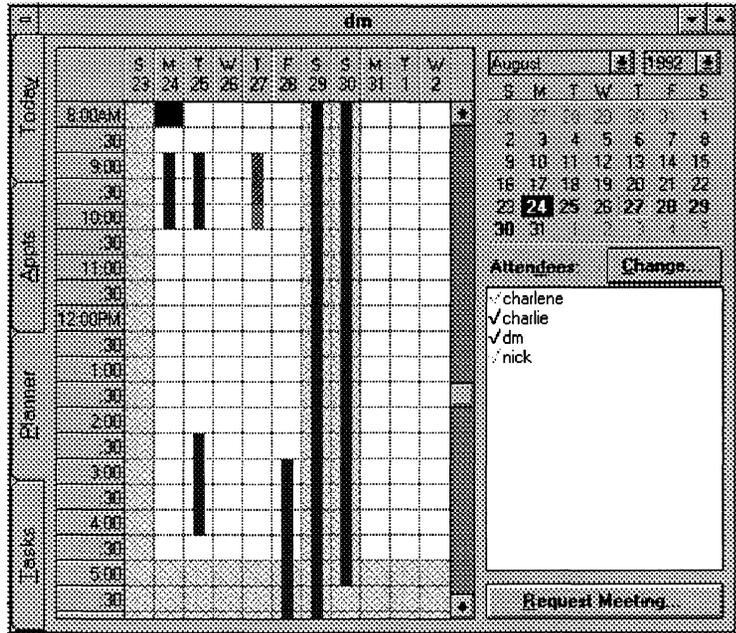
Whenever you want to access the schedules of other users on the postoffice, Schedule+ reads the data directly from the other user's online .CAL file. (Of

course, you can prevent others from viewing, reading, or modifying your schedule by setting access privileges to the schedule.)

A single postoffice file (.POF) in the CAL directory of the postoffice lists free/busy times for each user on the postoffice. When you want to schedule a meeting with another user, Schedule+ reads in the data in the .POF file to display an overlay of all users' busy times, as illustrated in Figure 13.2.

Figure 13.2

When you specify the attendees for a meeting, Schedule+ shows when everyone is available



With every attendee's free time displayed, it's easy to choose a time for your meeting.

When you schedule an event with others, Schedule+ automatically creates a Request form that you can send to the others asking them to attend the meeting. The "Custom Message Types for Schedule+" section later in this chapter describes the automatic Request and Response forms.

Merging Online and Offline Calendars

As we've already mentioned, users can use Schedule+ both online and offline, and Schedule+ is designed so that users always have an online and offline calendar file. You can use your online calendar while your workstation is connected to the Mail server. If you have an assistant, that person can also make changes to your online calendar. You can also use the offline calendar when the Mail server is unavailable or when you are away from the office.

Schedule+ has a mechanism to keep online and offline calendar files synchronized. Your local (offline) file will be updated to match your server (online) file automatically each time you start and quit Schedule+ and each time you connect to or disconnect from the server. It will also be updated in the background while you are running Schedule+ if your server file changes.

The merging of calendar files happens as follows:

- Schedule+ adds all of the appointments that you have added offline to the postoffice file.
- Schedule+ deletes all of the appointments that you have deleted offline to the postoffice file.
- When you change an existing appointment, the changes made to the local file always take precedence over the changes made to the postoffice file. If you change the text of an appointment offline and another person changes the start time, both changes will be applied because these changes are not in conflict. If you change the start time, end time, start date, end date, reminder notification time, or reminder notification date for an appointment, all of these attributes will be set to the values stored in the local file.
- When an overlap occurs as a result of merging the online and offline files, Schedule+ doesn't notify you specifically; instead, both appointments are entered in the calendar. For example, suppose you schedule an appointment on Jan. 1 at noon while working offline and your Schedule+ assistant creates an appointment at the same time on the postoffice file. Because your assistant has authority to add to and modify your scheduled activities, it appears to Schedule+ that you made two changes to the schedule at the same time. Schedule+ enters both appointments in the merged calendar files.

Offsite Calendar Files

As with Microsoft Mail, you can work with Schedule+ offline at home or on a portable computer. You can take a copy of your Schedule+ files from your local computer to work with while you're away from the office.

To take your calendar home or on the road, choose the Move Local File command on the File menu to copy the calendar to a floppy disk. Then load Schedule+ on the destination computer and choose Move Local File command again. The command moves the calendar file to the location you specify and changes the SCHDPLUS.INI so it searches for your calendar in the new location.

Archiving Old Calendar Information

As you might imagine, a calendar file full of scheduling information could soon get cumbersome. That is why Schedule+ includes an archiving feature.

Archiving enables users to remove past data from their calendars and store the data in a compressed format for later reference. This helps you to minimize disk space use on the server without having to completely discard past Schedule+ data.

User Accounts for People and Resources

Another key resource used by Schedule+ is the user account information maintained by Microsoft Mail. The list of user accounts for the workgroup is kept by Mail in the address book. It is also displayed in Schedule+ when you want to select attendees for a meeting.

With Schedule+, you can plan schedules not only for workgroup users but for resources such as conference rooms and AV equipment. This feature makes it easy for everyone in the workgroup to see when resources are available and to reserve the resources in advance.

To set up a resource on Schedule+, you first need to create an account for it by using Mail (just as you would for a workgroup member). Then you can log on to Schedule+ under that resource account and check "This account is for a Resource" in the General Options command. This automatically changes the default access privilege to "create appointments and tasks." That privilege means that people can reserve the resource by selecting it from the address book while they select attendees for a meeting. Users can also use the Open Other's Appt Book command to look at the calendar for the resource and book time in it directly. It is possible to assign an assistant to a resource and alter the default privilege to View Free/busy Times if you want somebody to be in control of allocating the resource.

Setting Up Resources in Schedule+

Creating a Schedule+ resource account is a two-step process for the administrator. First you must create a Mail account for each resource. Then you need to sign in to each resource account to set Schedule+ options for that account.

As the administrator, when you log on to each resource account, you are also logging into Mail. When Microsoft Mail is used with Windows for Workgroups, special steps are required to make it possible to log on to multiple accounts on the same workstation. The following information explains how.

Logging into multiple accounts from the same computer requires special steps because Microsoft Mail creates and uses an MSMAIL.MMF file that is specific to a particular Mail account. Normally, a separate .MMF file is required for each user. However, it isn't possible to create multiple MSMAIL.MMF files on the same computer. Because the MSMAIL.MMF file is protected by the Mail account password, you can work around this difficulty by simply giving each resource account the same password. As the administrator, when you set up the resource accounts from your computer, the password must be your Microsoft Mail password. This enables you to log on to Schedule+ under various accounts, while using the same .MMF file.

Note Be sure not to delete the MSMAIL.MMF file at any point.

When you sign in the first time, Mail will create the MSMAIL.MMF file. When you sign in again with the second resource account, Mail enables the existing MSMAIL.MMF file to be used because the password for the second account is the same as for the first account. You can continue signing in with each resource account and Mail will continue to use the same MSMAIL.MMF file.

When you sign in to the resource account from your computer, Mail will be able to use your MSMAIL.MMF file because the passwords for both the resource account and your account are now the same.

This process won't alter the contents of the MSMAIL.MMF file in any way, and you can continue using Mail as usual after the resource accounts are set up.

Custom Message Types for Schedule+

When you invite individuals to attend a meeting or when you reserve a resource, Schedule+ automatically prepares a Request form. When you respond to someone else's meeting request, Schedule+ generates a Response form. This section describes the forms Schedule+ automatically generates and the custom message types associated with each.

As described in Chapter 12, "Mail," Microsoft Mail enables you to create and use custom message types to send specific kinds of messages. Schedule+ uses this feature to define meeting requests, meeting responses, and meeting cancellations. In all, Schedule+ defines five message types in the [**Custom Messages**] section of your MSMAIL.INI file:

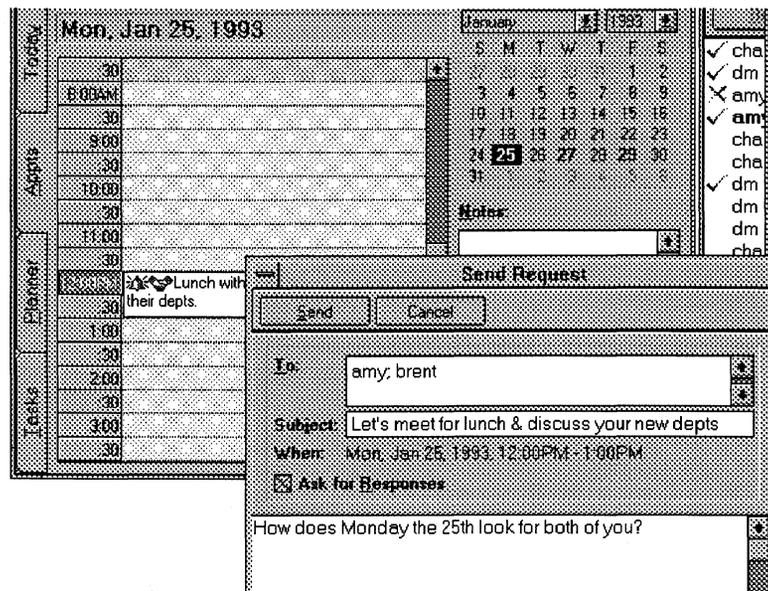
- Meeting Request (defined as message type "IPM.Microsoft Schedule.MtgReq")

- Positive Meeting Response (message type "IPM.Microsoft Schedule.MtgResp")
- Negative Meeting Response (message type "IPM.Microsoft Schedule.MtgRespN")
- Tentative Meeting Response (message type "IPM.Microsoft Schedule.MtgRespA")
- Meeting Cancellation (message type "IPM.Microsoft Schedule.MtgCncl")

These message types create special Request and Response forms that you can use to schedule meetings with others in your workgroup. For example, when you add a new appointment, choose a time, and specify attendees, the Send Request form shown in Figure 13.3 is displayed automatically:

Figure 13.3

Schedule+ includes a Send Request form associated with the custom message type IPM.Microsoft Schedule.MtgReq



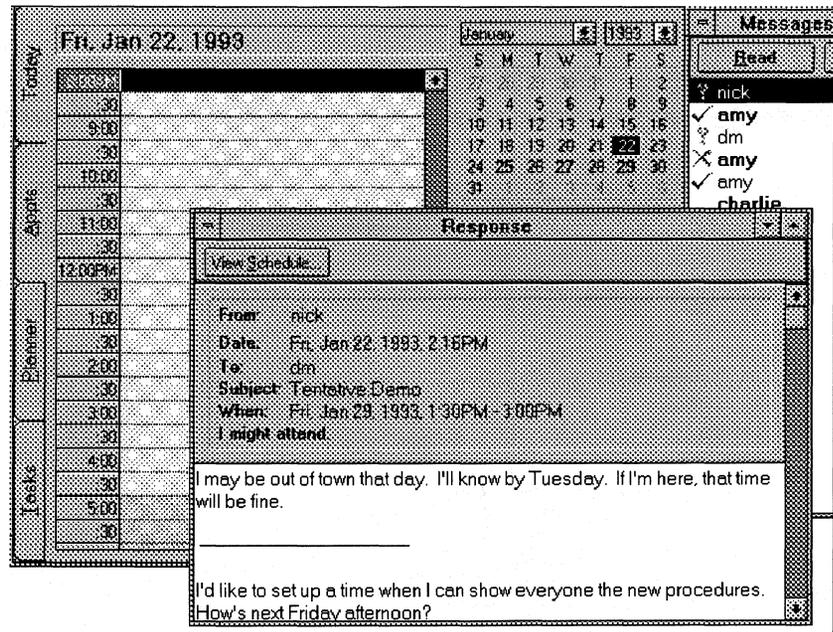
This message is deposited in the outbox and sent to users just as any mail message would be sent.

When a meeting attendee accepts the meeting, another automatic form, called the Response form, is generated and sent as a mail message. Three variations of the Response form (positive, negative, and tentative) are defined by custom message types defined by Schedule+. These three message types define the Response forms and a portion of the response message (such as "I might attend," shown in the Response form in Figure 13.4). They also append "Yes:", "No:", or "Tentative:" to the front of the original request title to create a response-message title. When the messages are displayed in the recipient's

Messages window, these custom message types also display symbols (✓, X, or ?) to the left of each message to make it easy to see responses at a glance.

Figure 13.4

Schedule+ also defines three Response forms for negative, positive, and tentative responses to meeting requests



Because meeting messages are sent using Mail facilities, they can be viewed from within Schedule+ or from within Microsoft Mail. Only one copy of each message is maintained, however, because what appear to be duplicates are really only pointers to the original. So when you delete the message in one view, it is also deleted in the other.

Schedule+ Interoperability

Schedule+ has a flexible architecture that makes communication with other schedule and calendar systems possible. Developers can modify their schedule and calendar software to share information with Schedule+ by using the following:

- Microsoft Mail File Format API
- Schedule+ Interoperability Specification

The following sections describes these options.

The Microsoft Mail File Format API (FFAPI)

The Microsoft Mail File Format API (FFAPI) version 3.0 makes it possible for calendar and schedule software vendors to modify their products so that they can exchange meeting requests with Schedule+.

With FFAPI and the Interoperability format, two types of communication can take place between Schedule+ users and users of other applications:

- Users can look at each other's free/busy information.
- Users can send meeting messages to each other through the mail system.

The free/busy information indicates for each 30 minute period of the day whether a user is free or busy. The range of dates covered is controlled by the administrator and can include the current month and up to 12 months in the future, but never includes information for past months. The scheduling client can present this information to let users find free times to schedule meetings with each other.

Free/busy information is distributed among servers by a program called Schedule Distribution. This program runs at a regular interval defined by the administrator. The Schedule Distribution program picks up changes to the free/busy information for each user on the server and sends it as a *schedule distribution message*—a specially formatted Mail message—to the other servers.

Meeting messages are sent from Schedule+ through Microsoft Mail to notify users of meeting requests and cancellations, and to let users send responses to the meeting organizer. Users can assign an assistant who manages their calendar for them. The assistant can initiate and respond to meeting requests on the user's behalf. The mail messages used for meeting messages consist of textual body and a binary attachment containing meeting information.

For more information or to receive the Schedule+ Interoperability Specification, call Microsoft Customer Service.

Schedule+ Interchange Format Files

The Schedule+ Interchange format is a text file with a .SCH extension. An interchange file can be imported to Schedule+ by choosing File/Import Appointments, then selecting Schedule+ from the Show Files Of Type list box. Likewise, a Schedule+ calendar can be exported by choosing File/Export Appointments, then selecting Schedule+ as the file format. The file format supports importing and exporting projects, tasks, appointments, meetings, and notes.

Interchange Format Syntax

The Interchange format file begins with a header describing the owner of the file (who the schedule belongs to), and when it was exported or created. It is followed by descriptors for projects, tasks, notes, standard appointments, and recurring appointments. Although meetings are supported in the interchange format, they are not documented here. It is assumed that meetings can be imported as standard appointments by the scheduling or calendar software that is exchanging data with Schedule+. The sequence of descriptors is not significant, except that projects must precede tasks.

The following section describes the syntax for individual descriptors.

If you export a Schedule+ calendar to the interchange format, you will find that each descriptor begins with a line defining a field called "aid." This is an appointment ID defined and maintained by Schedule+. It is not required input for an import file and shouldn't be included in the import file.

Key to Syntax

→ = tab	m = month or minute
<i>italic</i> = required input	d = day
non-italic = fixed field	h = hour
definitions	y = year

Header

SCHEDULE+ EXPORT BY *mailbox name* ON *m/dd/yy* AT *hh:mm AM/PM*

Non-Private Project

FixedAppt:

→szText

project description

→fTask → → → → →T

→aidProject → → → → → → *project id # (integer, number in sequence)*

End

Private Project

FixedAppt:

→szText

project description

→aaplWorld → → → → →Read

→fTask → → → → →T

→aidProject → → → → → → *project id # (integer, number in sequence)*

End

Note The "fTask" line is present for all projects; other lines, like the "aaplWorld" line above, are present only for private projects.

Standard Task

FixedAppt:

→szText

task description

→fTask → → → → →T

→bpri → → → → →priority # (1–35, where 1–9 = priority 1–9 and 10–35 = priority A–Z)

→aidParent → → → → →project id # of associated project

End

Standard Private Task

FixedAppt:

→szText

task description

→aaplWorld → → → → →Read

→fTask → → → → →T

→bpri → → → → →priority # (1–35, where 1–9 = priority 1–9 and 10–35 = priority A–Z)

→aidParent → → → → →project id # of associated project

End

Task with Due Date and Start Work Date

FixedAppt:

→dateStart → → → → →due date in format m-d-yyyy hh:mm

→dateEnd → → → → →due date in format m-d-yyyy hh:mm

→szText

task description

→fTask → → → → →T

→nAmtBeforeDeadline → → → → →integer # (with unit below, indicates start work before due date)

→tunitBeforeDeadline → → → → →unit: Day, Week, or Month (indicates start work before due date)

→bpri → → → → →priority # (1–35, where 1–9 = priority 1–9 and 10–35 = priority A–Z)

→aidParent → → → → →project id # of associated project

End

Note The dateStart and dateEnd lines ask for *hh:mm*. This must be supplied but won't be used for these date-oriented (rather than time-oriented) tasks.

Standard Appointment, Marked Private

FixedAppt:

→dateStart → → → → → start date and time in format m-d-yyyy hh:mm
(time is 24 hour)→dateEnd → → → → → end date and time in format m-d-yyyy hh:mm
(time is 24 hour)

→szText

appointment description

→aaplWorld → → → → → Read

End

Standard Appointment with Alarm

FixedAppt:

→dateStart → → → → → start date and time in format m-d-yyyy hh:mm
(time is 24 hour)→dateEnd → → → → → end date and time in format m-d-yyyy hh:mm
(time is 24 hour)

→szText

appointment description

→dateNotify → → → → → specification of alarm in format m-d-yyyy
hh:mm (24 hour time)→nAmt → → → → → integer defining how long before an appointment
the alarm should go off

→tunit → → → → → the unit (Hour, Day, Week, Month) defining the alarm

End

Note The dateNotify line asks for *hh:mm*. Place any time here; a value is required. The actual time used for the alarm is set in the nAmt and tunit lines. If the alarm is in minutes, the tunit line is not required.

Recurring Appointment

RecurAppt:

→ymdStart → → → → → start date of recurrence pattern in format m-d-yyyy

→ymdEnd → → → → → end date of recurrence pattern in format m-d-yyyy

→wgrfValidMonths → → → → → used to define recurrence pattern (see
next section for detail)→bgrfValidDows → → → → → used to define recurrence pattern (see
next section for detail)→trecur → → → → → used to define recurrence pattern (see next section
for detail)

→timeStart → → → → → start time in format hh:mm

→timeEnd → → → → → end time in format hh:mm

→szText

Recurring appointment description

End

Note The ymdEnd line is required only if the recurring appointment has an end date. If it does not, omit the ymdEnd line. See the following section, "Recurrence Patterns," for more information about this type of entry.

Notes for One Month

MonthNotes:

→date of Note in format m-d-yyyy

note #1 text

→date of Note in format m-d-yyyy

note #2 text

End

Note All notes for the month are listed together in sequence.

Recurrence Patterns

The recurrence pattern is defined by three fields. In combination, these fields define the following recurrence patterns: daily, weekly, biweekly, monthly defined by a pattern, monthly defined by a date, yearly defined by a pattern, and yearly defined by a date.

The three fields are used to define the recurrence types and patterns as follows:

Field	Description
wgrfValidMonths bbbbbbbbbbbb	<p>Enumerates valid months of the year, where bbbbbbbbbbbb is 12 bits, each being either T or F and representing January through December.</p> <p>All bits are T in every recurrence type except yearly. In the yearly case, all but one bit will be F. The T bit indicates which month the yearly appointment falls into in either of the yearly recurrence patterns.</p>
bgrfValidDows bbbbbb	<p>Enumerates valid days of the week where bbbbbbb is seven bits, T or F, representing Sunday through Saturday. (Note that in this syntax the week always starts on Sunday regardless of the user's start of week designation in Schedule+ Options.)</p> <p>All bits are T in the case of a monthly or yearly recurrence defined by the specific date. In the daily-every weekday, recurrence or the monthly or yearly recurrences defined as the x "weekday," the first and seventh bits are F and all others are T.</p> <p>In the monthly or yearly recurrence where field 2 is weekend day (see description of fields below), the first and seventh bits are T and all others are F.</p> <p>In all other recurrence types, one or more bits are T to indicate the days of the week on which the appointment falls.</p>

trecur [Week *bb* 0 |
IWeek *bbbb* 0 |
Date *x*]

trecur is followed by one of these terms to describe the recurrence type:

Week *bb* 0—daily or weekly; **trecur** and **bgrfValidDows** are used in conjunction to define the daily or weekly recurrence. The days of the week are indicated as described above.

Week *bb* 0—biweekly; **trecur** and **bgrfValidDows** are used in conjunction to define the biweekly recurrence.

IWeek *bbbb* 0—monthly or yearly, defined by a pattern instead of the date. If you look at the choices for the monthly recurrence in Schedule+, you'll see that the pattern is defined by two fields. For example, a monthly appointment can be on the first (field 1) weekday (field 2) of the month. The yearly pattern is defined by three fields. The first two are the same as the monthly fields and field 3 is the month of the year.

IWeek is followed by five bits with T or F; only one of these bits can be T. Field 2 of the monthly or yearly recurrence pattern is defined by **bgrfValidDows**. For example, if field 2 is a weekday, then **bgrfValidDows** would be FTTTTTF.

Field 3 in the yearly pattern is defined by **wgrfValidMonths**.

Date *x*—monthly or yearly, defined by the date *x* (the day of the month, 1–31). If it is a monthly occurrence, then all months are T in **wgrfValidMonths**. If it is a yearly occurrence, then one month of the year is indicated by T in **wgrfValidMonths**, and the rest are F.

Sample Schedule+ Interchange Format File

```
SCHEDULE+ EXPORT BY user1 ON 9/16/1992 AT 10:44 amFixedAppt:
→ szText
Personal Projects
→ aaplWorld→ → → → Read
→ fTask→ → → → T
→ aidProject→ → → → → 1
End
FixedAppt:
→ szText
pay bills (belongs to Personal Project)
→ aaplWorld→ → → → Read
→ fTask→ → → → T
→ bpri→ → → → → 4
→ aidParent→ → → → → 1
End
FixedAppt:
→ dateStart→ → → → → 9-27-1992 12:00
→ dateEnd→ → → → → 9-27-1992 12:00
→ szText
confirm presenters (belongs to Trade Show project)
→ fTask→ → → → T
→ nAmtBeforeDeadline→ → → → → 2
→ tunitBeforeDeadline→ → → → Day
→ bpri→ → → → → 1
→ aidParent→ → → → → 2
End
```

```

FixedAppt:
→ dateStart→ → → → → 10-17-1992 08:00
→ dateEnd→ → → → → 10-17-1992 09:00
szText
meeting with John
→ dateNotify→ → → → → 10-17-1992 08:00
→ nAmt→ → → → → 1
→ tUnit→ → → → → Month
End
FixedAppt:
→ dateStart→ → → → → 9-17-1992 14:00
→ dateEnd→ → → → → 9-17-1992 15:30
→ szText
phone call with ABC Corp
→ aaplWorld→ → → → → Read
End
RecurAppt:
→ ymdStart→ → → → → 9-16-1992
→ wgrfValidMonths→ → → → → TTTTTTTTTTTT
→ bgrfValidDows→ → → → → FFFTFFF
→ trecur→ → → → → Week FF 0
→ timeStart→ → → → → 16:30
→ timeEnd→ → → → → 17:30
szText
Weekly appointment without end date
End
RecurAppt:
→ ymdStart→ → → → → 9-01-1992
→ ymdEnd→ → → → → 6-01-1993
→ wgrfValidMonths→ → → → → TTTTTTTTTTTT
→ bgrfValidDows→ → → → → TTTTTTT
→ trecur→ → → → → Date 1
→ timeStart→ → → → → 16:30
→ timeEnd→ → → → → 17:30
szText
Monthly Appointment (the first of each month) with end date
End
MonthNotes:
→ 9-17-1992
call dentist
→ 10-31-1992
Halloween
End

```

Scheduling for the MS-DOS User

By using the Schedule+ Interchange format described in the preceding sections, third-party application developers are able to update their MS-DOS scheduling programs to integrate seamlessly with Schedule+. This means that users of the MS-DOS companion to Windows for Workgroups can also participate in group scheduling.

Extending the Reach of Schedule+

Schedule+ running with a workgroup postoffice limits users to scheduling meetings and sharing calendars or tasks with people in the same workgroup. No cross-workgroup scheduling is possible. It is possible to extend the reach of Mail and Schedule+ to other workgroups and throughout the company by purchasing the *Microsoft Mail and Schedule+ Extensions for Windows for Workgroups* (hereafter called *Extensions*). This product offers the following benefits to the Schedule+ user:

- Schedule+ users can share free/busy information and send meeting requests and responses to users on wide area networks.
- Schedule+ users can share full calendars with users on workgroups or on other LANs.
- Schedule+ users can invite any user on the electronic mail system to a meeting, even if that person is not using Microsoft Mail or Schedule+. The meeting request will be translated by the gateway into a text format that the foreign mail system can display.
- An administration program makes it easy to centrally control how much calendar data can be stored on the Mail server, to delete old calendar files, and to set up the server-to-server communication options with flexibility.

For more information about installing the Extensions product on a Windows for Workgroups postoffice, see the “Extending the Reach of Mail” section in Chapter 12, “Mail.”

Questions and Answers about Schedule+

This section answers some common questions about Schedule+.

Q Does Schedule+ support OLE or attachments (such as a meeting schedule)?

A No, but Mail does. You can create a message to invite attendees to a meeting and refer them to a separate mail message that includes OLE objects and/or attachments.

Q Does Schedule+ include all hours of each day as valid meeting times when the auto-pick feature is used to search for a meeting time?

A Auto-pick only looks at weekdays (Monday–Friday) within the time defined by your start-work and end-work settings in the General Options command to determine a meeting time. Schedule+ looks only at the meeting initiator's settings to determine valid times. Schedule+ doesn't recognize the non-working hours of other users and the user isn't able to define days other than Saturday or Sunday as non-working days.

Q Does Schedule+ support recurring meetings? For example, I'd like to schedule a weekly meeting with my group.

A Schedule+ supports recurring *appointments*, but not recurring *meetings*. Schedule+ doesn't support recurring meeting requests. Instead, you must request the meeting for a single occurrence and then remind the attendees to use the Recurring Appointment command to enter it for each week. Alternately, you can ask everyone attending your weekly meeting to set up their own recurring appointment for that time each week.

Q Are there options for altering the time and date displays (for example, using a 24-hour clock or a European date format)?

A Yes. From the Windows Control Panel, select the International icon. In the dialog box, you can specify the format used for the date and the time which will affect the displays in Schedule+ also.

Q Does Schedule+ support customized reminder sounds?

A Yes. You can modify the sound that is assigned to the Schedule+ Reminders in the Sound option in Control Panel. (You must first install the Sound Driver for PC-Speaker using the Drivers option in Control Panel.)

Q Because each resource requires a user account, does this mean that each resource requires a user license?

A No. Microsoft's licensing is per workstation running the software, not per user account used.

Q What are the dimensions of the print sizes in Schedule+?

A The formats are designed to match the most popular appointment book sizes. They are the following:

- Standard = 8 1/2" x 11 1/2"
- Junior = 5 1/2" x 8 1/2 "
- Pocket = 3 3/4" x 6 3/4 "

Q Are meetings scheduled by somebody else distinguished in any way?

A Yes. If you double-click on an appointment created by someone else, the name of the person who created it will be indicated at the bottom of the appointment details. However, the appointment is not identified differently by an icon or special color in the appointment book.

Q How can I view or modify someone else's calendar?

A Schedule+ enables you to grant access privileges to other users with the Set Access Privileges command from the Options menu. Once you have assigned privileges to another user, that user can view your calendar while being signed in under his/her own account by choosing Open Other's Appt. Book from the File menu.

Q What is the difference between the "Modify" and the "Assistant" privileges?

A The modify privilege is part of the set of privileges granted to your assistant. In addition to modifying privileges, the assistant can send meeting messages on your behalf by requesting a meeting while viewing your calendar. The assistant also receives meeting messages designated for you and can respond to them on your behalf.

Q When I specify someone as my Schedule+ assistant, do we both receive meeting messages for my meetings or does only my assistant receive them?

A You have the ability to specify either of the above. If you choose General Options from the Options menu, you will see a check box that specifies Send Meeting Messages Only To My Assistant.

Q Because I can view the meeting request messages from Mail or Schedule+, does this mean that there are actually two messages?

A No. The Schedule+ Messages box and the Mail inbox just provide two views of the same message. The message is only stored in one place. When you delete it from one view, it is deleted in the other.

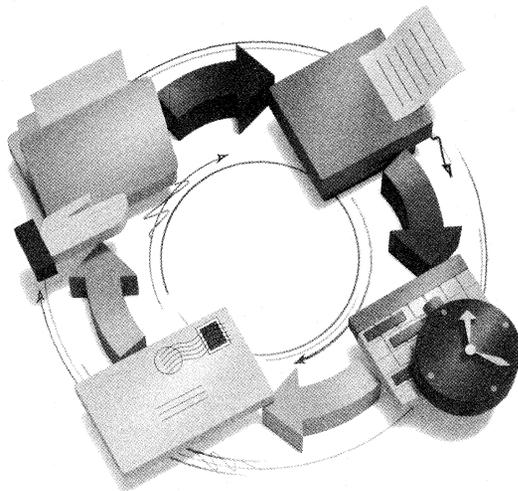
Q Does Schedule+ require Microsoft Mail? What happens if I want to upgrade to a multi-server network?

A Schedule+ requires only the software components that make up the client portion of Microsoft Mail. Schedule+ is transport-independent in the same way that the Mail client is transport-independent. To run Schedule+ against a different back-end, you need the following:

- Microsoft Mail.
- A messaging driver that enables the client to run against another transport.
- A scheduling driver that tells Schedule+ where to find the server calendars and how to access calendars of users on other servers. (This driver is very simple—about 6K of code.)

Microsoft includes a schedule driver for each messaging driver, so you really need nothing extra to run Schedule+ against a different back-end.

Troubleshooting Windows for Workgroups



Troubleshooting Windows for Workgroups

Chapter 14 Troubleshooting Windows for Workgroups 14-1

About Troubleshooting	14-3
Troubleshooting Setup	14-7
Troubleshooting Windows for Workgroups Configuration	14-11
Troubleshooting Windows for Workgroups Running in Standard Mode....	14-25
Troubleshooting Windows for Workgroups Running in 386 Enhanced Mode	14-27
Troubleshooting for MS-DOS-Based Applications	14-30
Troubleshooting Fonts	14-32
Troubleshooting Printing	14-33
Troubleshooting Networks	14-37
Troubleshooting GP Faults	14-42
Troubleshooting Multimedia	14-45

Troubleshooting Windows for Workgroups

This chapter provides information about troubleshooting Microsoft Windows for Workgroups for both general users and experts. If you have trouble installing Windows for Workgroups, or if Windows for Workgroups doesn't run as well as you expected, this chapter will help you find out why and show you how to isolate and solve common problems. In addition to the troubleshooting material provided here, the troubleshooting flowcharts provided in Appendix C, "Flowcharts for Troubleshooting Windows for Workgroups," will help you to troubleshoot problems that you may encounter with Windows for Workgroups.

Related Information

- *Windows for Workgroups Getting Started*: Chapter 6, "Troubleshooting."
- *Windows for Workgroups User's Guide*: Chapter 10, "Configuring Hardware."
- *Windows for Workgroups Resource Kit*: Appendix C, "Flowcharts for Troubleshooting Windows for Workgroups."

Contents of This Chapter

About Troubleshooting	14-3
Getting Started with Troubleshooting.....	14-3
Creating a "Clean Boot" for Troubleshooting.....	14-5
Troubleshooting Setup.....	14-7
Troubleshooting TSRs During Setup.....	14-8
Troubleshooting MS-DOS Mode Setup	14-9
Troubleshooting Windows Mode Setup	14-10
Troubleshooting Windows for Workgroups Configuration	14-11
Troubleshooting the Desktop Configuration.....	14-11
Troubleshooting TSR Compatibility Problems.....	14-11
Troubleshooting Display Problems.....	14-15
Troubleshooting MS-DOS Configuration Problems.....	14-20
Troubleshooting EMS Memory Problems.....	14-22
Troubleshooting Windows for Workgroups Running in Standard Mode....	14-25

Troubleshooting Windows for Workgroups Running	
in 386 Enhanced Mode.....	14-27
Troubleshooting for MS-DOS-Based Applications.....	14-30
Troubleshooting Fonts.....	14-32
Troubleshooting Printing.....	14-33
A Strategy for Troubleshooting Printing.....	14-33
Tips for Specific Printing Problems.....	14-35
Troubleshooting Networks.....	14-37
Network Adapter Troubleshooting.....	14-37
Network Communication Troubleshooting.....	14-38
Troubleshooting Novell NetWare.....	14-40
Troubleshooting GP Faults.....	14-42
Tips for Troubleshooting GP Faults.....	14-43
Dr. Watson and GP Faults.....	14-45
Troubleshooting Multimedia.....	14-45

About Troubleshooting

As a general orientation to troubleshooting, begin with these basic problem-solving actions:

- Try to reproduce the problem using the minimum system configuration described in “Creating a ‘Clean Boot’ for Troubleshooting” later in this chapter.
- Try to reproduce the problem in both standard mode and 386 enhanced mode, to see if the problem is mode-specific.
- Try to reproduce the problem while running another application and, if possible, while running the applications provided with Windows for Workgroups.
- Check the SETUP.TXT, README.WRI, PRINTERS.WRI, and NETWORKS.WRI files in the WINDOWS directory, looking especially for notes related to your specific hardware or for TSRs, device drivers, and other third-party software that you are running with Windows for Workgroups.

Windows is a stable environment that was tested on thousands of computers before it was shipped. However, your system might have a display adapter or other hardware that was not in existence during the latest Windows development cycle. This means you might be installing or running Windows for Workgroups on an untested configuration. This chapter, and the README files in your WINDOWS directory, can help you find solutions.

Getting Started with Troubleshooting

You can use this decision tree to begin troubleshooting for Windows for Workgroups.

Decision Tree for Troubleshooting Windows for Workgroups

- During Setup, you get an error message or your system crashes.
→ See “**Troubleshooting Setup**”
and **Flowchart 1.2**
- Your display is garbled or wrong or you have other display problems.
→ See “**Troubleshooting Display Problems**”
and **Flowchart 2.3**

Decision Tree for Troubleshooting Windows for Workgroups (continued)

- You have memory-related configuration problems.
→ See **“Troubleshooting MS-DOS Configuration Problems”**
and **Flowchart 1.2**
- You have problems using the upper memory area.
→ See **“Troubleshooting EMS Memory Problems”**
and **Flowchart 5.8**
- You cannot run Windows for Workgroups in standard mode
or you get not-enough-memory messages.
→ See **“Troubleshooting Windows for Workgroups in Standard
Mode”** and **Flowcharts 1.1 and 5.9**
- You cannot run Windows for Workgroups in 386 enhanced mode, or the
system crashes, or you have hard disk problems.
→ See **“Troubleshooting Windows for Workgroups in 386 Enhanced
Mode”** and **Flowcharts 1.1, and Series 2**
- You get out-of-memory messages or the application won't start
when you try to run an MS-DOS-based application under Windows for
Workgroups.
→ See **“Troubleshooting MS-DOS-based Applications”**
and **Flowcharts 5.4, 5.5, and 5.8**
- You experience problems when using TrueType fonts.
→ See **“Troubleshooting Fonts”**
and **Flowchart Series 3**
- You cannot print from Windows for Workgroups or have problems with
printing.
→ See **“Troubleshooting Printing”**
and **Flowchart Series 4**
- You have problems installing or running Windows for Workgroups, or you
have problems printing over the network.
→ See **“Troubleshooting Networks”**
and **Flowcharts 1.3, 4.2, and 4.4**
- You receive GP-fault messages while running Windows-based
applications.
→ See **“Troubleshooting GP Faults”**
and **Flowcharts 5.3, 5.4, and 5.5**
- You have problems with multimedia accessories.
→ See **“Troubleshooting Multimedia”**
and **Flowchart Series 6**

Creating a “Clean Boot” for Troubleshooting

Use a “clean boot” configuration to boot your system for troubleshooting problems with incompatible TSRs and drivers and problems with conflicting uses of the upper memory area. Remember that you must reboot your system after any change in AUTOEXEC.BAT or CONFIG.SYS for the change to take effect.

To create a “clean boot” configuration for troubleshooting

1. Back up your hard disk before beginning.
2. Make a backup disk. To do this, place a disk in drive A and type the following command at the command prompt:
format a:
3. Copy current configuration files to this disk by typing the following commands at the command prompt:
copy \config.sys a:
copy \autoexec.bat a:
4. Minimize the configuration in your CONFIG.SYS and AUTOEXEC.BAT, including only the drivers and programs necessary for the computer to boot and run Windows, and removing all other TSRs or drivers.

Windows for Workgroups with NetBEUI

For testing purposes, reduce the CONFIG.SYS file to:

```
files=40
buffers=20
device=c:\windows\himem.sys
device=c:\windows\lega.sys
stacks=9,256
shell=c:\dos\command.com c:\dos /p /e:256

device=c:\windows\protman.dos /i:c:\windows
device=c:\windows\exp16.dos
device=c:\windows\workgrp.sys
```

*if you have an EGA monitor
if using MS-DOS 3.3 or later
make sure command.com
is in the c:\DOS directory*

drivername varies by adapter

Reduce the AUTOEXEC.BAT file to:

```
c:\windows\net start
path=c:\c:\dos;c:\windows
set temp=c:\windows\temp
prompt $p$g
```

Windows for Workgroups with NetBEUI and IPX

For testing purposes, reduce the CONFIG.SYS file to:

```
files=40
buffers=20
device=c:\windows\himem.sys
device=c:\windows\ega.sys
stacks=9,256
shell=c:\dos\command.com c:\dos /p /e:256
```

*if you have an EGA monitor
if using MS-DOS 3.3 or later
make sure command.com
is in the c:\DOS directory*

```
device=c:\windows\protman.dos /i:c:\windows
device=c:\windows\exp16.dos
device=c:\windows\msipx.sys
device=c:\windows\workgrp.sys
```

drivername varies by adapter

Reduce the AUTOEXEC.BAT file to:

```
c:\windows\net start
c:\windows\msipx
c:\windows\netx
path=c:\;c:\dos;c:\windows
set temp=c:\windows\temp
prompt $p$g
```

5. Save your current initialization files on the floppy disk by typing the following commands at the command prompt:

```
copy \windows\win.ini a:
copy \windows\system.ini a:
copy \windows\progman.ini a:
copy \windows\protocol.ini a:
```

If you want to establish a system configuration that ensures optimal performance, or if you want to customize other features in Windows for Workgroups 3.1, you can also begin with a standard configuration by using the “clean boot” configuration. Make one change at a time to make troubleshooting easier.

Note Use the standard drivers included in the Windows for Workgroups 3.1 or MS-DOS 5.0 (or above) packages. Avoid third-party drivers unless you can confirm that such drivers have been tested and are known to be compatible with Windows for Workgroups 3.1. If you have questions about whether a particular driver is compatible with Windows for Workgroups 3.1, contact the driver vendor.

Troubleshooting Setup

Windows Setup is two programs bound into one executable file. The first program is MS-DOS mode setup, which installs the critical system files used to start Windows for Workgroups. The second program is Windows mode setup, which installs the Windows applications and the supporting files, sets up the Program Manager groups, installs printer and network drivers.

Many installation problems are caused by your system configuration—for example, by TSRs or third-party memory managers that conflict with the Windows memory managers. Failure can occur at four key points in setup:

- During initial detection of TSRs in your system
- During hardware auto-detection in MS-DOS mode setup
- During the initial loading of Windows for Workgroups
- Before the network adapter selection

If Setup fails at any of these stages, you might see an error message describing the problem, or setup might crash and return to the command prompt, or your system might require rebooting. The following sections discuss possible solutions to problems you might experience during setup.

Tip To see a list of switches that you can use to start setup, type `setup /?` at the command prompt.

Troubleshooting TSRs During Setup

Windows for Workgroups setup can check your system for memory-resident programs, looking in AUTOEXEC.BAT and CONFIG.SYS and also checking the programs currently running in memory. If Setup finds a program with which Microsoft testing has uncovered problems running with Windows for Workgroups 3.1, a message appears on your screen, giving you the opportunity to continue or quit the Setup program.

To check for conflicting memory-resident software with Windows for Workgroups setup:

- Type `setup /t` at the command prompt and press ENTER.

If you receive a message that incompatible TSRs or drivers are present when you try to run Setup, we recommend that you take the following steps.

To solve Setup problems related to incompatible TSRs:

1. Read SETUP.TXT on Windows for Workgroups Setup disk #1 for information about TSRs you are running. This information will tell you if you should install Windows for Workgroups 3.1 while this program is loaded.

If you choose to install Windows for Workgroups 3.1 without changing your system configuration, follow the instructions on the screen to continue with Setup.

2. If you decide that you should not install Windows for Workgroups 3.1 with the programs loaded, follow the instructions on the screen to quit setup.
3. Uninstall the TSR by following instructions in the program's documentation or by removing the filename from your AUTOEXEC.BAT or CONFIG.SYS file.
4. Reboot your computer and start Windows for Workgroups setup again.
5. After Windows for Workgroups is successfully installed, you might want to restore the memory-resident programs.

Some TSRs run with Windows for Workgroups if you start them after starting Windows for Workgroups. Read the SETUP.TXT file on Windows for Workgroups installation Disk #1 for information about the best way to run a specific TSR with Windows for Workgroups.

Check the entries in the **[incompTSR1]** and **[incompTSR2]** sections in the SETUP.INF file in your Windows SYSTEM subdirectory for up-to-date lists of software that conflicts with Windows for Workgroups setup. The TSRs listed in **[incompTSR2]** can cause problems with setup. We recommend that you uninstall any of this software before running Windows for Workgroups setup. The TSRs and device drivers listed in **[incompTSR2]** can cause problems if they are running when you run either setup or Windows for Workgroups 3.1.

Troubleshooting MS-DOS Mode Setup

In the first stage of MS-DOS mode setup, Setup uses an auto-detection scheme to determine your system's hardware configuration, specifically identifying the type of processor, computer, display adapter, mouse, and network adapter.

If setup hangs your system or pops out to the command prompt at this point and your system doesn't have any TSRs running, then the setup auto-detection scheme is probably causing the problem. Run Setup again, installing Windows for Workgroups in the same directory you chose the first time. Setup automatically detects that you had a problem during installation and will not try to detect the network adapter. If you still have trouble, restart your computer and type **setup /i** at the MS-DOS command prompt: The **/i** switch tells Setup to ignore auto-detection. You will need to tell Setup the type of network adapter that you have installed.

If you turn off auto-detection, Setup will select a standard configuration. When the hardware configuration screen appears, the list will be similar to this:

Computer:	MS-DOS System
Display:	VGA
Mouse:	Microsoft, or IBM PS/2
Keyboard:	Enhanced 101 or 102 key US and Non US keyboards
Keyboard Layout:	US
Language:	English (American)

If your system does not match the default settings, you might need to select the appropriate settings manually, following the instructions on screen.

These computers can be specifically selected during setup:

- MS-DOS System
- AST: All 80386- and 80486-based computers
- AT&T PC
- AT&T NSX 20: Safari Notebook
- Everex Step 386/25
- Hewlett-Packard (all computers)

- IBM PS/2 model L40SX
- IBM PS/2 model P70
- Intel386™ SL-based system with APM
- MS-DOS System with APM
- NCR: All 80386- and 80486-based computers
- NEC PowerMate SX Plus
- NEC ProSpeed 386
- Toshiba 1200XE
- Toshiba 1600
- Toshiba 5200
- Zenith®: All 80386-based computers

Troubleshooting Windows Mode Setup

The second point where Setup might fail occurs when MS-DOS mode setup is finished and Windows mode setup is starting. During MS-DOS mode setup, modifications are made to WIN.INI, identifying setup as the shell. The last thing that MS-DOS mode setup does is to start Windows for Workgroups. Then Windows for Workgroups begins loading the Windows files, in this order:

1: WIN.COM	4: MOUSE.DRV	8: GDI.EXE
2: KRNL286.EXE or KRNL386.EXE	5: DISPLAY.DRV	9: USER.EXE
3: SYSTEM.DRV	6: SOUND.DRV	10: WFWNET.DRV
	7: COMM.DRV	11: SETUP.EXE

If you are not running any TSRs and your computer hangs when Windows mode setup is starting to run for the first time, a file was probably unable to load. Most problems with setup failing at this point are due to an incorrect display driver selection. Either the auto-detection scheme incorrectly detected the system configuration or your custom choices were incorrect.

Certain display adapter chip sets lead Setup to detect the display adapter incorrectly. Typically, this error is due to an incomplete implementation of BIOS on older display adapters. For example, if the card is detected as a Video 7 VGA with 512K and it is an older computer, the display adapter might not be identifying itself correctly.

An easy test is to rerun the Setup program and select a basic display adapter (usually EGA or VGA) to see if that works.

For more troubleshooting tips about display adapters, see “Troubleshooting Display Problems” later in this chapter.

Troubleshooting Windows for Workgroups Configuration

Five main topics are described in this section:

- Desktop configuration problems
- TSR compatibility problems
- Display problems
- MS-DOS configuration problems
- Expanded memory management problems

Troubleshooting the Desktop Configuration

You might get an error message when you start Windows for Workgroups that indicates a .GRP file is invalid or damaged, and that instructs you to reconstruct the group. This error can occur if the following happens:

- A group (.GRP) file has been deleted or damaged.
- The files were moved into another directory, but the PROGMAN.INI file was not updated to reflect this change.

You might also need to rebuild groups if no group windows or icons appear when you use Program Manager after starting Windows for Workgroups. To rebuild groups, follow the procedure outlined in Chapter 2, "Application Basics," in the *Windows for Workgroups User's Guide*.

Troubleshooting TSR Compatibility Problems

Setup does not detect all memory-resident programs that might cause problems running with Windows for Workgroups. Also, if you run a TSR after installing Windows, Windows for Workgroups itself cannot detect a program to warn you if there is a problem. Two kinds of problems might occur:

- The interaction of the program with Windows for Workgroups might result in an unexpected system response such as a system crash or an incorrect display.
- Either the program or Windows for Workgroups might require that you take special steps to get these programs to run together.

If you have problems running Windows for Workgroups or a TSR program, check the list in SETUP.TXT in the WINDOWS subdirectory to see if there is a known conflict and a solution. Otherwise, begin troubleshooting by removing the program from memory to see how well Windows for Workgroups runs when the program is not present.

TSRs That Cause Problems with Setup

These programs might cause problems with setup and should be removed before installing Windows for Workgroups, but should work with Windows for Workgroups after installation:

- ASP Integrity Toolkit version 3.7 causes various problems with setup.
- Data Physician Plus version 2.0 (VirAlert) causes problems during setup. Version 3.0 appears to correct this problem.
- Norton Anti-Virus version 1.0 can cause setup to crash. Version 1.5 appears to correct this problem.
- PC-Kwik version 1.59 might cause setup to crash. Version 2.0 appears to address this problem.
- SoftIce Debugger causes Windows mode setup to crash.
- Vaccine causes setup to hang.
- VDefend, a PC-Tools Deluxe TSR, should not be in use during installation, but you can run it with Windows after setup is complete.
- Virex-PC version 1.11 causes setup to crash.
- ViruSafe version 4.0 causes setup to crash. Version 4.5 appears to correct the problem, but its authors recommend that you run ViruSafe with the /C-switch with Windows for Workgroups.

TSRs That Cause Problems Running with Windows for Workgroups

These programs cause problems running with Windows for Workgroups 3.1:

- ANARKEY version 4.00 might prevent Windows for Workgroups 3.1 from loading under certain circumstances. Version 4.01 appears to correct these problems.
- APPEND, an MS-DOS utility, interferes with the ability of Windows and Windows applications to build valid paths for files they want to access. Do not use APPEND with Windows for Workgroups 3.1.
- DOSCUE, a command-line editor, produces unreliable results with Windows for Workgroups 3.1. We recommend that you do not use this program with Windows for Workgroups 3.1.
- GRAPHICS, an MS-DOS utility, is loaded into all VMs for MS-DOS-based applications running under Windows for Workgroups 3.1. Printing from those VMs can be unpredictable, because the VM that has the focus sends its output to the printer, and printing becomes intermixed if you switch between MS-DOS-based applications.

- JOIN, an MS-DOS utility, works fine with Windows for Workgroups 3.1 if you do not change the state of the joined drives during a Windows for Workgroups 3.1 session. This includes adding or removing joined drives from within Windows for Workgroups. We recommend that you do not use JOIN when running Windows Setup or Windows for Workgroups 3.1.
- LANSight™ version 2.0 is a utility for controlling and monitoring workstations attached to a Novell network, and it requires TSRs to be loaded on the workstations before Windows for Workgroups is started. If a message asking permission to view the screen is sent from the supervisor workstation to a workstation running Windows for Workgroups 3.1, the workstation running Windows for Workgroups might require rebooting.
- Lockit version 3.3 (PC Vault, Secure It) strip spaces out of the **load=** entry in WIN.INI. Version 3.3+ appears to correct this problem.
- MIRROR, an MS-DOS utility, can't be removed from memory with the **mirror /u** command if it is loaded from Windows standard mode using the File Run command.
- Newspace version 1.07, a disk compression utility, is not compatible with Windows for Workgroups 3.1.
- Norton Utilities® version 5.0 programs Diskreet and Ncache are not compatible with Windows for Workgroups 3.1 and can cause your system to hang. Version 6.01 solves this problem, with some limitations.
- Norton Utilities version 6.01 DiskMon, the disk monitoring program, may cause inconsistent results under Windows for Workgroups 3.1 if you are trying to access files in a directory that DiskMon is monitoring. We do not recommend that you use this TSR with Windows for Workgroups 3.1.
- Printer Assistant is a TSR that allows workstations on NetWare 286 and 386 to share a printer. Some Windows applications print garbage mixed with correct output when this TSR is in use.
- XGAAIDOS.SYS causes Windows for Workgroups 3.1 to freeze at the opening logo in standard mode.

TSRs That Need Special Consideration

These programs will run with Windows for Workgroups if you take special action to make the two programs work together:

- BOOT.SYS, a power user's tool for booting under multiple configurations from a menu, creates several sections in CONFIG.SYS and AUTOEXEC.BAT. Windows for Workgroups Setup only modifies the first section in CONFIG.SYS and AUTOEXEC.BAT. You need to manually modify the other sections to use Windows for Workgroups 3.1 with the alternate configurations.

- Doubledisk version 2.5 creates “phantom” disk drives that Windows for Workgroups 3.1 might try to access. Vertisoft (makers of Doubledisk) has a DRVOFF utility to make these phantom drives invisible to Windows for Workgroups 3.1. Contact Vertisoft to get this utility if you want to run Doubledisk with Windows for Workgroups.
- FASTOPEN, a RAM resident MS-DOS utility, might need to be removed in low-memory situations for Windows for Workgroups 3.1 to load and function properly. Also, FASTOPEN causes File Allocation Table (FAT) problems when used with disk defragmenting utilities.
- LaserTools Control Panel version 2.2 might cause your system to hang in both standard and 386 enhanced modes if the TSR is loaded from within Windows for Workgroups 3.1. If you use this product, we recommend that you load it before starting Windows for Workgroups 3.1.
- The LANtastic KBFLOW TSR should not be loaded before you start Windows for Workgroups 3.1. To use KBFLOW, start Windows for Workgroups 3.1 first and run the TSR from within Windows for Workgroups.
- Le Menu version 1.0 can cause environment information such as PATH, PROMPT, and WINDIR to be lost when you run an MS-DOS-based application under Windows for Workgroups. If you are going to start Windows for Workgroups 3.1 from a Le Menu menu option, we recommend that you set up Windows for Workgroups 3.1 as a batch file menu option so that COMMAND.COM can be loaded before Windows for Workgroups 3.1. (See the Le Menu documentation.)
- Logitech Mouse Software versions 5.0 and 6.0 must be loaded from within an MS-DOS-based application after it is launched in Windows for Workgroups if you want to use Click and Logimenu, even if the TSRs were loaded before starting Windows for Workgroups 3.1. After quitting Windows for Workgroups 3.1, reload these TSRs for them to function from the command prompt.
- Norton Utilities version 6.01 Ncache program prevents you from creating a permanent swap file if this caching program is on a disk drive. Windows for Workgroups 3.1 will work with this program, however.
- PC-Tools Deluxe™ version 6.0 (Desktop) might cause your computer to hang if you launch it from an MS-DOS-based application running under standard mode Windows. In 386 enhanced mode, it might cause your computer to reboot. Version 7.0 of PC-Tools appears to correct these problems. Even with version 7.0, however, you should not run Desktop from within an MS-DOS-based application if the TSR was loaded prior to Windows for Workgroups 3.1. We recommend that you run Desktop from a PIF.
- Pyro! version 1.0, a screen saver, will blank the screen after the delay period has expired if it is loaded before Windows for Workgroups is started. At that point, it is impossible to restore the Windows screen.

- SideKick® versions 1.0 and 2.0 and SideKick Plus cause several problems with Windows for Workgroups 3.1. We recommend that you run SideKick from within Windows for Workgroups with a PIF instead of loading it as a memory-resident program.
- SPEEDFXR is not compatible with Windows for Workgroups 3.1 as a TSR. Running SPEEDFXR -X appears to work better with Windows for Workgroups 3.1, because it does not remain in memory.
- SUBST, an MS-DOS utility, works fine with Windows for Workgroups 3.1 if you do not change the state of the substituted drives while running Windows for Workgroups. This includes adding or removing substituted drives from within Windows for Workgroups.
- Trantor T100 Host Adapter Driver (TSCSI.SYS) identifies the SCSI hard disk drive as a removable drive, so your system might crash if you try to access this drive through File Manager. Other unexpected results can occur while deleting files through File Manager.

For information about running network drivers with Windows, see “Troubleshooting Networks” later in this chapter.

Troubleshooting Display Problems

Many problems with garbled displays, inadequate screen refreshing, and other odd behavior are related to the display adapter or display driver. If you cannot find another explanation in this chapter for faulty behavior in your system, check the details in this section closely for a description of symptoms you may be experiencing.

Display drivers actually have three parts:

- **The Windows display driver** (*.DRV; for example, VGA.DRV), which contains the code that knows how to talk both to the display adapter board (hardware) and Windows (software).
- **The grabber**, which contains the code that supports data exchange between Windows and non-Windows applications. The standard mode grabber only supports copying and pasting text; the 386 enhanced mode grabber supports copying text and graphics up to the standard VGA resolutions (video mode 13). The grabber is built into the display driver file, but requires supporting font files. These files use the filename extension .2GR for standard mode and .3GR for 386 enhanced mode (replacing the .GR2 and .GR3 extensions used in Windows 3.0).
- **The Virtual Display Driver** (VDDx.386), which provides virtual display support for running non-Windows applications in virtual machines in 386 enhanced mode.

The naming conventions listed parenthetically are not required. However, most drivers use these naming conventions.

Display-related problems typically fall into two categories:

- Problems with third-party display drivers (those provided by the display adapter manufacturer).
- Problems with display adapters that use nonmaskable interrupts (NMI).

This section will help you determine the source of and solutions for display problems.

1. Windows or Windows applications are not displayed correctly.

In this case, the display driver may be incorrect or corrupted, and a correct driver must be installed. Or the problem may be related to NMIs, described in item #4 of this section.

2. You can't copy and paste data between non-Windows applications and Windows for Workgroups.

Usually this is because the grabber files are incompatible. You might encounter grabber problems in an incomplete installation of a third-party display driver. Windows might run fine, but non-Windows applications are not displayed correctly. To correctly install third-party display drivers, you must copy the driver and the appropriate grabber files, and you must also update the entries for `286grabber=` or `386grabber=` in the `[boot]` section of `SYSTEM.INI`.

3. You can't start non-Windows applications, or an error message identifies an incompatible display adapter.

The problem is probably that your system does not have a complete set of drivers for that display adapter. Or else that display adapter is not supported by Windows for Workgroups.

If the display adapter can be configured as standard VGA, set the card for that mode, and make sure Setup detects the appropriate display adapter, so that Windows recognizes the display adapter as a standard VGA board.

Also, check with the manufacturer to see if updated Windows 3.1 display driver files (`.DRV`, `.2GR`, `.3GR`, and `VDDx.386` files) are available. These same display drivers will work with Windows for Workgroups.

4. The Windows logo appears in 386 enhanced mode, then disappears and leaves the screen blank.

A few display adapters use nonmaskable interrupts (NMI) to switch video modes to support higher resolutions and to switch modes on the fly. Display drivers shipped with Windows for Workgroups do not support this type of dynamic mode switching, and Windows will not update the display.

A typical scenario with NMI display adapters is that Windows for Workgroups runs correctly in standard mode, but when you try to run Windows for Workgroups in 386 enhanced mode, the logo screen appears briefly, then scrolls off the bottom of the screen, leaving the screen blank. Windows for Workgroups is actually running, but the screen is not being updated. (At this point, you can usually press ALT+F4 and ENTER to quit Windows for Workgroups and return to the command prompt.)

To solve the problem, check the manual for the display adapter to find out how to turn off the NMI or extended modes for the display adapter. For example, some ATI Technologies display adapters use NMI support, but ATI Technologies also provides Windows display drivers that handle NMI correctly.

5. An incorrect-system-version error message appears when you try to run an MS-DOS-based application in 386 enhanced mode.

The VDDx.386 and .3GR file versions are probably mismatched. If the Virtual Display Device and 386 grabber have mismatched versions, edit SYSTEM.INI to make sure that one of the following configurations is installed:

- These are the SYSTEM.INI entries for Windows for Workgroups 3.1 VGA and Super VGA:

```
[boot]                                [386enh]
386grabber=vga.3gr                    display=*vddvga
```

The "*" in the **display=** entry indicates that the information is built into the WIN386.EXE file, rather than existing in a separate file.

- These are the SYSTEM.INI entries for some third-party and OEM display adapters that require Windows 3.0 versions of VDD and 386 grabber files:

```
[boot]                                [386enh]
386grabber=vga30.3gr                  display=vddvga30.386
```

Some OEM display drivers for Windows 3.1 that are shipped with display adapters also include VDD files, 386 grabber files, or both. Be sure that the correct files are installed for the display adapter and that SYSTEM.INI is updated. If the OEM driver disks include a 386 grabber file, but SYSTEM.INI contains the entry `display=*vddvga` in the **[386enh]** section, you must change this entry to `display=vddvga30.386`. Also, the VDDVGA30.386 file must be copied and, if necessary, expanded into the Windows SYSTEM subdirectory from the Windows for Workgroups Setup disks.

6. Your system has a VGA display adapter, and you experience problems running Window for Workgroups in 386 enhanced mode.

Some VGA-compatible cards such as Video 7 and Paradise use additional memory to enhance the performance of their cards. Display adapter boards with the main chips manufactured by Tseng Labs, Chips and Technologies, Paradise, and Headland Technologies also work this way.

Windows detects most of these cards and automatically excludes the additional memory. However, you must add an `emmexclude=C400-C7FF` entry in the **[386enh]** section of SYSTEM.INI if:

- Your system has both an 8514 and a VGA display adapter but you configured Windows for Workgroups for the 8514.
- Your system has an enhanced VGA adapter that Windows for Workgroups does not recognize.
- Your system has both a VGA display adapter (primary) and an 8514 (secondary), and you have problems switching between full-screen non-Windows applications when running in 386 enhanced mode.

For more information about other problems that can be solved with entries for `emmexclude=`, see “Troubleshooting Windows for Workgroups Running in 386 Enhanced Mode” later in this chapter.

7. Your system has a VGA adapter, and an application does not display properly.

The digital-to-analog converter (DAC) on the VGA card might need to be updated. Contact the display adapter manufacturer for an upgrade. In the meantime, if your application has a command to refresh the screen, you might be able to temporarily correct the display so that you can use the application. For example, if the display is a color display, you can type `mode co80` at the command prompt and press ENTER.

See the application’s documentation for information about commands that refresh the screen.

8. Your system has a SuperVGA adapter, and you experience display problems.

If you do not update your display driver to a Windows 3.1 driver, Windows for Workgroups might not work properly when running in 386 enhanced mode. Update your Windows for Workgroups system configuration to use the Windows for Workgroups 3.1 SuperVGA driver if you have a video card and multifrequency monitor with 800x600 16-color resolution but are currently using a third-party Super driver or one of these Windows 3.0 Supplemental Drivers:

- ATI Mode 54h (800x600 16-color) version 2.00
- CHIPS Super VGA 82C451 (800x600 16-color)
- CHIPS Super VGA 82C452 (800x600 16-color)
- Paradise VGA (800x600 16-color)

The Windows for Workgroups 3.1 SuperVGA driver also contains features and performance enhancements, including the ability to obtain over 50 percent more screen space when running Windows for Workgroups.

9. Your system has a TIGA or DGIS display adapter, and you want improved performance.

If you are using a Texas Instruments® Graphics Architecture (TIGA) or Direct Graphics Interface Standard (DGIS) display adapter, use the Windows for Workgroups 3.1 drivers for this display. Windows provides two versions of the TIGA display driver: “TIGA (Small Fonts)” and “TIGA (Large Fonts).”

We recommend that you use the Large Fonts version if you use the display at resolutions of 1024x768 and above. If you are using a 16-inch or larger monitor and want to fit more information on screen, use the Small Fonts version. To change the resolution, you must quit Windows for Workgroups and use the TIGAMODE utility or another resolution utility supplied with your display adapter. Unless you want to use a different version of the TIGA driver (Small Font or Large Font) to change font sizes, you don't have to run Windows for Workgroups setup after changing your TIGA screen resolution.

If you use a TIGA-based display adapter and run Windows for Workgroups in 386 enhanced mode, you can increase the amount of conventional memory available to non-Windows applications by loading the TIGACD driver from WINSTART.BAT instead of from the AUTOEXEC.BAT file. This ensures that the TIGACD driver will load only when Windows for Workgroups is running and will not occupy conventional memory space in 386 enhanced mode.

We do not recommend loading the TIGACD driver from WINSTART.BAT if you run TIGA applications outside of the Windows environment. Also, to run Windows for Workgroups in standard mode on a TIGA display adapter, you must load the TIGACD driver into memory before starting Windows for Workgroups.

10. You experience specific display problems with the PS/2, Toshiba plasma display, Video Seven 256-color adapter, or other hardware.

See the README.WRI file in the WINDOWS directory. The README file contains up-to-date information about problems with display adapters for specific hardware.

Troubleshooting MS-DOS Configuration Problems

This section describes some problems related to the upper memory area and other MS-DOS issues. For information about problems with expanded memory, see “Troubleshooting EMS Memory” later in this section.

1. Your device drivers or programs won’t load in the high memory area.

You should be able to run programs in the upper memory area if all the following conditions are met:

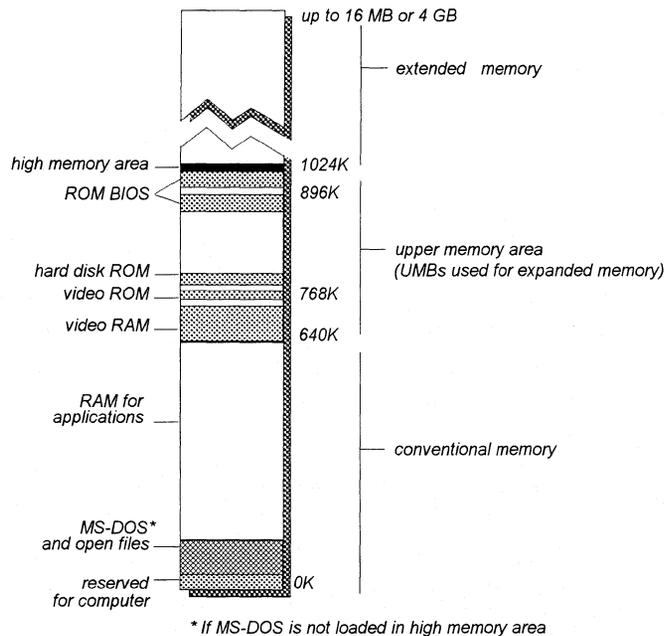
- The computer has an 80386 or higher processor.
- The computer has at least 350K of extended memory available.
- CONFIG.SYS contains a **device** command for HIMEM.SYS before any other **device** or **devicehigh** commands.
- CONFIG.SYS contains either a **dos=umb** or **dos=high,umb** command.
- CONFIG.SYS contains a **device** command for EMM386.EXE, which must include either the **noems** switch or the **ram** switch. This command must appear after the **device** command for HIMEM.SYS but before any **devicehigh** commands.
- CONFIG.SYS contains a **devicehigh** command for each device driver you want to load into the UMBs.
- AUTOEXEC.BAT contains a **loadhigh** command for each memory-resident program (TSR) you want to run in the UMBs.

If you have set up the system to run drivers and programs in the UMBs, but when you use the **mem /c** command, nothing appears in the upper memory section of the **mem** command output, you might be loading EMM386 with the **ram** switch in CONFIG.SYS, but your system might not have enough space in the UMBs to load programs or drivers.

For more information about EMM386.EXE and its startup parameters, see pages 605–609 of the *Microsoft MS-DOS 5.0 User's Guide and Reference*.

Figure 14.1

PC memory map



2. Your computer stops only when you load a specific driver or memory-resident program (TSR) with the devicehigh or loadhigh command.

You might not be able to load that driver or TSR into a UMB. MS-DOS 5.0 device drivers and memory-resident programs should load into the UMBs without any problems (except HIMEM, EMM386, and SMARTDrive must be loaded in conventional memory).

Make sure the device drivers work by replacing the **devicehigh** command with the **device** command, so the device is loaded into conventional memory. You might also have to remove a size switch. If loading the device driver with the **devicehigh** command was the problem, your computer will now work properly.

3. You get a cannot-find-file error message, and you are running the MS-DOS APPEND utility.

When Windows opens a file such as WINWORD.EXE, it records the complete drive and path of this file, so that the file can be reopened later without regard to the current drive and directory. For example, Windows might record a path as C:\WINWORD\WINWORD.EXE.

The MS-DOS APPEND utility prevents Windows from reliably determining the proper path of files. For example, if C:\EXCEL is the current drive and directory when a file is opened, and D:\WINWORD is on the APPEND search path, APPEND might cause Windows to mistakenly build the filename as C:\EXCEL\WINWORD.EXE.

Later, when Windows tries to open this file again, an error occurs because the file is not in the drive and directory indicated. This error results in the message:

```
Change Disk
Cannot find WINWORD.EXE, Please insert in drive A:
```

Windows for Workgroups Setup will warn you if APPEND is found in your configuration and ask you to remove it before Setup proceeds. But Windows will not warn you about its presence if you install APPEND after Setup. APPEND is usually found in AUTOEXEC.BAT, and it should be removed or commented out.

Troubleshooting EMS Memory Problems

This section discusses solutions for several problems that can occur if you use EMM386.EXE with Windows for Workgroups 3.1.

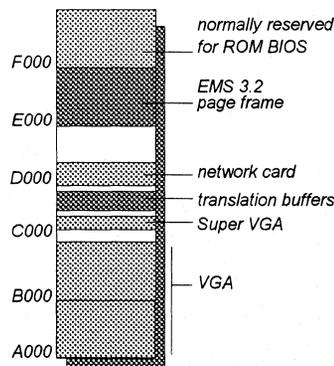
For more information about EMM386 and its startup parameters, see pages 605–609 of the *Microsoft MS-DOS 5.0 User's Guide and Reference*. For more information about how to use program information files (PIFs) to provide expanded memory to non-Windows applications, see Chapter 8, "MS-DOS Applications," in the *Windows for Workgroups User's Guide*.

Tip To see a list of the switches for controlling EMM386.EXE after it has been loaded, type `emm386 /?` at the command prompt.

Figure 14.2

Example of UMBS in the upper memory area

For a VGA or V7VGA display adapter, a network card at D000-D7FF, and an EMS 3.2 page frame at E000-EFFF



1. Network or other adapter conflict with EMM386 UMB space.

If your system has a network adapter, other card or driver that might conflict with EMM386, check its documentation to see what address space the card or driver uses. If the conflict occurs over a network adapter, check with the network adapter documentation to find which memory addresses are being used. Then use the `x=` switch to prevent EMM386 from using those addresses. For example, if your system's network card uses the D800h-DFFFh memory addresses, the command line that loads EMM386 in CONFIG.SYS should look like this:

```
device=c:\windows\emm386.exe noems x=d800-dfff
```

You can experiment with the command that loads EMM386 to determine which memory addresses are being used by the other programs or devices. For details about the command-line switches that can be used with EMM386, see Chapter 11, "Managing Memory and Performance," in the *Windows for Workgroups User's Guide*.

2. You get an invalid-path-for-EMM386 error message when you start Window for Workgroups in 386 enhanced mode.

If this happens, add the `/y=path` switch to the command line that loads EMM386 in CONFIG.SYS. This switch specifies the location of EMM386 when you start Windows for Workgroups. For example, if EMM386.EXE is in the C:\DOS directory, the command in CONFIG.SYS would be:

```
device=emm386.exe /y=c:\windows\emm386.exe
```

Make sure you specified the correct directory in CONFIG.SYS for the HIMEM.SYS and EMM386.EXE files. Also, make sure HIMEM and EMM386 are functioning properly. To do this, make a test boot disk with a CONFIG.SYS file that contains only the `device` commands for loading HIMEM and EMM386. The command that loads EMM386 should be:

```
device=c:\windows\emm386.exe noems x=a000-ffff
```

And include a test AUTOEXEC.BAT file that contains the following:

```
path=c:\dos
prompt $p$g
set TEMP=C:\
```

If the computer stopped when you used the test disk to boot the system, you might have a problem with HIMEM or the hardware.

If the computer functions properly when you boot with this test disk, HIMEM works with EMM386.

3. Some other programs conflict with EMM386.

If you use a third-party disk-caching program other than SMARTDrive, it might not be working properly with EMM386. Remove that program temporarily and put a command in your AUTOEXEC.BAT file to load SMARTDRV.EXE. Contact the manufacturer of the third-party disk-caching program for more information about using it with EMM386.

4. Another program uses the same UMB space as EMM386.

If you suspect a memory-resident program (TSR) is causing the problem, edit the AUTOEXEC.BAT file to remove the **loadhigh** command so the program loads into conventional memory. If loading the program with the **loadhigh** command was the problem, the computer should now work properly.

5. Microsoft Windows for Workgroups 3.1 does not provide expanded memory after EMM386 is installed.

The symptoms of this problem vary. Common symptoms include:

- When you start an MS-DOS-based application, the application displays an out-of-memory message and stops running, or it behaves erratically and displays out-of-memory messages when you try to perform certain tasks.
- Windows for Workgroups 3.1 stops when you start some non-Windows applications, but you have no problem starting other non-Windows applications.

To solve the problem, change the **device** command line for EMM386 so that it includes the **ram** switch instead of the **noems** switch. If the application runs normally after you restart the computer, the **noems** switch caused the problem.

After you add the **ram** switch, other programs or drivers might not load into UMBs. (Use the **mem /c** command to find out.)

If your programs and device drivers still do not load into UMBs, the system might not have enough UMBs available to run those programs. See the computer's hardware documentation to find out which addresses are used.

Troubleshooting Windows for Workgroups Running in Standard Mode

If you have successfully installed Windows, then using the `win /s` switch should force Windows for Workgroups to start in standard mode, provided that:

- The computer has an 80286 or higher processor.
- With at least 2MB RAM memory
- An XMS driver such as HIMEM.SYS is already loaded.

If you cannot start Windows for Workgroups in standard mode, the problem is usually related to the XMS driver or to not having enough extended memory.

1. **RAMDrive or other resident software may be using extended memory and preventing Windows for Workgroups from accessing that memory.**

Decrease the size of the RAM drive or do not load the memory-resident software.

2. **The XMS driver is out of date or incompatible with Windows for Workgroups 3.1.**

Replace the driver with the version of HIMEM.SYS in the Windows for Workgroups package.

3. **HIMEM.SYS may be incorrectly identifying the machine type.**

HIMEM uses an algorithm to detect the machine type, then uses the appropriate routine to access extended memory. When HIMEM loads at boot time, it reports which A20 handler is installed. A20 is a technical reference to the line on the processor that allows access to extended memory. During boot time, the display should read:

Installed A20 handler number x.

The value for *x* identifies the routine used to access the A20 line.

Certain machine types cause HIMEM to mistakenly identify which A20 handler to use. To force HIMEM to use the correct driver, you must use the `/m` switch in the `device=himem.sys` line in CONFIG.SYS, where **m** identifies the type of computer as either the number or name from the following table.

Number	Name	Computer type
1	at	IBM AT or 100% compatible
2	ps2	IBM PS/2
3	ptlcascade	Phoenix Cascade BIOS
4	hpvectra	HP Vectra (A and A+)
5	att6300plus	AT&T 6300 Plus
6	acer1100	Acer 1100
7	toshiba	Toshiba 1600 and 1200XE
8	wyse	Wyse 12.5 Mhz 286
9	tulip	Tulip® SX
10	zenith	Zenith ZBIOS
11	at1	IBM PC/AT
12	at2	IBM PC/AT (alternative delay)
12	css	CSS Labs
13	at3	IBM PC/AT (alternative delay)
13	philips	Philips®
14	fasthp	HP Vectra
15	ibm7552	IBM 7552 Industrial Computer
16	bullmicral	Bull Micral 60

For example, either of the following lines can be used to tell HIMEM to use the IBM AT method of accessing extended memory:

```
device=c:\dos\himem.sys /m:at or device=c:\dos\himem.sys /m:1
```

The table above lists values for some of the most popular hardware. Contact your hardware manufacturer for specific values if your system does not match values in the table.

For more details about using HIMEM.SYS with Windows for Workgroups 3.1, see Chapter 11, "Managing Memory and Performance," in the *Windows for Workgroups User's Guide*.

4. Your system doesn't have enough extended memory.

To make more extended memory available:

- Make sure your computer has as much physical extended memory as the program needs. If the computer does not have enough physical extended memory, you might want to contact your hardware vendor to determine how to add more to the system.

- If CONFIG.SYS includes a **device** command for RAMDrive or EMM386, these programs could be using too much extended memory. If so, reduce the amount of memory specified by editing the **device** command for each driver. For information about editing **device** commands in CONFIG.SYS, see pages 283–287 of the *Microsoft MS-DOS 5.0 User's Guide and Reference*.
- Make sure there are no commands in CONFIG.SYS and AUTOEXEC.BAT to start unnecessary programs that use extended memory.

Troubleshooting Windows for Workgroups Running in 386 Enhanced Mode

If you have successfully installed Windows, Windows for Workgroups will start automatically in 386 enhanced mode provided that:

- The computer has an 80386 or higher processor.
- With at least 3 Mb of RAM memory.
- An XMS driver such as HIMEM.SYS is already loaded.

This section describes solutions to problems you might encounter when trying to run Windows for Workgroups in 386 enhanced mode.

1. You can run Windows for Workgroups in standard mode, but not in 386 enhanced mode.

If Windows for Workgroups runs properly only in standard mode, this indicates that there may be a conflict in the Upper Memory Blocks (UMBs). This conflict occurs when Windows for Workgroups does not detect a network adapter, an extended video board, a scanner board, or some other device.

To determine if there is a UMB conflict, try running Windows for Workgroups with the **/d:x** switch, which excludes all of the upper memory area (similar to setting **emmexclude=A000-EFFF** in the **[386enh]** section of SYSTEM.INI).

If Windows for Workgroups runs when you use this switch, there was a conflict with some device in the UMBs. To solve the problem, you need to identify the configuration of boards or devices in the computer. Also, use MSD.EXE to see if either shadow ROM or shadow RAM is enabled in the E000 segment. After you determine which memory addresses the devices use, add an entry for the **emmexclude=** in SYSTEM.INI to avoid accessing that address range. For example, many high-resolution video boards use the address range above C000 to C7FF. In that case, the correct entry in SYSTEM.INI would be **emmexclude=C000-C7FF**.

Limiting the excluded range to the specific address range is important because Windows for Workgroups uses that memory address range to place the API translation buffers and page frame. If Windows for Workgroups cannot locate enough free memory in the UMBs, it will allocate conventional memory for the translation buffers, taking memory away from non-Windows applications that you want to run. For more information, see “Managing Memory and Performance,” in Chapter 11 *Windows for Workgroups User's Guide*.

Note You can run MSD.EXE or use the MS-DOS `mem /p` command to see what drivers might be using the UMBs and their respective addresses. Also, you can type `win /?` at the command prompt to see a list of switches that you can use to run Windows for Workgroups 3.1.

2. The system crashes when you run Windows for Workgroups in 386 enhanced mode.

When Windows for Workgroups starts in 386 enhanced mode, it installs the API translation buffers in any free UMB area it finds. Windows for Workgroups might also compete for memory with adapter cards that don't sign in at the memory area they occupy. The standard is for adapters to leave a ROM signature 0xAA55 followed by the size of the area that is being used, but not all cards do this. If Windows for Workgroups tries to use an area that appears to be free but that is also being used by an adapter card, the system might crash.

With 386 loadhigh drivers such as Qemm and 386MAX, there can be additional contention problems within the UMBs, because the loadhigh drivers might already be using that area. With conventional memory at a premium on most computers, you must consider how much memory to allocate to the loadhigh drivers and which areas to leave open for the translation buffers.

To keep the translation buffers from competing for memory, add the entry `emmexclude=A000-EFFF` to the `[386enh]` section of SYSTEM.INI. This entry tells Windows for Workgroups not to position the translation buffers in the UMBs. Windows for Workgroups will allocate the buffers in conventional memory.

If Windows for Workgroups 386 enhanced mode crashed before implementing this switch, and now it doesn't crash, the next step is to isolate the specific area causing Windows for Workgroups to crash:

- If your system has a nonstandard video board, you can begin by excluding the range C600-C800, which some display adapters use without warning. If your system uses shadow RAM or has copied

either system BIOS or video BIOS into the UMBs, then you should turn the feature off, or exclude the area that is being used (usually `emmexclude=E000-EFFF`).

- If the display adapter supports extended modes (like SuperVGA), also try turning off the extended mode support, which is typically a nonmaskable interrupt (NMI). For details, see item #4 in “Troubleshooting Display Problems” earlier in this chapter.

3. You experience hard disk problems when running Windows for Workgroups in 386 enhanced mode.

Windows 386 enhanced mode, without SMARTDrive loaded, supports only standard ST506 and ESDI controllers. To use 386 enhanced mode with SCSI or other nonstandard devices, you must load SMARTDrive. When Windows detects SMARTDrive, it directs all disk access requests through SMARTDrive, so that SMARTDrive can function effectively as a double-buffering hard disk device driver, which eliminates most compatibility problems.

But even if SMARTDrive is loaded, in some configurations you might experience hard disk access errors, such as the following:

Cannot Read from Drive C:
Cannot Write to Drive C:
Drive C: Not Ready Error

If this happens, add the entry `VirtualHDirq=off` in the `[386enh]` section of SYSTEM.INI. Also, you must have SMARTDrive installed without the `/B-` switch. For example, add the `VirtualHDirq=off` entry to SYSTEM.INI if you use a Plus Hardcard or a SCSI hard disk drive using DMA.

This is the highest level of hard disk compatibility that can be achieved under Windows for Workgroups 3.1. If you still experience hard disk problems, contact the disk manufacturer for further assistance.

4. You get an internal-stack-overflow message while running Windows for Workgroups in 386 enhanced mode.

Stacks are temporary data structures that MS-DOS and applications use for processing hardware events. If you get a stack-overflow error:

- And if your system uses MS-DOS 3.2, include this line in CONFIG.SYS:
`stacks=9,192`
- And if your system uses MS-DOS 3.3 or above, include this line in CONFIG.SYS:
`stacks=9,256`

5. Typing slows down when you run more than one application, but your system seems to have plenty of memory.

The timeslice for multitasking is taking processing time away from the foreground application. You can increase the foreground priority in the 386 Enhanced dialog box in Control Panel or in the application's PIF. Better still, you can increase the priority of the foreground application only when it receives a keystroke by adding the entry **KeyBoostTime=.005** (or some larger increment) in the **[386enh]** section of SYSTEM.INI.

Troubleshooting for MS-DOS-Based Applications

This section describes some solutions to problems that can occur while running non-Windows applications. See also the section "Troubleshooting Windows for Workgroups Configuration" in this chapter, because it discusses many problems that also relate to non-Windows applications.

1. You receive an "Unexpected MS-DOS Error #11" message.

If you receive this error message when you try to run an MS-DOS-based application from Windows for Workgroups 3.1, it is a Kernel internal error that indicates Windows for Workgroups has tried to execute a file that has an invalid format.

If this happens in standard mode, then either the WINOLDAP.MOD file or the 286 grabber is corrupted; if in 386 enhanced mode, then either the 386 grabber or WINOA386.MOD file is corrupted.

To solve this problem, copy the grabber file plus WINOLDAP.MOD or WINOA386.MOD file from the source disks, and expand these files, using EXPAND.EXE (which can be found on your Windows for Workgroups 3.1 installation disks or in your MS-DOS directory).

2. You experience problems running non-Windows applications in standard mode.

- If the application doesn't run in the startup directory specified in the application's PIF, check the Working Directory setting for the program item in Program Manager. This setting overrides PIF settings.
- If data is garbled or lost, check the Directly Modifies options in PIF Editor to ensure that the application is controlling the keyboard and any COM ports it uses. Also, make sure No Save Screen is not checked.
- If you can't paste screen captures, make sure No Screen Exchange is not checked in PIF Editor.
- If you can't switch back to Windows, make sure Prevent Program Switch isn't checked, and that Keyboard is not checked in the Directly Modifies options in PIF Editor.

3. You experience problems running non-Windows applications in 386 enhanced mode.

- If your system runs out of memory when you are running a high-resolution graphics application, check High Graphics in the PIF's Video Memory option and check Retain Video Memory.

Conversely, to free as much memory as possible if your system has a Hercules or CGA adapter, check Low Graphics and make sure Retain Video Memory is not checked.

- If the mouse doesn't work when you run the application in full-screen mode, make sure that an MS-DOS mouse driver is installed. See also Flowchart 2.8 for specific troubleshooting procedures.

Also, try choosing Full Screen option if your system is running out of memory.

- If the application doesn't display properly, make sure the Monitor Ports options are checked. If text is garbled or the cursor is in the wrong place, make sure Emulate Text Mode is not checked.
- If you can't paste into an MS-DOS-based application, make sure Allow Fast Paste is not checked in the PIF Advanced Options dialog box.

4. You experience performance problems when running some applications at the same time, but each application runs okay alone.

You might have a device driver or memory-resident application that requires expanded memory or extended memory, but EMS or XMS memory is locked in the PIF for another non-Windows application. Examine the PIFs of all applications that run simultaneously, and clear the EMS Memory Locked and XMS Memory Locked options if you are trying to run applications simultaneously.

5. Lotus 1-2-3 version 3.1 won't start.

If Lotus 1-2-3 no longer runs properly, or it prevents you from opening large files, there could be several causes:

- Your system might not have enough XMS memory available. You need at least 384K. For information about how to make more extended memory available, see "Troubleshooting Windows for Workgroups Running in Standard Mode" earlier in this chapter.
- The PIF for Lotus 1-2-3 has been modified so that it no longer specifies enough extended memory. Specify that 384K of XMS memory is needed in the PIF. For more information, see Chapter 8, "PIFs and PIF Editor," in the *Windows Resource Kit*.

- EMM386 isn't set up to work with Lotus 1-2-3. For example, you might have loaded EMM386 with the **noems** switch.

Because Lotus 1-2-3 version 3.1 and EMM386 both run in protected mode, they need to communicate to avoid conflicts. Lotus and EMM386 can use the DPMI protocol to communicate. DPMI assumes that a page of expanded memory exists. A page is not available if EMM386 uses the **noems** switch. You must change the CONFIG.SYS setting—for example:

```
device=c:\windows\emm386.exe ram p0=d000
```

The **ram** switch gives programs access to both expanded memory and the UMBs. The **p0=d000** setting specifies a page for EMS.

When you restart the computer after making this change, you'll be warned that a LIM 3.2-compatible page frame does not exist. The standard size for a page frame is 64K, but since Lotus 1-2-3 is using XMS memory (not EMS), it only needs a 16K page. This provides more space for programs in UMBs.

6. You experience problems running a specific application or network application.

Check the README.WRI or NETWORKS.WRI file for up-to-date notes on specific applications.

7. You receive error messages such as general protection faults, application execution errors, or system integrity violations.

See Flowcharts 5.3, 5.4, and 5.5 for specific troubleshooting procedures.

Troubleshooting Fonts

1. You experience problems with your system when you choose TrueType fonts.

Some display drivers that cache fonts in unused video memory have problems displaying TrueType fonts. If you experience problems, contact your display adapter manufacturer for updated display drivers.

2. You use a third-party printer driver, and you cannot print TrueType fonts.

Most printer drivers created for Windows 3.0 do not support TrueType fonts. Use one of the printer drivers in your Windows for Workgroups 3.1 package, or contact your vendor for an updated driver.

3. You are using a font converter, and you experience system problems when you use TrueType fonts.

Some font converters are not completely compatible with TrueType, causing problems in code segments while running. If you experience system crashes or other problems, do not use that font converter. Contact the manufacturer for an updated version.

4. You have a minimum configuration for running Windows, and you experience problems when you use TrueType fonts.

You might have problems if you attempt to use TrueType on a system with less than 2MB of memory. If you have such a system, to avoid low-memory situations, choose vector fonts or bitmap fonts, and avoid TrueType.

5. You use Bitstream® Facelift™ and have a high-resolution display, and you experience problems when you use TrueType fonts.

Facelift is incompatible with some Windows for Workgroups 3.1 display drivers. Contact Bitstream for software updates, or disable Facelift when you want to use TrueType. Or use your old Windows 3.0 display drivers with Facelift.

6. You have a type manager installed in your system, but you want to remove it.

See Flowcharts 3.2, 3.3, and 3.4 for instructions on disabling Adobe Type Manager®, Bitstream Facelift, and HP Intellifont.

Troubleshooting Printing

Many of the printing problems reported to Microsoft Product Support Services are actually hardware or configuration problems not related to Windows. This section provides tips for general troubleshooting for printing, then provides solutions to some specific problems related to printing in Windows for Workgroups.

A Strategy for Troubleshooting Printing

If you have problems printing, begin with these steps to isolate a minimal number of variables. By eliminating the variables, you can solve the problem.

- **Read the PRINTERS.WRI file in the WINDOWS directory.**

The PRINTERS.WRI file contains up-to-date information about printing from Windows with specific hardware.

- **Make sure you can print from MS-DOS.**

Copy a file to the printer port to make sure the communication path is working correctly. To do this, at the command prompt, type **dir > lptx** or **dir > comx** (where *x* is the number for the port that the printer is connected to).

This method will not work for PostScript printers. To check a PostScript printer, copy TESTPS.TXT to the printer from your Windows SYSTEM subdirectory.

If you cannot print from MS-DOS outside of Windows, it's not a Windows printing problem. Contact your printer manufacturer or distributor.

- **Check the printer's cable connections.**

Make sure the printer is turned on and that it is online. Check that it has paper and toner.

- **Use a SET TEMP statement in AUTOEXEC.BAT.**

Make sure you include a **SET TEMP** statement in AUTOEXEC.BAT and that it points to an existing directory. For testing purposes, we recommend that you use the statement **SET TEMP=C:**. Verify that enough space is free on the hard disk for spooling temporary files while printing. If the TEMP environment variable points to a network drive, you must have create and write access to the target directory.

- **Make sure you are using the correct printer driver for the printer.**

The current printer driver can be seen by choosing the Printers icon in Control Panel. If the highlighted printer name doesn't match the printer that you are using, then you need to install a printer driver that does match the printer, as described in Chapter 5, "Print Manager," in the *Windows for Workgroups User's Guide*.

- **If you print to a COM port, you might need to configure the COM port with the MS-DOS Mode command or with Control Panel settings.**

See the MS-DOS *User's Guide* for information about the **mode** command. In Windows, choose the Ports icon in Control Panel to configure the COM port. Make sure that the printer is selected, and then try to print. If there is a program that redirects or reconfigures the printer at the MS-DOS level, you should print to LPTx.DOS (where *x* is the port number), which tells Windows to let the operating system handle the printing.

- **Check the communication settings for serial printers.**

If you have a serial printer, make sure the communications settings are correct. The settings on the printer and the Windows for Workgroups 3.1 settings should be the same. In Windows for Workgroups, the communications settings are set in Control Panel. Almost all serial printers, including Hewlett-Packard plotters and most PostScript printers, use a setting of **9600 baud, no parity, 8 bits, 1 stop bit, and hardware handshaking**.

To make sure that the printer has the same settings as Windows for Workgroups, use the printer's interactive front panel or DIP switches.

Handshaking for PostScript printers can be set in the Options section of the Windows Printer Setup dialog box. The PostScript driver will send a short program to the printer telling it to switch to the desired handshaking mode.

- **If you are not printing on a Windows for Workgroups computer.**

You might need to use LPTx.DOS if you are printing to a network-redirected port that is also present in hardware. For example, your system might have a physical LPT1 port in the computer and also have a virtual LPT1 on the network. This is may only be necessary if your system is *not* printing to another users Windows for Workgroups network printer.

For more information, see "Troubleshooting Networks" later in this chapter.

Tips for Specific Printing Problems

The following items suggest solutions to specific printing problems.

1. **You get a cannot-print error message.**

This error message can occur on a network or on a physical parallel port. If you are not printing to another Windows for Workgroups computer you might have to use the LPTx.DOS method.

Windows for Workgroups 3.1 requires that the Centronics Paper Out signal be Off and that the Select In signal be On. If one of these conditions is not met, Windows gives a cannot-print error message when you try to print. The possible causes of such a problem include:

- The printer is offline, out of paper, or jammed.
- The printer has faulty cabling.

- The printer uses an incompatible print-sharing device, switchbox, or buffer.

For testing purposes, remove all such devices and connect the printer directly to the computer's parallel port with a single cable. If the problem persists, try a different cable. If you use a switchbox to switch printers electronically, test printing with the switchbox removed. If you are printing over a network, print locally (directly connected to the computer) to see if the network is part of the problem.

Check the TEMP variable in your AUTOEXEC.BAT file. Make sure you haven't run out of disk space.

2. You get a printer error message in the middle of printing a document.

You can set the Windows printer time-outs for each individual printer in the Printers section of the Control Panel. The time-out options are Device Not Selected and Transmission Retry. The Device Not Selected option indicates how long Windows for Workgroups will wait for the printer to be online before it gives an error. The Transmission Retry option sets the time that the printer will stay busy before Windows for Workgroups gives an error.

Try increasing the Transmission Retry setting. The Transmission Retry time-out starts counting as each character is sent to the printer. If the time-out lapses before the printer is ready to accept another character, you'll get a printer error message.

3. Your PostScript print jobs are incomplete or missing.

- Check the communication settings and the cable.
- Check that the printer has enough memory to handle the print job.

If the problem is intermittent, or if it only happens with a particular application, download the PostScript Error Handler to help debug the problem.

To get a PostScript error report

1. Choose the Printer icon in Control Panel, then choose the Options button in the Printer Setup dialog box.
2. Choose the Advanced button and, in the Advanced Options dialog box, check the Print PostScript Error Information option. Specify any other options you want in the dialog box, then choose the OK button.
3. Repeat the procedure that resulted in the problem.

If you see a PostScript error message, the problem is being caused by your application or by the degree of compatibility of your PostScript printer.

If you don't see a PostScript error message, recheck the other solutions discussed in the previous section.

Troubleshooting Networks

Problems may arise from the addition of network hardware in your system. The Windows for Workgroups Setup program attempts to detect the correct network adapter installed in your computer and its related configuration settings. Troubleshooting may be necessary if any of the settings are configured incorrectly or conflict with existing hardware.

You might also experience problems when running network specific applications with Windows for Workgroups. Make sure you check the NETWORKS.WRI file in the WINDOWS directory for up-to-date information about running Windows for Workgroups with other network applications.

- Software conflicts can often be the cause of network errors as well, so when isolating problems it is best to work with a clean boot configuration. For information on creating a clean boot see "Creating a 'Clean Boot' for Troubleshooting" earlier in this chapter.

Network Adapter Troubleshooting

Most network-related problems are related to the system configuration. Conflicts with network hardware can commonly occur in three areas:

- Interrupt conflicts with devices or drivers
- Memory port addresses (I/O addresses)
- Shared-memory addresses

1. You suspect interrupt conflicts with either device drivers or hardware.

If you suspect interrupt conflicts, refer to the list of IRQs in Troubleshooting Flowchart 2.7. MSD.EXE, installed in your WINDOWS directory by Setup, can be used from the command prompt to see what devices or boards are using specific IRQs.

2. You suspect I/O port address conflicts.

If you suspect that the memory port addresses are conflicting, determine the I/O port addresses for the other add-on boards in the system. Then select an address that does not conflict.

Typical I/O port addresses allowed on network adapters do not conflict with the computer. However, the addresses specified for other add-on boards often conflict. On some computers, the CMOS Setup program identifies the I/O ports that the communications devices and other standard devices. Make sure the I/O port of the network adapter does not conflict with these.

3. You suspect shared-memory address conflicts.

The shared-memory addresses are located in the UMBs and can cause subtle problems when video cards or other peripherals are attempting to share the same UMB area. For example many ESDI and SCSI hard drive controllers use shared RAM. If the network adapter is installed at the same location, network or data corruption can occur on the hard drive or network.

Some network adapters require a `x=` option on the EMM386 command line in your CONFIG.SYS file. These adapters may also require the use of the **ExcludeHighRegion=** or **EMMExclude=** entry in the **[386Enh]** section of the SYSTEM.INI file to tell Windows for Workgroups to avoid certain memory ranges. For more information, see Chapter 11, "Managing Memory and Performance," in the *Microsoft Windows for Workgroups User's Guide*. If you are using 386Max, QEMM, or another memory manager, see your memory-manager documentation to find out whether these programs may also require you to exclude portions of the UMA.

Also, shadow RAM and shadow ROM can cause memory address conflicts, so you might want to investigate the CMOS Setup to check for shadowing configuration options.

Network Communication Troubleshooting

Common network communication problems are related to either a network adapter configuration problem or device failure, or problems with the network cabling. A quick test to help isolate a problem to a workstation is to attempt the same operation on another computer. If you are able to view and connect to resources from another workstation, but not from your computer, this will help to isolate the problem to your computer rather than to the network itself.

Network Adapter Drivers and Configuration

If your Windows for Workgroups workstation can no longer see or connect to other computers on the network, you must first start by confirming the network drivers are loading properly in the CONFIG.SYS and AUTOEXEC.BAT files. Faster booting computers make this difficult, as many error messages don't stop a PC from booting and may not even be seen. One solution is to pause the boot process.

- To pause place two MS-DOS **pause** statements in the AUTOEXEC.BAT. One at the beginning, and another after **net start**.

Pause the AUTOEXEC.BAT by placing pause statements here:

```
pause
c:\windows\net start
pause
path=c:\;c:\dos;c:\windows
set temp=c:\windows\temp
prompt $p$g
```

Novell NetWare users will want to place the second pause after the NETX command.

```
pause
c:\windows\net start
c:\windows\msipx
c:\windows\netx
pause
path=c:\;c:\dos;c:\windows
set temp=c:\windows\temp
prompt $p$g
```

Windows for Workgroups drivers are generally the last device lines of a CONFIG.SYS, so the first pause in the AUTOEXEC.BAT will let you to confirm that these drivers did not report any errors. The second pause after **net start** will allow you to confirm that the protocol *binds* to the network driver. Novell users can also confirm that MSIPX and NETX have connected to a NetWare file server.

If any Windows for Workgroups drivers fail to load, you will want to begin isolating the problem by excluding the network adapter from the upper memory area, as stated earlier in this chapter and again in troubleshooting network adapters. Next, confirm your selection for network adapter in Control Panel Network match your physical adapter model, IRQ, I/O and RAM settings.

After restarting the computer an error still occurs, look for hardware conflicts with other cards or peripherals. Often you can remove all these components except for those that are absolutely necessary to boot and communicate with the network.

Network Cabling Problems

If the configuration of your network adapter and network protocol seem to check out, another cause of the inability to see other workstation on the network may be related to the network cabling itself.

If you are using a thin-wire Ethernet network, as used with the *Windows for Workgroups Starter Kit*, here are some general trouble-shooting guidelines that may help you to isolate network cabling problems:

- Ensure that a terminator is attached to each end of the cable segment, and that each terminator has 50 ohms of resistance (± 2 ohms).
- Ensure that all network cable segments are connected together and that there are no breaks. It is also a good idea to check the T connectors for faults, and the cable to verify that it is not shorted.
- Check to make sure the length of the cable segment has not exceeded the maximum cable length supported by the network adapter. The maximum cable segment length for Ethernet is approximately 600 feet (185 meters).
- Check to make sure the length of cable user to interconnect workstations is not less than the minimum length defined by the Ethernet specification. The minimum cable length between workstations is approximately 2 feet (0.5 meters).
- Check to make sure you have not exceeded the maximum number of nodes on the cable segment. The Ethernet specification supports up to 30 nodes on a cable segment.

Troubleshooting Novell NetWare

First, review the NETWORKS.WRI file for any additional information.

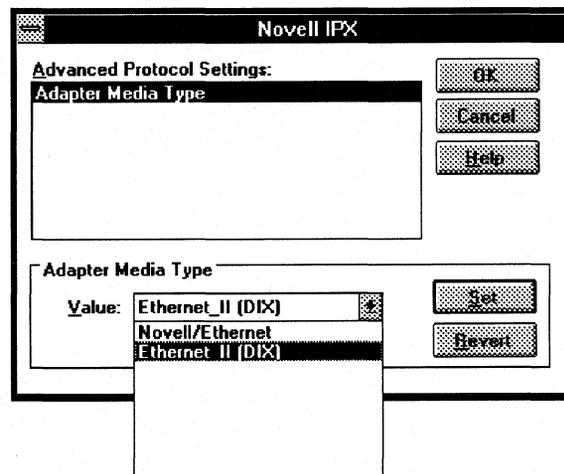
If Novell NetWare is selected as a Compatible Network for Windows for Workgroups and an error message is generated by NETX stating "File Server cannot be found", you may need to adjust one parameter depending on your network topology.

- If you are using a Token-Ring network and the network adapter driver on the Novell server does not support source routing (for example, IBM token-ring drivers prior to version 2.4), remove the ROUTE.COM statement from your AUTOEXEC.BAT file.

- If you receive an error stating that IPX can't find a Novell server on the network, you may need to change the Adapter Media Type option for IPX. If you are using a **Ethernet** network and the network adapter driver on the Novell server uses Ethernet II Frame Type, change the Adapter Media Type as follows:
 1. Go to Control Panel Network icon and select Adapters.
 2. Highlight the adapter in use and select Setup.
 3. Select Protocols and highlight **Novell IPX** under Protocols in Use. If **Novell IPX** is not listed, press Cancel and add the ability to communicate with Novell NetWare servers in the Network option of Control Panel Network.
 4. Select Settings
 5. Highlight **Adapter Media Type** and change the value to **Ethernet_II (DIX)** and select Set. Exit from Windows for Workgroups for Workgroups by pressing OK and restart the computer when prompted.

Figure 14.3

Novell IPX dialog
box showing
Adapter Media
Type options



- Some Novell network shells and drivers cause problems. For example, shells earlier than version 3.01e can cause problems, especially if you use the EMSNET3.COM, EMSNET4.COM, XMSNET3.COM, or XMSNET4.COM shell. These shells have exhibited random problems, but the most significant and reproducible problems are related to printing.
 - Windows for Workgroups 3.1 setup replaces your current NetWare shell with the provided NETX.COM. This version of NETX.COM will run on any MS-DOS 3.0 or higher computer.
 - If you send a job to the printer and get garbage, determine which shell is loaded. Closer inspection of the garbage printout might reveal that the print job started at address 0:0 and printed out the interrupt vector table, the MS-DOS environment, or other factors like TSRs. These clues can help you find the source of the problem.
- When you add the ability to communicate on a Novell network to Windows for Workgroups, your CONFIG.SYS file is modified to include the setting LASTDRIVE=P. Windows for Workgroups can then use MS-DOS drive letters up to P to connect with other Windows for Workgroups PCs. Thus, the Novell default login drive will be Q, although you can map Novell drives over A through P.
 - Many batch file menus, including those that come from Novell have been written by the system administrator to use NetWare drives F through Z. If your menu program has been written this way, confirm the Novell drives are mapped correctly and available before menu execution.
- After Windows for Workgroups is installed, applications that use Novell named pipes DOSNP.EXE to communicate with NetWare servers may no longer work correctly. Applications that use named pipes to communicate with Microsoft LAN Manager servers will work with Windows for Workgroups.

Many network applications that use named pipes can also be configured to use another method of communicating on the network, such as IPX or SPX. Contact your network-application vendor for additional details.

Troubleshooting GP Faults

A General Protection fault (GP fault) occurs when a Windows application writes to a memory space where it doesn't have access. This corrupts any other code that already occupies that memory space. In Windows 3.0, after a GP fault happened, sometimes the memory was so corrupted that even applications that didn't cause the problem could trigger a GP fault message; the system remained unstable so that you had to quit Windows for Workgroups and reboot the system.

Windows for Workgroups 3.1 and Windows-based applications designed especially for Windows 3.1 are significantly more reliable. An internal feature called Parameter Validation checks for invalid parameters when Windows API functions are used by an application program.

If you do see a GP fault message while running Windows for Workgroups 3.1, the message will provide specific information about which application caused the error and where the error occurred. You can probably continue running the application after a GP fault, so that you can save your work before quitting the application.

You can use Dr. Watson, a utility provided with Windows for Workgroups 3.1, to learn more about what is causing a GP fault. For details, see “Dr. Watson and GP Faults” later in this section.

You can also use the Windows for Workgroups 3.1 local reboot by pressing CTRL+ALT+DEL if a Windows application or an MS-DOS-based application is hung or is not responding to the Windows command to close. When you use local reboot, you will see an error message with instructions for closing the application.

Tips for Troubleshooting GP Faults

This section describes a few reasons why a GP fault might occur.

- 1. You are running an application that was designed for an earlier Windows version (2.x) under Windows for Workgroups 3.1.**

If an application was designed for Windows versions earlier than 3.0, it displays a dialog box when you try to start it, describing this error. If the application has not been specifically designed to run under Windows 3.0 or later versions, it will trigger a GP fault.

- 2. You manually selected an incorrect computer during Windows for Workgroups installation, or Setup’s auto-detection misidentified elements of your system.**

When you install Windows for Workgroups 3.1 on some computers, you can’t use Setup’s auto-detection as described in “Troubleshooting Setup” earlier in this chapter. Instead, you have to override the default and specifically make the correct selection, or Windows for Workgroups will not operate properly. If you don’t specifically select these computers, you might get a GP fault.

For details about system selections in Setup, see “Troubleshooting MS-DOS Mode Setup” earlier in this chapter.

3. Your CONFIG.SYS or AUTOEXEC.BAT files contain either incompatible terminate-and-stay-resident programs (TSRs) or unsupported network drivers.

You can test your system for incompatible TSRs or unsupported drivers by temporarily removing them so that the system is at a minimum operable configuration. If doing this solves the GP fault problem, then you've begun to isolate the cause.

To find the specific cause, replace the lines for the TSRs and network drivers in CONFIG.SYS or AUTOEXEC.BAT one by one until the problem reappears. Then you'll know which one is causing the GP fault.

4. Your system has a page-mapping conflict when you run Windows for Workgroups in 386 enhanced mode.

Page-mapping conflicts only occur in Windows for Workgroups 386 enhanced mode. And this usually occurs because Windows for Workgroups doesn't see the page frame and so maps over it when you are using a third-party EMS manager instead of EMM386.EXE.

Test for a page-mapping conflict by running Windows for Workgroups with the **win /s** switch (standard mode) or the **win /3 /d:x** switch (excluding the upper memory area). If the problem doesn't happen, you might be experiencing a page-mapping conflict in the UMBs.

To isolate the problem, start by excluding the entire UMB from mapping with an **emmexclude=A000-EFFF** entry in the **[386enh]** section of SYSTEM.INI, then begin narrowing the range as described in "Troubleshooting Windows for Workgroups Running in 386 Enhanced Mode" earlier in this chapter.

If you have a Micro Channel (MCA) bus computer, such as an IBM PS/2, you might be able to determine the location of hardware adapters by booting with the computer's Reference Disk. If you have a standard ISA computer, consult the documentation for the adapters, or contact the manufacturer's technical support services to find out which memory locations they use.

5. Your system has an incorrect version of MS-DOS.

If your computer is not running the right version of MS-DOS, you might get GP faults. You should only use OEM versions of MS-DOS (such as COMPAQ MS-DOS or IBM PC-DOS) on OEM hardware platforms. For example, only use COMPAQ MS-DOS on COMPAQ computers.

To determine which version of MS-DOS to use, follow this rule:

- If your computer's manufacturer has an OEM version of MS-DOS, use it.
- If your computer's manufacturer doesn't have an OEM version of MS-DOS, use Microsoft MS-DOS and upgrade to version 5.0.

Dr. Watson and GP Faults



Dr. Watson is a Microsoft diagnostic tool that provides detailed information on the internal state of Windows for Workgroups when a system error occurs. If an application error occurs, Dr. Watson automatically creates a special file in your WINDOWS directory named DRWATSON.LOG. Dr. Watson also prompts you for details about the circumstances under which the application error occurred. Sometimes Dr. Watson detects a fault that might not be fatal, and you are given the opportunity to ignore the fault or close the application.

When you choose the Ignore option after an application error is detected, Windows for Workgroups continues without performing the faulting instruction. You might be able to save your work in a new file at this point, but do not continue using the application for other actions. We recommend you quit Windows for Workgroups after saving your file, then restart Windows for Workgroups.

If you ignore a fault, the same dialog box might appear again. If choosing Ignore a few times doesn't allow you to continue operation, choose Close to terminate that application, because that is the only option available.

When you choose Close, the General Protection Fault dialog box appears, with an explanation and some data that may be useful in tracking the source of the error. Save this information. Contact your application vendor to discuss the source of the application error.

Troubleshooting Multimedia

This section provides some tips for troubleshooting the multimedia accessories.

1. Your multimedia features don't work.

Make sure your computer has installed the multimedia extensions to Windows by opening the Control Panel in Program Manager. The multimedia extensions add the following icons: Display, Drivers, Joystick, MIDI Mapper, and Screen Saver. If these icons aren't there, you must run Windows Setup to install the multimedia extensions.

2. Your application won't play audio.

- Make sure you have the speakers plugged into the correct port. With some configurations, it's easy to confuse the microphone port with the speaker port.
- Make sure that speakers are getting the power they need to operate. Some speakers require an external power source or batteries.
- Check the volume setting for your PC audio hardware. You might have it turned all the way down.
- Choose the Drivers setting in the Control Panel to make sure you have the correct audio driver installed and that the settings are configured according to the hardware manufacturer's documentation.

3. Audio is too loud or too soft.

If you can hear audio, but it's too loud or too soft, check the volume control on your PC hardware and make any necessary adjustments. Also check the batteries in your speakers to see if they need to be replaced.

4. You can't hear audio when running an MS-DOS-based application that supports audio.

Several non-Windows applications support audio. If you try to run one of these applications in 386 enhanced mode, the following message might appear: "This application will not be able to use audio." This is because Windows 386 enhanced mode takes over the audio hardware and won't relinquish control to an MS-DOS-based application. You can run the application with audio in two ways:

- Quit Windows for Workgroups and run the application from MS-DOS.
- Quit Windows, and restart Windows for Workgroups in standard mode by typing `win /s` at the command prompt. This will let you run the non-Windows application from Windows for Workgroups with audio intact, but not allow you to act as a server.

5. Not able to share or access a CD-ROM drive.

- A CD-ROM can be shared as if it were any other local drive. Select the File Manager CD-ROM icon and then select the directories you want to share.
- If you have an external drive, make sure the cable is securely attached and connected to the correct port. Also, some drives require an external terminator. Make sure this is installed if needed. See your CD-ROM drive documentation for details.

- Make sure the CD-ROM driver MSCDEX.EXE version 2.21 is installed.

If Windows for Workgroups Setup detects a MSCDEX driver in your AUTOEXEC.BAT it will be upgraded to version 2.21. It will also add the /s to enable CD-ROM drive sharing.

- If running Novell NetWare with Windows for Workgroups, you should load MSCDEX before you load MSIPX, and after Net Start in the AUTOEXEC.BAT.
- Other users accessing the CD-ROM drive across the network do *not* need to load the MSCDEX.EXE driver.

6. Your CD drive can't read the CD disc correctly.

- Make sure you place the disc in its drive caddy before inserting it into the drive.
- Make sure you insert the disc with its label side up.
- If you upgraded your PC with an MPC upgrade kit, make sure you installed and configured the hardware according to the manufacturer's directions.
- Make sure to use the correct kind of disc. CD-ROM discs contain information and multimedia applications; CD audio discs contain music. You can't use File Manager or any other MS-DOS or Windows file commands on an audio disc.
- Even though discs are quite durable, they are plastic and can become damaged. Excessive heat can warp or even melt a disc. Scratches and fingerprints on the surface can cause problems. Clean a disc by wiping with a soft cloth from the center of the disc outward in a straight line. Do not use alcohol-based cleaning solutions.

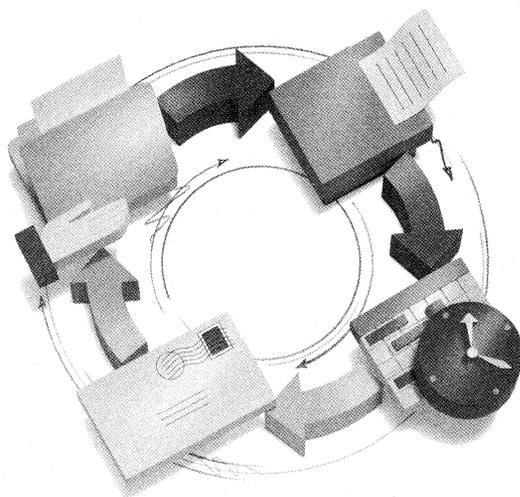
7. Music Box doesn't play correctly.

If Music Box won't play a disc or plays very slowly, check the following:

- Make sure you have a disc in the drive before starting Music Box. Make sure to place the disc in its drive caddy before inserting it into the drive.

- Make sure you're using the right kind of disc. Music Box can't play a disc that doesn't have audio.
 - Make sure you're using the most current version of the CD-ROM driver software (MSCDEX) provided with Windows for Workgroups. If you're using an older version, this can cause Music Box to play too slowly.
- 8. Sound Recorder doesn't record.**
- Make sure the microphone is turned on. And make sure the microphone is plugged into the correct port.
 - Make sure you speak clearly and distinctly into the microphone.
 - Make sure your PC has enough disk space to hold the recorded file.
 - Make sure the output volume level of your audio hardware is set correctly. If you set it too low, you can see the audio recording, but you won't be able to hear it when you play it back.
- 9. The Media Player won't play MIDI files.**
- Make sure the correct MIDI driver is installed. Choose the Drivers setting in the Control Panel to see whether you have a MIDI driver set up and ready for use.
 - Make sure you set Media Player to play MIDI files through the Device menu.
 - Make sure you have the right MIDI mapping setup. If you added MIDI hardware, check the MIDI Mapper settings in Control Panel.
- 10. The Media Player won't play animation.**
- Make sure you set Media Player to play animation files through the Device menu.
 - Make sure the animation driver is installed. Choose the Drivers setting in Control Panel to see whether you have the Microsoft MCI Driver for MMP set up and ready for use.
- 11. The Media Player won't play audio files.**
- Make sure the correct audio driver is installed. Choose the Drivers setting in Control Panel to see whether you have the correct audio driver set up and ready for use.
 - Make sure you set Media Player to play audio files through the Device menu.
 - Make sure the file you want to play is a .WAV file.

References, Resources, and Appendixes



Glossary

1

Appendix A

Windows for Workgroups Resource Directory

A-1

Microsoft Support Services	A-2
Microsoft Product Support Services Worldwide	A-3
Microsoft Consulting Services	A-9
Microsoft Solutions Channels—Consulting Program	A-9
Microsoft Solutions Channels—Training Program	A-10
Microsoft Certified Professional Program	A-10
Microsoft University/Technical Training Straight from the Source	A-12
Online Services.....	A-19
Microsoft Developer Network	A-20

Appendix B

Microsoft Mail Configuration Guide

B-1

Introduction	B-3
How to Use This Appendix	B-7
Master Connectivity Map.....	B-10
One Postoffice, No External, Multiple Mail Clients	B-13
One LAN with One Postoffice and Remote Users.....	B-15
One LAN with One FAX Gateway.....	B-17
One LAN with Multiple Postoffice Configuration	B-19
Microsoft Mail Connection	B-20
Multiple LANs Joined Asynchronously.....	B-22
Hub Postoffice for WANs.....	B-23
Hub Postoffice for Gateway Access	B-24
Wide-Area Networking Through a Gateway	B-26
X.400 Gateway	B-28
X.400 Gateway to AT&T or Other ADMD	B-30
X.400 Gateway Backbone	B-32
Gateway to PROFS	B-34
PROFS Distribution Manager	B-36

PROFS Gateway Backboning.....	B-38
SNADS Gateway.....	B-40
SMTP Gateway.....	B-42
MHS Gateway.....	B-44
MCI Gateway.....	B-46
3+Mail Gateway.....	B-48
Configuration Guide Glossary.....	B-49

Appendix C **Flowcharts for Troubleshooting Windows for Workgroups** **C-1**

Flowchart Series 1: Troubleshooting Setup.....	C-3
Flowchart Series 2: Troubleshooting Hardware Problems.....	C-11
Flowchart Series 3: Troubleshooting Fonts.....	C-20
Flowchart Series 4: Troubleshooting Printing.....	C-24
Flowchart Series 5: Troubleshooting Operating Problems.....	C-28
Flowchart Series 6: Troubleshooting Multimedia.....	C-40

Appendix D **Accessibility for Individuals with Disabilities** **D-1**

Microsoft Support Services for the Deaf and Hard-of-Hearing.....	D-2
Access Pack for Windows.....	D-2
Keyboard Layouts for Single-Handed Users.....	D-3
Documentation on Audio Cassettes and Floppy Disks.....	D-3
Products for Blind and Low Vision Individuals.....	D-4
Customizing Windows for Workgroups.....	D-4
Getting More Information.....	D-6

Appendix E **Integrating Additional Protocols with Windows for Workgroups** **E-1**

About Additional Protocol Integration.....	E-2
NDIS Transport Protocol Integration.....	E-3
Integration with Banyan VINES.....	E-21
Integration with Other Network Operating Systems.....	E-21

Index **1**

Glossary

Other glossary resources

- The Glossary in Windows for Workgroups online help for definitions of terms related to the screen elements in Microsoft Windows.
- The *Computer Dictionary*, Microsoft Press, 1991, for definitions of general computing terms.

10BASE2 An implementation of the Ethernet IEEE standard on thin coaxial cable, a baseband medium, at 10 megabits per second. The maximum segment is just under 200 meters.

10BASE5 An implementation of the Ethernet IEEE standard on twinaxial cable, a baseband medium, at 10 megabits per second, with a 500 meter maximum segment length.

10BASE-T An implementation of the IEEE standard on 24 gauge unshielded twisted-pair wiring, a baseband medium, at 10 mbps.

16-bit An adjective that describes systems and software that handle information in words that are 2 bytes (16 bits) wide.

32-bit An adjective that describes hardware or software that manages data, program code, and program address information in 32-bit-wide words.

802.3 Defined by the IEEE, these standards govern the use of the CSMA/CD (Carrier Sense Multiple Access/Collision Detection) network access method used by Ethernet networks.

802.4 Defined by the IEEE, these standards govern the use of the token bus network access method.

802.5 Defined by the IEEE, these standards govern the use of the token ring network access method.

A

adapter segment A name sometimes used for the upper memory area, at hexadecimal addresses A000 through EFFF (640K to 1024K).

API An abbreviation for application programming interface, which is a set of routines that an application program uses to request and carry out lower-level services performed by the computer's operating system. For Windows, the API also helps applications manage windows, menus, icons, and other GUI elements. For a local area network, an API provides applications with routines for requesting services from lower levels of the network.

application Software such as a word processor or spreadsheet that you use to perform a specific type of work.

application layer The 7th layer of the OSI (Open Systems Interconnection) model concerned with application programs such as electronic mail, database managers, and file-server software.

ArcNet *Attached Resource Computer Network*: a local area network developed by Datapoint Corporation; designed as a token-passing bus architecture.

attribute A characteristic or property; for example, the color of a line, the length of a data field, or a typeface characteristic such as size.

B

backbone A high-speed cable joining together several network bridges, or possibly used to connect 2 or more stations.

backfilling To designate memory on an expanded memory card and make it available for use as conventional memory.

bank switching A technique for sequentially reading page frames in expanded memory. The 64K page frames are brought into the upper memory block from an expanded memory card, then read and swapped back to the card so that the next 64K page frame can be brought in.

baseband A signaling technique in which the signal is transmitted in its digital form and not changed by modulation techniques.

base input/output (I/O) port Specifies a channel through which information is transferred between your computer's hardware (such as your network card) and its CPU. The port appears to the CPU as an address.

base memory address Defines the address of the location in your computer's memory (RAM) that will be used by the network card to exchange information between your computer and the other computers you are connected to. This setting is sometimes called the *RAM start address*.

bezier curve A mathematical construct for describing a curve, used in CAD and illustration software to draw complex graphic shapes.

BIOS An acronym for basic input/output system, which is software (usually contained in ROM) that supports the transfer of information between system elements such as memory, disks, and the monitor.

bitmap A sequential collection of bits that represents, in memory, an image to be displayed on the screen.

bridge For a LAN, a device that links a network to another of the same type.

broadband Communications that take advantage of a transmission medium with a wide bandwidth such as coaxial cable. This allows many more bits per unit of time to be moved from point to point. services provided include video, voice, and additional data channels.

buffering The process of using buffers to hold data being moved to or from I/O devices such as serial ports and disk drives.

buffers A reserved part of memory where data is held temporarily until the data is transferred from storage to another location in memory. Some printers have their own buffers.

C

CEMM.EXE A 386 expanded memory manager provided with all Compaq 386 computers.

character-based interface The traditional user interface common to non-Windows applications, where all screen images are made up of text characters.

client A stand-alone workstation that uses the services of any type of server on a network.

client application Any Windows application that can accept, display, and store Object Linking and Embedding (OLE) objects. In DDE, the application that initiates the conversation.

client/server computing A mode of distributed network computing in which an application is executed cooperatively by two computers. The client component of the application (the front end) executes on one node, while the server component (the back end) runs on a second node. For PCs, communications between the nodes is implemented most frequently using the named pipes interprocess communication (IPC) protocol. Client/server computing allows more effective use of computing resources, higher performance, greater flexibility, simpler upgrades, and (for some applications) greater reliability and data integrity.

clock speed The rate in which the computer clock oscillates, usually given in MHz (one million cycles per second). The computer's clock rate is a prime determinant of overall processing speed. Typical clock speeds for AT-compatibles are 6, 8, 10, 12, 16, and 20 MHz; for 80386 machines, typical speeds are 16, 20, 25, and 33 MHz.

CMOS RAM A RAM chip that uses the complementary metal-oxide semiconductor process, which provides high speed and extremely low power consumption, making the chip useful for hardware such as the microcomputer clock.

code page For MS-DOS version 3.3 and later, a table that relates the keys on the keyboard to the characters displayed on the screen, thereby providing support for the character sets and keyboard layouts used in different countries. Code page 437 is the code page for the United States.

command interface A user interface that requires the user to enter commands at the command prompt. Also called command-line interface.

command prompt A screen symbol such as `C:>` that requests user input. Also called the MS-DOS prompt.

compound document A document file that contains embedded and linked data that was created in other kinds of applications.

conversation Two Windows applications using DDE to exchange data. The conversation is conducted through a channel. DDE client is the application that initiated the conversation; DDE server is the application that responds to the DDE client.

context switching The kind of multitasking that is done in standard mode Windows, where the CPU switches from one task to another, rather than allocating time to each task in turn, as in *timeslicing*.

conventional memory The physical memory in a PC from 0K to 640K.

CPU An abbreviation for the computer's central processing unit.

CSMA/CD (Carrier Sense Multiple Access with Collision Detection) A contention technique that allows multiple stations to successfully share a broadcast channel by avoiding contention via carrier sense and deference, and managing collisions via collision detection and packet retransmission.

cut-and-paste A method of moving data into another document or file, which could be for a different but compatible application.

D

database server Software such as Microsoft SQL Server that provides high-performance database access by splitting the DBMS function into a front-end component (where data is manipulated by users or applications) and a database-intelligent, back-end component (where data is stored, retrieved, and managed). In PC networks, the front-end component often resides on a personal computer controlled by a single user, while the back-end component resides on a high-performance PC that services requests for data submitted over the network by users.

datagram A packet (in the internet protocols) that, because it is not carried on a virtual circuit, includes complete destination address specifications (provided by the user, not the network) along with whatever data it carries. It is a one-way construct much like a telegram. All packets sent into a packet switched network are essentially datagrams.

Data Link Layer Layer 2 of the OSI model, it defines protocols governing data packetizing and transmission into and out of each node.

DB-15 A standardized connector with 15 pins for Ethernet transceivers.

DBMS An abbreviation for database management system.

DDE See *dynamic data exchange*.

demand paging The common implementation of virtual memory, where pages of data are read into memory from storage in response to page faults. See also *paging* and *virtual memory*.

device driver The software that the system uses to communicate with a device such as a display, printer, mouse, or communications adapter.

discardable code segment A Windows application code segment in memory, which can be discarded, overwritten, or reloaded from disk when necessary.

disk cache A part of RAM that is set aside to temporarily hold data read from disk. A disk cache doesn't have to hold an entire file, as a RAM disk

does, but can hold parts of running application software or parts of a data file. Disk-caching software such as SMARTDrive manages the process of swapping data to and from the disk cache.

DLC (Data Link Control) Protocols governing packet information including destination address, source address, and control information.

DLL See *dynamic-link library*.

DMA An abbreviation for Direct Memory Access, which is memory access that doesn't use the microprocessor, often used to transfer data directly between memory and a peripheral device such as a hard disk drive.

DOS An acronym for *disk operating system*, which is the operating system software that is loaded from a disk when you boot your computer. See also *MS-DOS*.

DOS Extender See *extended memory manager*.

double buffering The use of two buffers rather than one to temporarily hold data being moved to and from an I/O device. Double buffering increases data transfer speed because one buffer can be filled while the other is being emptied.

downloadable fonts Fonts that reside on your hard disk and that are sent to the printer as needed.

DPMI An abbreviation for DOS Protected Mode Interface, an industry standard that allows MS-DOS applications to execute code in the protected operating mode of the 80286 or 80386 processor. The DPMI specification is available from Intel Corporation.

dynamic data exchange (DDE) The form of interprocess communications (IPC) used by Microsoft Windows to support exchange of commands and data between two applications running simultaneously. In Windows 3.1, this capability is enhanced with *Object Linking and Embedding*.

dynamic-link library (DLL) An executable code module for Microsoft Windows that can be

loaded on demand and linked at run time, and then unloaded when the code is no longer needed.

E

EISA Acronym for Extended Industry Standard Architecture, a bus standard introduced in 1988 by a consortium made up of AST Research, Compaq, Epson, Hewlett-Packard, NEC, Olivetti, Tandy, Wyse, and Zenith.

EMM See *expanded memory manager*.

EMS See *Expanded Memory Specification*.

encapsulation In object-oriented programming, the grouping of data and the code that manipulates it into a single object. If a change is made to an object class, all instances of that class (that is, all objects) are changed. Encapsulation is one of the benefits of object-oriented programming.

Ethernet A local area network originally developed by Xerox Corporation. Xerox, Intel Corporation, and Digital Equipment Corporation drew up a standard for a 10-Mbps Ethernet, a design which became the basis for the IEEE's 802.3 specification.

expanded memory Memory provided by a physical add-in memory card or through a 386 expanded memory manager. Expanded memory is made available to application software as 16K pages, mapped into a 64K page frame.

expanded memory manager (EMM) A driver that provides a software interface to expanded memory. An expanded memory manager is specific to a particular physical expanded memory board or uses the 80386 processor to emulate expanded memory in extended memory.

Expanded Memory Specification (EMS) A description of a technique for adding memory to PCs to bypass the 640K limit on addressable memory. The EMS supports additional memory through a hardware expansion board and an expanded memory manager. To use expanded memory, an application must be specifically designed to interact with an expanded memory manager. There are two expanded memory specifications: LIM 3.2 provides for a maximum of four

contiguous 16K bankable pages; LIM 4.0 provides for a maximum of sixty-four 16K bankable pages, which do not have to be contiguous.

extended memory The memory at physical addresses from 1024K that can be accessed by a 80x86 CPU running in protected mode. Windows version 3.0 and later take full advantage of extended memory, including all memory above 640K, when running in standard mode or 386 enhanced mode.

extended memory manager (XMM) A driver such as HIMEM.SYS that is necessary to access extended memory. Applications do not have to be customized to interact with an extended memory manager to use extended memory under Windows 3.x. In some applications, the extended memory manager is referred to as an MS-DOS Extender.

Extended Memory Specification (XMS) A Microsoft standard that provides a way for real mode applications to access extended memory. Non-Windows applications running under Windows 3.x can use the XMS to access extended memory. The XMS document and sample source code is available from the Microsoft Information Center.

F

fault tolerance The ability of a system to recover from an error, a failure, or a change in environmental conditions (such as loss of power). True fault tolerance provides for fully automatic recovery without disruption of user tasks or files, in contrast to manual means of recovery such as restoring data loss with tape backup files.

file server A network node that usually consists of fixed-disk storage and a CPU, for storing programs and data shared by users on a network. File servers offer operating system-type file and print capabilities, while database servers provide database intelligence such as transaction processing, indexing, logging, security, and so on.

font In Windows, a complete set of characters for one style of a specific typeface, including all the letters, numbers, and punctuation marks. For example, Courier New is a font.

font family In Windows, a group of typefaces with similar characteristics. The families that Windows recognizes for font installation are Roman, Swiss, Modern, Script, and Decorative. For example, Arial, Arial Bold, Arial Bold Italic, Arial Italic, Small Fonts, and MS Sans Serif are all part of the sans serif Swiss font family.

G

grabber Software that supports exchange of data in video memory between non-Windows applications and Windows 3.x.

graphical user interface (GUI) An interface that allows users to choose commands and other options by pointing to a graphical icon and then activating the choice with either the keyboard or a mouse.

group In Windows, a collection of applications in the Program Manager window.

GUI See *graphical user interface*

H

high memory area (HMA) The first 64K of extended memory, into which MS-DOS 5.0, drivers and utilities can be loaded to save conventional memory.

hub A special unit in a network using a star configuration to which each workstation is connected. The hub provides a common connection so that all of the computers can communicate with one another.

I

IEEE The Institute for Electrical and Electronic Engineers.

Impedance A measurement of resistance to current that a transmission medium offers. It is measured in ohms.

independent software vendor (ISV) A participant in the computing industry marketplace whose primary focus is development of software.

interface 1) The user interface, where people communicate with the software through

commands and other devices. 2) The connections in software that allow an application to work with the operating system, or that allow the operating system to work with the hardware. 3) The hardware cards, plugs, and other devices used to move data from place to place.

interrupts A request-for-attention signal sent by either hardware or software to the CPU that causes the CPU to suspend some operations and transfer control to an interrupt handler (designated, for example, as Int 13).

IPX Internetwork Packet Exchange

I/O An abbreviation for input/output.

IRQ Abbreviation for “interrupt request lines,” which are hardware lines over which I/O devices, keyboards, and disk drives can send interrupts to the CPU. The IRQs are built into the hardware, with preassigned priority levels.

ISA Abbreviation for Industry Standard Architecture, an unofficial designation for the bus design of the IBM PC/XT.

ISO Abbreviation for International Standards Organization, an international association of standard-setting organizations.

ISO/OSI model The International Standards Organization Open Systems Interconnection model, which standardizes levels of services and types of interaction for exchanging data through a communications network. The model separates computer-to-computer communications into seven layers—the application layer, data-link layer, network layer, physical layer, presentation layer, session layer, and transport layer.

item A reference to a piece of data (such as an integer, a string, a range of cells in a worksheet, a chart, or a bitmap) that can be passed between two applications engaged in a DDE conversation.

K

kernel 1) The part of an operating system that performs basic functions such as allocating hardware resources. 2) The central part of other

system software. 3) One of the core dynamic-link libraries that make up Windows.

keyword A predefined word in a computer or command language.

L

LAN See *local area network*.

LAN Manager The Microsoft networking environment.

LIM (Lotus/Intel/Microsoft) 3.2 and 4.0
See *Expanded Memory Specification*.

local area network (LAN) Two or more computing units connected for local resource sharing, or a network in which communications are limited to an area such as a single office building and do not extend across public rights-of-way. A LAN allows PCs to have access to common data and peripherals, and it typically consists of PCs with adapter cards, file servers, printers, gateways to departmental or corporate computers, and network software to integrate these components. More sophisticated LANs permit sharing of computing resources, along with sharing printers and file storage facilities. For example, networks equipped with MS LAN Manager support *client/server computing*.

local heap A memory storage area limited to 64K in size.

logging on The procedure that links a user to a network.

M

MCI An abbreviation for Media Control Interface, a standard control interface for multimedia devices and files.

Media Access Control (MAC) A sublayer, as defined by the IEEE Project 802, of the Data Link networking layer. This sublayer communicates directly with the network adapter card and is responsible for delivering error-free data between two computers on the network.

memory-mapped files An operating system feature that maps a file into a program’s address

space, allowing an application to read and modify a file's content as if the file were an in-memory data structure. Memory-mapped files use demand paging to allow applications to access and manipulate files regardless of how big the files are. Without memory-mapped files, every application that works with large files must include code to handle the movement of data between the disk and the main memory.

memory protection Strict management of system memory to prevent the corruption of any application's program code or data by interference from another application. Memory protection ensures that concurrent applications are compatible and prevents them from interfering with system processes.

Microsoft Consulting Services (MCS) The Microsoft consulting service, founded in 1990 to help clients develop mission-critical business systems and to work with major systems integrators to provide specialized expertise in PC technologies in cooperative processing environments.

MIPS An acronym for millions of instructions per second, which is a measure of processing speed that refers to the average number of machine language instructions performed by the CPU in one second. A typical Intel 80386-based PC is a 3 to 5 MIPS machine, whereas an IBM System 370 mainframe typically delivers between 5 and 40 MIPS. MIPS measures raw CPU performance, but not overall system performance.

Monolithic protocol stack A protocol driver that manages all functions of the MAC through the Transport layer in one protocol driver.

movable code segment A Windows application code segment that can be moved around in memory.

MSIPX An NDIS-compliant IPX protocol stack provided by Microsoft for Windows for Workgroups.

MS-DOS Microsoft Disk Operating System, the standard operating system for computers equipped with the Intel 8086 and 8088 microprocessors, and

for 80286 and 80386 microprocessors running in 8086 mode.

MS-DOS-based application A term used in this document as a shorthand term to refer to an application that is designed to run with MS-DOS but not specifically with Windows and that may not be able to take full advantage of all Windows features (such as memory management).

MS-Net Microsoft Network, the MS-DOS networking systems software product shipped in 1985. Largely superseded by Microsoft LAN Manager.

multitasking The concurrent management of two or more distinct tasks by a computer. Although a computer with a single processing unit can only execute one application's code at a given moment, a multitasking operating system can load and manage the execution of multiple applications, allocating processing cycles to each in sequence. Because of the processing speed of computers, the apparent result is the simultaneous processing of multiple tasks. Standard mode Windows performs multitasking only in the form of context switching; 386 enhanced mode allows multitasking in the form of timeslicing.

N

NCB Network Control Block; the structure in which the redirector packages SMB requests meant for remote workstations.

NDIS Network Device Interface Specification; jointly developed in 1989 by Microsoft and 3Com; A standard defining an interface for communication between the MAC layer and the transport protocol (or simply protocol) drivers that reside on layers 3 and 4 of the OSI model.

NetBIOS An API used by application programs on a PC LAN that uses MS-DOS or some version of UNIX, providing application developers with a uniform set of commands for requesting lower-level network services.

NetBEUI NetBIOS Extended User Interface; The primary protocol used by Windows for Workgroups. A small and efficient protocol

designed for use on a departmental LAN of 20 to 200 workstations, first introduced by IBM in 1985.

NetWare A local area network operating system produced by Novell, Inc. Several families of NetWare products exist, including NetWare 386.

network A configuration of data processing devices and software connected for information interchange.

network adapter card also called a network interface card or NIC; installed in each computer on the LAN and includes a connector port on the back of the network card for connecting to a cabling medium.

network application A network program, or combination of a program and data, that performs a task (typically involving two or more computers).

network layer Layer 3 of the OSI model which defines protocols governing data routing.

O

object Encapsulated data in a compound document. For example, a single cell, a range of cells, or an entire spreadsheet could be embedded as an object in a word processing text file. Any data can be an object if it was created in a Windows application that uses OLE.

object handler A dynamic-link library for an OLE server application. This DLL acts as an intermediary between the client and server applications; for example, to improve performance when it's necessary to redraw an object in the window of the client application.

Object Linking and Embedding (OLE) An enhancement to DDE protocol that allows you to embed or link data created in one application in a document created in another application, and subsequently edit that data in the original application without leaving the compound document.

OEM An abbreviation for original equipment manufacturer.

Open Software Foundation (OSF) A consortium of computing industry vendors organized

to develop an open, UNIX-based operating system standard. OSF's first operating system release, called OSF/1, is built on the IBM UNIX-based AIX kernel and the Motif graphical user interface.

Open Systems Interconnect (OSI) An ISO publication that defines seven independent layers of communication protocols. Each layer enhances the communication services of the layer just below it and shields the layer above it from the implementation details of the lower layer. In theory, this allows communication systems to be built from independently developed layers.

operating system The software that governs the operation of a computer, providing services such as resource allocation, scheduling, input/output control, and data management. Popular operating systems include MS-DOS and UNIX.

OSI See *Open Systems Interconnect*.

P

package In a compound document, an embedded icon that contains an object, a file or part of a file, or a command.

page fault An interrupt that results when software requests a memory location that is being saved for auxiliary storage.

page frame A physical address in the upper memory area where a page of expanded memory can be mapped. The page frame itself can be a maximum of four 16K pages.

paging A technique for implementing virtual memory, where the virtual address space is divided into fixed-sized blocks called pages, each of which can be mapped onto any physical addresses available on the system.

palette management A mechanism to choose which colors to display when the display device supports a limited number of colors for simultaneous display. For example, a VGA monitor can display 16 colors out of an overall palette of 256, so a 16-bit bitmap with more than 16 colors would have to be palettized.

paste A function that copies the contents of the Clipboard and inserts it at the current cursor position.

path 1) In MS-DOS, the route the operating system follows to locate files on a disk or disks. 2) On a network, the route between any two nodes.

PC-NET The IBM PC LAN program, an MS-DOS-based networking systems software product developed by Microsoft and IBM, which has been shipping since 1985.

physical layer The layer of the OSI model that establishes protocols for voltage, data transmission timing, and rules for "handshaking."

PIF An abbreviation for program information file, which contains information about a non-Windows application, such as how much memory it needs. Windows uses this information to run the application in the most efficient way.

pipe 1) A portion of memory used to pass data from one process to another, symbolized in MS-DOS by the | character. For example **dir | sort** sends the output of the directory listing to the **sort** command. 2) In a sample statement for a Windows initialization file or in an MS-DOS command, the symbol used to show that only one of the possible values can be used in the statement.

pitch Denotes the horizontal size of a fixed-width font in characters per inch.

plenum The short space in many buildings between the false ceiling and the floor above, and is used to circulate warm and cold air through the building. Fire codes are very specific on the type of wiring that can be routed through this area.

plenum cabling Coaxial cable containing special materials in its insulation and cable jacket certified to be fire-resistant and which produce a minimum amount of smoke in order to minimize poisonous chemical fumes.

point size The vertical size of a font. A point is about 1/72 inch.

POSIX A set of software standards being developed by the IEEE POSIX Working Group, to allow applications to be written to a single

target environment in which they can run unchanged on a variety of systems. The POSIX specifications define characteristics for operating systems, DBMS, data interchange, network services, user interface, and programming interfaces. Several vendors plan to make their operating systems POSIX-compliant.

presentation layer The 6th layer of the OSI model concerned with protocols for network security, file transfers, and format functions.

print server A network node, usually consisting of fixed-disk storage and a CPU, that controls one or more printers that can be shared by users.

printable screen fonts Windows screen fonts that can be translated for output on the printer.

program group A set of several programs whose icons can be acted on as a single entity in Windows Program Manager. For example, if icons are associated with a single group icon, clicking the program group icon causes the icons for each of the programs in the group to be displayed.

protected mode An operating mode for the Intel 80286 and higher processors that supports multitasking, data security, and virtual memory.

Protocol A set of rules or procedures commonly agreed on by industry wide committees such as the IEEE. Also could be defined as a specific set of instructions used to control how and when to format and send data via communication channels.

provider interface An API for the provision of centralized administrative support of the client workstation over the network. Providers of network services will use this Microsoft Windows API, which will be fully open to the development community.

PVC Poly-vinyl chloride; a chemical used to construct the insulation and the cable jacket for most types of coaxial cable. PVC coaxial cable is flexible and can easily be routed in the exposed areas of an office.

Q

query 1) A specific request for data or instructions. 2) A request for information from a database using specific conditions; for example, a request for a list of all customers with balances greater than \$1,000.

Query-By-Example (QBE) A program product developed by IBM to work in conjunction with DB2, a mainframe relational DBMS. QBE is used to write queries graphically.

R

raster fonts A set of characters for screen display or printing that are stored as bitmaps in specific character sizes.

registration database The REG.DAT file, which provides the information used by File Manager and Print Manager to support drag-and-drop and other actions, and which is used by OLE applications to find file types and OLE capabilities for embedded objects.

relational database 1) A database organized and accessed according to relationships between data items. 2) A data structure perceived by its users as a collection of tables. A relational database consists of tables, rows, and columns. Typical examples are dBase IV from Borland and R:Base 5000 from Microrim. Relational databases differ from non-relational databases in that there are no system dependencies stored within the data. Hierarchical databases are not relational, because they contain pointers to other data.

resource 1) Any facility of a computing system or operating system required by a job or task, including memory, input/output devices, processing unit, data files, and control or processing programs. 2) A network component such as a file, printer, or serial device that is shared by other components of the network. 3) In Microsoft Windows, the definition of elements such as fonts, templates, accelerators, and error messages are all stored in resource files.

ring A network configuration where a series of attached devices are connected by unidirectional transmission links to form a closed path.

routine A program, or a sequence of instructions called by a program, that has some general or frequent use.

S

scalability The ability of a computing element such as a process, processor, or structure to grow seamlessly.

SDK The Windows Software Development Kit.

server On a local area network, the computer that runs the administrative software to control access to the network. The server makes network resources available to the workstations. See also *file server* and *print server*.

server application 1) On a network, a process or program that runs on a server, typically as a network resource; for example, a communications program that organizes and allocates network communications resources. More sophisticated server programs, known as client/server applications, run interactively with an application resident on network nodes. See also *client/server computing*. 2) In a compound document, any Windows application that allows you to edit an object when Windows informs it that you selected the object in an OLE client application. 3) In a DDE conversation, the application responding to the DDE request made by the client application.

session layer The 5th layer of the OSI model concerned with network management functions including passwords and network monitoring and reporting.

shadow ROM A process used in many 386 machines to map ROM BIOS activities into faster 32-bit RAM memory. Shadow memory must be loaded with BIOS routines each time the computer boots.

shared resource A printer, file, or serial computer communications device made available through the LAN to multiple computers not physically attached to the resource.

source code A set of programming language instructions that must be translated to machine instructions before the program can run.

spooler A program that intercepts the data going to a device driver and writes it to the disk. The data is later printed or plotted when the required device is available. A spooler prevents the intermixing of output from different sources.

Structured Query Language (SQL) A database query and programming sublanguage. SQL (commonly pronounced “sequel” and originally developed for IBM mainframe computers) is an established set of statements used to add, delete, or update information in a table, or request information from one or more tables in the form of a report. There is now an ANSI-standard SQL definition for all computer systems.

system administrator The person responsible for planning computer or network installation and for ensuring the successful installation and use of the system by other users.

T

TCP/IP See *Transport Control Protocol/Internet Protocol*.

timeslicing A method of multitasking that allocates CPU attention to tasks in fractions of a second, making it appear to users that the multitasking operations are happening simultaneously. Tasks are either assigned priority levels or processed in sequence to maintain order.

Token Ring A local area network which is an implementation of IEEE standard 802.5, the standard for the token-ring access method. The access method, more than the ring shape, distinguishes token-ring networks from other LANs.

topic Information that defines the “subject” of a DDE conversation and represents some unit of data that is meaningful to the DDE server conversation. For most applications that operate on files, this is a filename (for example, SALES.XLS).

Transceiver An AUI (Attachment Unit Interface) device for receiving and transmitting data that often provides collision detection as well.

Transport Control Protocol/Internet Protocol (TCP/IP) A set of network protocols that has become an industry standard in engineering, government, and educational local area network environments. The TCP/IP protocol family includes transport, file transfer, terminal emulator, messaging, and network management definitions.

transport layer The 4th layer of the OSI model concerned with protocols for error recognition and recovery as well as regulation of information flow.

transparent application integration The ability to manipulate information with a variety of applications within a single document, without moving from one application to another to produce the desired results. This is implemented as *Object Linking and Embedding* in Windows 3.1.

TSR Terminate-and-stay-resident software program.

U

upper memory area The part of physical memory that starts at 640K and ends at 1024K (hexadecimal addresses A000 through FFFF). Also called the *adapter segment*, because that portion of memory is often used by hardware adapters such as display adapters.

upper memory blocks (UMB) The unused parts of the upper memory area. For an 80386 or higher computer, information can be mapped (or copied) from another type of memory to upper memory blocks, freeing more conventional memory.

user context A user session created by an operating system in response to a logon request, and typically characterized by privilege sets that strictly define the user’s authority to access system resources and information. Contexts restrict unauthorized access to facilities and data and protect the system itself from user and applications interference, accidental or otherwise. Contexts are

a feature of most multi-user operating systems, usually integrated with the security system.

V

VAR Value-added reseller. This term often refers to a reseller of computer products who also provides both integration and software customization services to business clients.

VCPI An abbreviation for Virtual Control Program Interface. An 80386 memory management standard created by Phar Lap Software in conjunction with other software developers to provide an interface between applications using MS-DOS Extenders on an 80386 machine and applications using 386 expanded memory managers. For example, the 386 expanded memory managers QEMM, 386MAX, and CEMM support the VCPI specification.

vector fonts A set of lines that connect points to form characters.

virtual device A device that software can refer to but that doesn't physically exist.

virtual hard drive memory factor The available space on a hard drive partition that Windows can address as physical memory.

virtual machine (VM) Software that mimics the performance of a hardware device. For Intel 80386 and higher processors, a virtual machine is protected memory space that is created through the processor's hardware capabilities.

virtual memory A technique of simulating additional memory for an application to use. Applications access the memory through virtual addresses, which are mapped onto physical addresses on the disk. Two common ways to implement virtual memory are *paging* and *segmentation* (which was used by 80286 and earlier Intel processors to address memory).

Virtual Memory Manager The facility in Windows 386 enhanced mode that manages the task of swapping data in and out of virtual memory for multiple non-Windows applications running virtual machines. The Windows Virtual

Memory Manager can manage paging in up to 48 MB of virtual memory.

visual programming The use of graphical development tools and visual metaphors to create software.

VMM An abbreviation for Virtual Memory Manager.

W

wide area network (WAN) A network that provides data communication capability in areas larger than those serviced by a local area network. A WAN typically relies on serial line protocols to interconnect subnetworks.

Windows application A term used in this document as a shorthand term to refer to an application that is designed to run with Windows and does not run without Windows. All Windows applications follow similar conventions for arrangement of menus, style of dialog boxes, and keyboard and mouse use.

workstation A terminal or personal computer, usually connected to a mainframe or to a network, on which a user can run applications. A workstation generally cannot share its resources with other network nodes or a host computer. However, on a LAN Manager network, an OS/2 workstation can share resources by invoking the peer server option. See also *server*.

X

X.25 A standard network communications protocol for implementing the internetwork communications function specified by the third network layer of the ISO/OSI model. X.25 is a packet-switched protocol often used in wide-area network configurations. MS LAN Manager is supported by X.25 communications packages provided by various third-party vendors.

XMM See *extended memory manager*.

XMS See *Extended Memory Specification*.

Windows for Workgroups Resource Directory

This appendix contains a list of resources to support learning and using Microsoft Windows for Workgroups.

With Windows accepted as a standard for PC computing, many companies are making products and services available to enhance the Windows and Windows for Workgroups environment. Microsoft also offers a wide array of support services.

Contents of the Resource Directory

Microsoft Support Services	A-2
Microsoft Product Support Services Worldwide	A-3
Microsoft Consulting Services	A-9
Microsoft Solutions Channels—Consulting Program	A-9
Microsoft Solutions Channels—Training Program.....	A-10
Microsoft Certified Professional Program	A-10
Microsoft University Technical Training Straight from the Source	A-12
Online Services.....	A-19
Microsoft Developer Network	A-20

Microsoft Support Services

The following numbers can be called for assistance with using Microsoft Windows for Workgroups. Outside the U.S., contact the Microsoft subsidiary office that serves your area.

Microsoft FastTips

An interactive, automated system providing support by touch-tone phone at no charge through toll lines. FastTips provides fast access to answers to common questions, delivered by phone recording or fax. It also includes a library of technical notes that are available by fax or mail. FastTips are available 7 days a week, 24 hours a day, including holidays.

Microsoft Windows for Workgroups	(206) 635-7245
Microsoft Excel for Windows	(206) 635-7071
Microsoft Word for Windows	(206) 635-7231

Microsoft Download Service

Use your modem (1200, 2400, 9600 baud, no parity, 8 data bits, 1 stop bit) to access technical notes and printer drivers online at no charge. Available 7 days a week, 24 hours a day, including holidays.

Microsoft Download Service	(206) 936-6735
----------------------------	----------------

Dedicated Support Numbers

Support at no charge through a toll line Monday through Friday, 6 A.M. to 6 P.M., Pacific time, excluding holidays.

Microsoft Windows for Workgroups	(206) 637-7098
Microsoft Excel for Windows	(206) 635-7070
Microsoft PowerPoint	(206) 635-7145
Microsoft Word for Windows	(206) 462-WORD
Microsoft Works for Windows	(206) 635-7130

Microsoft TDD/TT (Text Telephone)

Microsoft Product Support Services is available for the deaf and hard of hearing. Using a special TDD/TT modem, call Monday through Friday, 6 A.M. to 6 P.M., Pacific time, excluding holidays.

Microsoft TDD/TT	(206) 635-4948
------------------	----------------

Microsoft's support services are subject to Microsoft's prices, terms, and conditions in place in each country at the time the services are used.

Microsoft Product Support Services Worldwide

Microsoft subsidiary offices and the countries they serve are listed below.

Area	Telephone numbers
-------------	--------------------------

Argentina	Microsoft de Argentina S.A. Phone: (54) (1) 814-0356 Fax: (54) (1) 814-0372
Australia	Microsoft Pty. Ltd. Phone: (61) (02) 870-2200 Fax: (02) 805-1108 Bulletin Board Service: (612) 870-2348 Technical Support: (61) (02) 870-2131 Sales Information Centre: (02) 870-2100
Austria	Microsoft Ges.m.b.H. Phone: 0222 - 68 76 07 Fax: 0222 - 68 16 2710 Information: 060 - 89 - 247 11 101 Prices, updates, etc.: 060 - 89 - 3176 1199 CompuServe: msce (Microsoft Central Europe) Technical support: Windows, Windows for Workgroups, Microsoft Mail: 0660 - 65 - 10 Excel for Windows, Excel for OS/2, PowerPoint for Windows: 0660 - 65 - 11 Word for MS-DOS, Windows Write: 0660 65 - 12 Word for Windows, Word for OS/2: 0660 - 65 - 13 Works for MS-DOS, Works for Windows, Publisher, WorksCalc, WorksText: 0660 - 65 - 14 C PDS, FORTRAN PDS, Pascal, Macro Assembler PDS, QuickC, QuickC for Windows, QuickPascal, QuickAssembler, Profiler: 0660 - 65 - 15 COBOL PDS, Basic PDS, QuickBASIC, Visual Basic: 0660 - 65 - 16 MS-DOS: 0660 - 65 - 17 Macintosh Software: 0660 - 65 - 18 Project for Windows, Project for MS-DOS, Multiplan, Mouse, Flight Simulator, Paintbrush, Chart : 0660 - 67 - 38 FoxPro: 0660 - 67 - 61
Baltic States	See Germany
Belgium	Microsoft NV Phone: 02-7322590 Fax: 02-7351609 Technical Support Bulletin Board Service: 02-7350045 (1200/2400/9600 baud, 8 bits, no parity, 1 stop bit, ANSI terminal emulation) (Dutch speaking) Technical Support: 02-5133274 (English speaking) Technical Support: 02-5023432 (French speaking) Technical Support: 02-5132268 Technical Support Fax: (31) 2503-24304
Bermuda	See Venezuela

Bolivia	See Argentina
Brazil	Microsoft Informatica Ltda. Phone: (55) (11) 530-4455 Fax: (55) (11) 240-2205 Technical Support Phone: (55) (11) 533-2922 Technical Support Fax: (55) (11) 241-1157 Technical Support Bulletin Board Service: (55) (11) 543-9257
Canada	Microsoft Canada Inc. Phone: 1 (416) 568-0434 Fax: 1 (416) 568-4689 Technical Support Phone: 1 (416) 568-3503 Technical Support Fax: 1 (416) 568-4689 Technical Support Bulletin Board Service: 1 (416) 507-3022
Caribbean countries	See Venezuela
Central America	See Venezuela
Chile	See Argentina
Colombia	See Venezuela
Denmark	Microsoft Danmark AS Phone: (45) (44) 89 01 00 Fax: (45) (44) 68 55 10
Ecuador	See Venezuela
England	Microsoft Limited Phone: (44) (734) 270000 Fax: (44) (734) 270002 Upgrades: (44) (81) 893-8000 Technical Support: Main Line (All Products): (44) (734) 271000 Windows Direct Support Line: (44) (734) 271001 Database Direct Support Line: (44) (734) 271126 MS-DOS 5 Warranty Support: (44) (734) 271900 MS-DOS 5 Fee Support Line: (44) (891) 315500 OnLine Service Assistance: (44) (734) 270374 Bulletin Board Service: (44) (734) 270065 (2400 Baud) Fax Information Service: (44) (734) 270080
Finland	Microsoft OY Phone: (358) (0) 525 501 Fax: (358) (0) 522 955

Area	Telephone numbers	<i>(continued)</i>
France	Microsoft France Phone: (33) (1) 69-86-46-46 Telex: MSPARIS 604322F Fax: (33) (1) 64-46-06-60 Technical Support Phone: (33) (1) 69-86-10-20 Technical Support Fax: (33) (1) 69-28-00-28	
French Polynesia	See France	
Germany	Microsoft GmbH Phone: 089 - 3176-0 Telex: (17) 89 83 28 MS GMBH D Fax: 089 - 3176-1000 Information: 0130 - 5099 Prices, updates, etc.: 089 - 3176 1199 Bulletin board, device drivers, tech notes : BTX: microsoft# or *610808000# CompuServe: msce (Microsoft Central Europe) Technical support: Windows, Windows for Workgroups, Microsoft Mail: 089 - 3176 - 1110 Excel for Windows, Excel for OS/2, PowerPoint for Windows: 089 - 3176 - 1120 Word for MS-DOS, Windows Write: 089 - 3176 - 1130 Word for Windows, Word for OS/2: 089 - 3176 - 1131 Works for MS-DOS, Works for Windows, Publisher, WorksCalc, WorksText: 089 - 3176 - 1140 C PDS, FORTRAN PDS, Pascal, Macro Assembler PDS, QuickC, QuickC for Windows, QuickPascal, QuickAssembler, Profiler: 089 - 3176 - 1150 COBOL PDS, Basic PDS, QuickBASIC, Visual Basic: 089 - 3176 - 1151 MS-DOS: 089 - 3176 - 1152 Macintosh Software: 089 - 3176 - 1160 Project for Windows, Project for MS-DOS, Multiplan, Mouse, Flight Simulator, Paintbrush, Chart: 089 - 3176 - 1170 FoxPro: 089 - 3176 - 1180	
Hong Kong	Microsoft Hong Kong Limited Technical Support: (852) 804-4222	
Ireland	See England	
Israel	Microsoft Israel Ltd. Phone: 972-3-752-7915 Fax: 972-3-752-7919	

Area	Telephone numbers	<i>(continued)</i>
Italy	Microsoft SpA Phone: (39) (2) 269121 Telex: 340321 I Fax: (39) (2) 21072020 Technical Support: Excel for Windows, Project for Windows, Works for Windows: (39) (2) 26901361 Word, Works for MS-DOS: (39) (2) 26901362 Windows, PowerPoint, Publisher, Windows for Workgroups, Works : (39) (2) 26901363 Basic, COBOL, Visual Basic, MS-DOS, Fox Products: (39) (2) 26901364 C, FORTRAN, Pascal, Macro Assembler (MASM), and SDKs: (39) (2) 26901354 LAN Man, SQL Server, Microsoft Mail, Microsoft Mail Gateways: (39) (2) 26901356	
Japan	Microsoft Company Ltd. Phone: (81) (3) 3363-1200 Fax: (81) (3) 3363-1281 Technical Support: MS-DOS Applications: (81) (3) 3363-0160 Windows Applications: (81) (3) 3363-5040 Language Products (Microsoft C, Macro Assembler [MASM], QuickC): (81) (3) 3363-7610 Language Products (Basic, FORTRAN, Visual Basic, Quick Basic): (81) (3) 3363-0170 All Products Technical Support Fax: (81) (3) 3363-9901	
Korea	Microsoft CH Phone: (82) (2) 552-9505 Fax: (82) (2) 555-1724 Technical Support: (82) (2) 563-9230	
Liechtenstein	See Switzerland (German)	
Luxemburg	Microsoft NV Phone: (32) 2-7322590 Fax: (32) 2-7351609 Technical Support Bulletin Board Service: (31) 2503-34221 (1200/2400/9600 baud, 8 bits, No parity, 1 stop bit, ANSI terminal emulation) (Dutch speaking) Technical Support: (31) 2503-77877 (English speaking) Technical Support: (31) 2503-77853 (French speaking) Technical Support: (32) 2-5132268 Technical Support Fax: (31) 2503-24304	
México	Microsoft México, S.A. de C.V. Phone: (52) (5) 325-0910 Fax: (52) (5) 280-0198 Technical Support: (52) (5) 325-0912 Sales: (52) (5) 325-0911	

Area	Telephone numbers	<i>(continued)</i>
Netherlands	Microsoft BV Phone: 02503-13181 Fax: 02503-37761 Technical Support Bulletin Board Service: 02503-34221 (1200/2400/9600 baud, 8 bits, No parity, 1 stop bit, ANSI terminal emulation) (Dutch speaking) Technical Support: 02503-77877 (English speaking) Technical Support: 02503-77853 Technical Support Fax: 02503-24304	
New Zealand	Technology Link Centre Phone: 64 (9) 358-3724 Fax: 64 (9) 358-3726 Technical Support Applications: 64 (9) 357-5575	
Northern Ireland	See England	
Norway	Microsoft Norway AS Phone: (47) (2) 95 06 65 Fax: (47) (2) 95 06 64 Technical Support: (47) (2) 18 35 00	
Papua New Guinea	See Australia	
Paraguay	See Argentina	
Peru	See Venezuela	
Portugal	MSFT, Lda. Phone: (351) 1 4412205 Fax: (351) 1 4412101	
Puerto Rico	See Venezuela	
Republic of China	Microsoft Taiwan Corp. Phone: (886) (2) 504-3122 Fax: (886) (2) 504-3121	
Republic of Ireland	See England	
Scotland	See England	
Spain	Microsoft Iberica SRL Phone: (34) (1) 804-0000 Fax: (34) (1) 803-8310 Technical Support: (34) (1) 803-9960	

Area	Telephone numbers	<i>(continued)</i>
Sweden	Microsoft AB Phone: (46) (8) 752 56 00 Fax: (46) (8) 750 51 58 Technical Support: Applications: (46) (8) 752 68 50 Development & Network products: (46) (8) 752 60 50 MS-DOS: (46) (071) 21 05 15 (SEK 4.55/min) Sales Support: (46) (8) 752 56 30 Bulletin Board Service: (46) (8) 750 47 42 Fax Information Service: (46) (8) 752 29 00	
Switzerland	(German speaking) Microsoft AG Phone: 01 - 839 61 11 Fax: 01 - 831 08 69 Information: 0049 - 89 - 247 11 101 Prices, updates, etc.: 0049 - 89 - 3176 1199 CompuServe: msce (Microsoft Central Europe) Technical support: Windows, Windows for Workgroups, Microsoft Mail: 01 - 342 - 4085 Excel for Windows, Excel for OS/2, PowerPoint for Windows: 01 - 342 - 4082 Word for MS-DOS, Windows Write: 01 - 342 - 4083 Word for Windows, Word for OS/2: 01 - 342 - 4087 Works for MS-DOS, Works for Windows, Publisher, WorksCalc, WorksText: 01 - 342 - 4084 C PDS, FORTRAN PDS, Pascal, Macro Assembler PDS, QuickC, QuickC for Windows, QuickPascal, QuickAssembler, Profiler: 01 - 342 - 4036 COBOL PDS, Basic PDS, QuickBASIC, Visual Basic: 01 - 342 - 4086 MS-DOS: 01 - 342 - 2152 Macintosh Software: 01 - 342 - 4081 Project for Windows, Project for MS-DOS, Multiplan, Mouse, Flight Simulator, Paintbrush, Chart: 01 - 342 - 0322 FoxPro: 01 / 342 - 4121 (French speaking) Microsoft SA, office Nyon Phone: 022 - 363 68 11 Fax: 022 - 363 69 11 Technical support: 022 - 738 96 88	
Uruguay	See Argentina	
Venezuela	Corporation MS 90 de Venezuela S.A. Phone: 0058.2.914739 Fax: 0058.2.923835	
Wales	See England	

Microsoft Consulting Services

Microsoft Consulting Services (MCS) has helped over 250 large organizations worldwide to capitalize on their investments in information technology.

MCS consultants are full-time Microsoft employees chartered to transfer expertise, knowledge, and skills to corporate developers. They can help you to do the following:

- Apply powerful technologies such as Windows, Windows NT, graphical user interface, 32-bit computing, messaging, object orientation, and pen computing.
- Design and build effective and robust desktop applications.
- Take full advantage of network, database, and development tools.
- Gain client-server computing expertise.
- Design and build mission-critical applications such as executive information systems, order entry, investment bank trading, sales force automation, payroll, and document management.

For more information about Microsoft Consulting Services, please contact an MCS office near you.

In the United States:	Canada	(416) 568-0434
Central Region (708) 495-5550	Australia	(61) (2) 870-2200
Northeast Region (617) 487-6500	France	(33) (1) 6986-4480
South Region (214) 458-1739	Germany	(49) (89) 3176-0
West Region (206) 635-1980	Italy	(39) (2) 210-7361
	United Kingdom	(44) (734) 270-001

Microsoft Solutions Channels—Consulting Program

Independent computer consultants who have demonstrated technical expertise with Microsoft products qualify for this program. They offer a broad range of consulting services, including customized design of user interfaces and applications. For more information on joining this program, or for the name of a qualified consultant, call 1-800-227-4679. (Outside the U.S., contact your Microsoft subsidiary office to see if they support a similar program in your region.)

Microsoft Solutions Channels—Training Program

The Training Channel Program is open to training centers and trainers committed to providing high quality training on Microsoft products. Training centers recommended by Microsoft offer a wide range of training on Windows and Applications for Windows and the Macintosh. For more information on joining this program, call 1-800-227-4679. For the name of a qualified trainer in your area, call 1-800-426-9400. (Outside the U.S., contact your Microsoft subsidiary to see if they support a similar program in your region.)

Microsoft Certified Professional Program

If you're a support coordinator, support engineer, systems engineer, reseller sales support engineer, consultant, network administrator, or anyone else who must gain and display technical expertise in Microsoft Windows, you should become a Microsoft Windows Certified Professional. Current certification—in this case for Microsoft Windows version 3.1—demonstrates that you are the source for the latest information about Microsoft Windows. When a new version of Windows is released, you will be notified and you will be given the allotted time to complete any new requirements to maintain current certification status.

You need to take and pass two exams to qualify as a Microsoft Windows operating system version 3.1 Certified Professional:

- Introduction to MS-DOS version 5.0 and Microcomputer Hardware
- Microsoft Windows operating system version 3.1 Administration

Upon successful completion of the Microsoft Windows Certified Professional requirements, you will receive a *Welcome Kit* that contains a Membership Card, Product Certification Certificate, *CompuServe Intro Kit*, product specific newsletter, a listing in the Microsoft Certified Professional Directory, and more.

To receive the latest Microsoft Certified Professional brochure containing information on certification requirements and preparing for exams for Windows or other Microsoft product certifications, call Microsoft Inside Sales at 1-800-227-4679. (Outside the U.S. contact your Microsoft subsidiary)

You can also receive it electronically as MCP.ZIP from Section 1 of the "MS Windows Advanced Users" Forum on CompuServe (GO WINADV).

The next page outlines the Microsoft Certified Professional Program Exam Topics for Windows.

Microsoft Windows Operating System Version 3.1 Administration Exam Topics

Installation and Setup

Windows installation: DOS mode and Windows mode events
 SETUP.INF: new default directory, creation of custom groups to house new applications
 Express and standard setup
 SYSTEM.INI
 WIN.INI
 CONTROL.INI
 PROGMAN.INI
 .GRP files

The Windows Files

WIN.COM: components, files executed when starting up in standard and 386 enhanced mode
 Windows core components
 DLLs
 Files required for Windows to run in standard mode
 Files required for Windows to run in 386 enhanced mode

PC Memory

Conventional, expanded, extended, and virtual memory
 HMA
 Expanded memory LIM3.2 and 4.0 specifications
 Loading MS-DOS into the HMA and advantages of MS-DOS 5
 UMB: characteristics, loading device drivers

Windows Memory Management

Standard and 386 enhanced mode: characteristics, minimum memory and hardware requirements
 Using conventional, expanded, extended, and virtual memory in standard and 386 enhanced mode
 LRU algorithm
 Temporary and permanent swap files
 HIMEM.SYS in Windows
 EMM386.EXE in Windows
 SMARTDRV.EXE in Windows
 Windows system resources

Windows Compound Documents, Fonts, and Printing

Typefaces and fonts
 Fixed and proportional spacing
 Windows screen fonts and printer fonts
 Raster and vector screen fonts
 TrueType fonts and standard fonts
 Character sets used by Windows screen fonts
 Types of Windows printer fonts
 Installing HPPCL soft fonts
 LPTx.OS2 port
 Supporting COM3 and COM4 with Windows
 Object Linking and Embedding and Dynamic Data Exchange

Non-Windows Applications and PIFs

PIFs and PIF Editor
 Levels of MS-DOS application support provided by Windows running in Standard mode and 386 Enhanced mode
 Creating a PIF for MS-DOS-based applications and TSRs requiring a given amount of conventional and extended memory
 Creating a PIF that will allow an application to run in a window while processing in the background
 _DEFAULT.PIF

Networks

Installing and sharing Windows on a network
 Setting up a Windows workstation with Windows installed locally
 Setting up a diskless workstation
 Locations of each of the key Windows files if running from an attached file server
 Advantages/disadvantages of setting up a network of Windows installed locally or from the file server
 Network persistence feature of Windows
 Winnet API
 Network drivers and virtual network driver

Microsoft University

Technical Training Straight from the Source

Microsoft University provides technical education and training on client-server architectures and Microsoft systems products to Information Systems Managers, Software Developers, and Systems Support Professionals.

Having provided technical training to over 20,000 Systems Support Professionals, Microsoft University understands that it's a challenge these days to implement new technologies while ensuring a smooth transition to new operating systems, environments, and applications. You've told us that the key to maximizing user productivity and minimizing system downtime is keeping existing and new systems running smoothly and efficiently during the transition, while sharpening your expertise to avoid problems with data integrity and user productivity.

MSU's Systems Support Professional curriculum provides technical training to address these real-world issues. MSU courses are in-depth, hands-on technical training to help you quickly apply these skills to your business solution, thus maximizing the return on your investment in Microsoft software.

Our Systems Support Professional courses focus on: tuning and optimization for maximum performance, customizing the user environment to enhance productivity, managing the interoperability issues between systems to ensure data integrity, and troubleshooting the system to quickly isolate, diagnose, and correct any problems.

To register, or for more information on Microsoft University education and training, call us at (206) 828-1507. MSU video courses are also available through resellers.

Solutions Oriented Training Options

Microsoft University offers *solutions-oriented* training options for all types of organizations. We recommend that you identify the end result you want from training and then evaluate each of the possible solutions listed below. These solutions can be used independently or in a combination that works best for your unique training needs.

Instructor-led courses at Microsoft University locations. MSU locations provide hands-on, lab-based training in state-of-the-art classrooms removed from office distractions. Our instructors have close ties with

Microsoft product development and they combine in-depth product knowledge with real-world business experience.

Instructor-led courses at an MSU Training Alliance member location. Many MSU courses may be taken from more than 75 Training Alliance members across the country. To be authorized, Training Alliance members and instructors must pass exactly the same rigorous standards of training and excellence in facilities as are required of MSU's facilities and instructors. MSU Training Alliance members use the identical course materials as those used at MSU locations and must maintain an excellent training record that is validated by positive student evaluations.

Instructor-led courses at your facility. This solution is flexible and cost-effective if you are training 12 or more people on one topic. Hosting a course on-site at your facility provides the confidentiality to openly discuss product strategy and design issues and minimizes travel expenses and time away from work. Virtually all MSU offerings can be delivered at your location.

Custom development and training. MSU designs and develops custom training solutions to meet your unique training requirements. Our existing courses can be modified or combined and can be enhanced with materials related to your environment. MSU will assist in any or all phases of custom course development, from instructional design and technical content evaluation to training instructors and producing classroom materials.

Video Courses. MSU's self-paced video courses are convenient, cost-effective resources that enable you to gain new knowledge and skills on-site and quickly bring new team members up to speed. Our courses integrate state-of-the-art video production and instructional design expertise to help you learn on your own and immediately apply what you've learned. Each video course includes a student reference guide and most include lab exercises for hands-on skill building.

Corporate Alliance Program. If you have more than 50 people to train, licensing MSU courses through our Corporate Alliance Program is a cost-effective solution. Virtually any MSU course can be licensed so that your own instructors may deliver the courses at your locations. The results are hands-on technical courses that save you the time and money spent developing and maintaining courses for your employees.

Systems Support Professional Curriculum

Microsoft University offers two courses specifically for support professionals who configure Windows for Workgroups systems and provide assistance to end users, the *Supporting Microsoft Windows for Workgroups* course and the *Moving to Microsoft Windows for Workgroups: Video Training for Support Professionals* video course. These advanced, hands-on technical courses will enable you to quickly become an expert on Windows for Workgroups, so you can assist your workgroup to fully maximize its many benefits! For more detailed information on our complete curriculum for systems support professionals, please call us at (206) 828-1507.

Supporting Microsoft Windows for Workgroups

Five days, \$1875 per person

This course provides students with the basic skills and knowledge to install, setup, implement, and support Microsoft Windows for Workgroups. The instructor led modules are reinforced with hands on labs so that students understand how to configure, customize, optimize, and troubleshoot the workgroup network, workgroup mail post office, and the shared workgroup scheduler. You will also understand how multiple Windows for Workgroups networks interoperate and how security is administered.

Course Outline

- Examining the Windows Operating System
- Memory Usage with Windows
- Optimizing Windows
- Customizing Windows Initialization Files and Setup
- Customizing MS-DOS-based Application Sessions
- Data and Application Sharing
- Registration Editor
- Fonts Used by Windows
- Troubleshooting Windows
- Setting Up Windows for Workgroups
- Windows for Workgroups Architecture
- Sharing Information Between PCs
- Securing the System
- Customization and Optimization
- Network Interoperability
- Workgroup Design and Implementation

Moving to Microsoft Windows for Workgroups: Video Training for Support Professionals

\$199 per video course; \$49 each additional set of student materials

This self-paced video course helps you quickly gain the skills needed to install, optimize, customize, and support Windows for Workgroups. Technical concepts and procedures are visually illustrated in the video through demonstrations and graphics. A student guide provides in-depth reference information, and lab exercises give you the opportunity to gain hands-on experience immediately. Upon course completion you will be able to automate the installation of Windows for Workgroups, optimize and customize systems, administer Mail and Schedule+, set up system security, and troubleshoot common problems encountered by end users. It takes about six to eight hours to complete this course.

Video Course Outline

- Getting Started
- Setting Up Windows for Workgroups
- Windows for Workgroups Architecture
- Workgroup Mail and Schedule+
- Sharing Information Between PCs
- Securing the System
- Customization and Optimization

- Network Interoperability
- Workgroup Design and Implementation

MSU's Systems Support Professional curriculum continues with the following courses:

Microsoft MS-DOS 5 Installation and Optimization: Video Training for Support Engineers and Advanced Users

\$149 per video course

Upgrading to Microsoft Windows 3.1: Video Training for Support Professionals

(Assumes prior experience supporting MS Windows 3.0)

\$199 per video course, \$49 each additional set of student materials

Supporting Microsoft Windows with Networks

(Formerly titled Microsoft Windows 3.1 for Support Professionals)

Five days, \$1875 per person

Administration of Microsoft Mail 3.0

Three days, \$1125 per person

Microsoft LAN Manager 2.1 for Network Administrators

Five days, \$1875 per person

Advanced Network Administration of Microsoft LAN Manager 2.1

Three days, \$1125 per person

Inter-Networking with Microsoft TCP/IP

Two days, \$750 per person

Introduction to Microsoft SQL Server

Two days, \$750 per person

Microsoft SQL Server 4.2 System Administration

Three days, \$1125 per person

Microsoft SQL Server 4.2 Implementation

Five days, \$1875 per person

Performance Tuning and Optimization of Microsoft SQL Server

Five days, \$1875 per person

EDUCATION AND TRAINING FOR OTHER KEY PEOPLE IN YOUR ORGANIZATION: INFORMATION SYSTEMS MANAGERS AND SOFTWARE DEVELOPERS

MSU also provides a comprehensive curriculum for Information Systems Managers and Software Developers. For more detailed information on these courses, please call us at (206) 828-1507.

Management Education Curriculum

Understanding New Technologies: A Video Seminar on the Building Blocks of Client-Server Architecture

\$495 per video set; additional sets of 10 Quick Reference Guides (glossary) for \$24.95 per set

Managing the Migration to Client-Server Computing

Three days, \$1195 per person

New Architectures for Enterprise Computing

Three days, \$1195 per person

Software Developer Curriculum

User-Interface Design for Microsoft Windows 3.1

One day, \$375 per person

Applications Development Using Microsoft Excel 4.0

Three days, \$1125 per person

Introduction to Programming for Microsoft Windows Using Microsoft Visual Basic

Three days, \$1125 per person

Programming in Microsoft Visual Basic 1.0

Four days, \$1500 per person

Building Applications for Microsoft SQL Server using Microsoft Visual Basic

Three days, \$1125 per person

Introduction to Programming in Microsoft C

Three days, \$1125 per person

Programming in Microsoft C

Five days, \$1875 per person

Programming in Microsoft C++

Five days, \$1875 per person

Programming for the Microsoft Windows Environment

(Formerly titled Microsoft Windows Programming Environment)

Five days, \$1875 per person

Fundamentals of Microsoft Windows Programming: Video Training for Developers

\$1595 per video course, \$249 each additional set of student materials

Exploring Controls: Video Training for Developers Using the Microsoft Windows Environment

\$199 per video course, \$49 each additional set of student materials

Programming Applications for Microsoft Windows 3.1

(Formerly titled Microsoft Windows Applications Programming)

Four Days, \$1500 per person

Microsoft Win32 Developers Conference Videotapes

\$999 per videotape set

Introduction to Microsoft SQL Server

Two days, \$750 per person

Microsoft SQL Server 4.2 Implementation

Five days, \$1875 per person

Programming Microsoft SQL Server in a Client-Server Environment

(Formerly titled Microsoft Transact-SQL)

Three days, \$1125 per person

Microsoft DB-Library

Two days, \$750 per person

Performance Tuning and Optimization of Microsoft SQL Server

Five days, \$1875 per person

**MICROSOFT UNIVERSITY AND
MSU TRAINING ALLIANCE
LOCATIONS**

The following is a listing of MSU regional locations and MSU Training Alliance member locations in North America. New members are continually joining our Training Alliance Program. Call this number or an updated listing of member locations in your area:
(206) 828-1507

EAST

MICROSOFT UNIVERSITY
New York, New York
(212) 484-1803

MICROSOFT UNIVERSITY
Waltham, Massachusetts
(617) 487-6656

MICROSOFT UNIVERSITY
Washington, DC
(202) 364-7010

CGI SYSTEMS, INC.
New York, NY
(800) 969-1980

CGI SYSTEMS, INC.
Washington, DC
(800) 969-1980

DESAI CONSULTING GROUP
Hartford, CT
(203) 233-0011

DIGITAL EQUIPMENT
CORPORATION
Boston, MA
(800) 332-5656

HI-TECH RESOURCES, INC.
Boston, MA
(800) 882-6420

INFORMATION
TECHNOLOGISTS, INC.
Philadelphia, PA
(215) 832-1000

JWP/INFORMATION
SERVICES
Boston, MA
(617) 575-8283

JWP/INFORMATION
SERVICES
New York, NY
(212) 564-3000

JWP/INFORMATION
SERVICES
Washington, DC
(202) 429-1922

LOGICAL OPERATIONS
Rochester, NY
(716) 482-7700

MICROAGE OF EXTON
Exton, PA
(215) 524-6665

MICROAGE OF EXTON
Philadelphia, PA
(215) 524-6665

NETWORK MANAGEMENT,
INC.
New York, NY
(703) 218-5221

NETWORK MANAGEMENT,
INC.
Rosslyn, VA
(703) 218-5221

VALINOR, INC.
Manchester, NH
(603) 668-1776

SOUTH

MICROSOFT UNIVERSITY
Atlanta, Georgia
(404) 392-7519

CEDALION SYSTEMS, INC.
Charlotte, NC
(800) 277-4LAN

CEDALION SYSTEMS, INC.
Research Training Park, NC
(800) 277-4LAN

COMPUTER LEARNING
CENTER
Knoxville, TN
(615) 691-1515

COMPUTER SAVVY, INC.
Ft. Lauderdale, FL
(305) 486-0644

DANKA COMPUTER
SOLUTIONS
Kansas City, MO
(816) 221-2900

DIGITAL EQUIPMENT
CORPORATION
Atlanta, GA
(800) 332-5656

HI-TECH RESOURCES, INC.
Atlanta, GA
(800) 882-6420

HI-TECH RESOURCES, INC.
Rockville, MD
(800) 882-6420

NETWORK EDUCATION
CENTER
Tampa, FL
(813) 282-0380

WORDLINK
St. Louis, MO
(314) 878-1422

SOUTHWEST

MICROSOFT UNIVERSITY
Dallas, Texas
(214) 716-2678

BSG EDUCATIONAL
SERVICES
Houston, TX
(800) 937-2001

COMPUCOM EDUCATION
CENTER
Dallas, TX
(800) 783-7894

DATAAIDS
Houston, TX
(713) 956-7200

EDUTECH, INC.
Dallas, TX
(214) 733-5555

EDUTECH, INC.
Houston, TX
(713) 827-1314

INTELOGIC TRACE
San Antonio, TX
(512) 593-5770

MIDWEST

MICROSOFT UNIVERSITY
Oakbrook Terrace, Illinois
(708) 628-4178

APPLIED COMPUTER
SERVICES, INC.
Chicago, IL
(708) 614-4900

BABBAGE-SIMMEL &
ASSOCIATES, INC.
Columbus, OH
(800) 288-5937

BENCHMARK COMPUTER
SYSTEMS
Minneapolis, MN
(612) 830-9920

CHECKPOINT, INC.
Chicago, IL
(312) 441-1220

CONNECT COMPUTER
COMPANY
Minneapolis-St. Paul, MN
(612) 944-0181

FOURTH WAVE
TECHNOLOGIES,
INC./USCONNECT
Troy, MI
(313) 362-2288

JWP/INFORMATION
SERVICES
Chicago, IL
(708) 617-4408

JWP/INFORMATION
SERVICES
Minneapolis, MN
(612) 557-7555

LANMIND, INC.
Chicago, IL
(312) 935-9900

NETWORK COMPATIBILITY
GROUP, INC.
Columbus, OH
(614) 436-2962

SYSTEMS EVALUATION &
ANALYSIS GROUP, INC.
Springfield, IL
(217) 523-1717

WORDLINK
Champaign, IL
(217) 359-9378

WORDLINK
Chicago, IL
(708) 397-4000

WORDLINK
Schaumburg, IL
(708) 397-4000

WEST

MICROSOFT UNIVERSITY
Bellevue, Washington
(206) 828-1507

MICROSOFT UNIVERSITY
Foster City, California
(415) 573-5090

MICROSOFT UNIVERSITY
Santa Monica, California
(310) 449-7400

AMERICAN DIGITAL
TECHNOLOGIES, INC.
Costa Mesa, CA
(714) 433-1300

DIGITAL EQUIPMENT
CORPORATION
Denver, CO

DIGITAL EQUIPMENT
CORPORATION
Santa Clara, CA
(800) 332-5656

EXPERT TECHNOLOGY
CORPORATION
Portland, OR
(503) 248-9918

HI-TECH RESOURCES, INC.
Los Angeles, CA
(800) 882-6420

INFOTEC DEVELOPMENT,
INC.
Los Angeles, CA
(800) 282-7990

INFOTEC DEVELOPMENT,
INC.
San Diego, CA
(800) 282-7990

INFOTEC DEVELOPMENT,
INC.
San Francisco, CA
(800) 282-7990

INFOTEC DEVELOPMENT,
INC.
Santa Ana, CA
(800) 282-7990

INFOTEC DEVELOPMENT,
INC.
Santa Clara, CA
(800) 282-7990

INNOVATIVE SOLUTIONS,
INC.
Sacramento, CA
(916) 928-1700

KENNEDY & ASSOCIATES
San Diego, CA
(619) 483-1692

LEARNSOFT
San Diego, CA
(619) 546-1400

NETBASE COMPUTING, INC.
San Francisco, CA
(310) 214-8181

NETBASE COMPUTING, INC.
Torrance, CA
(310) 214-8181

NEW HORIZONS COMPUTER
LEARNING CENTER
Santa Ana, CA
(303) 773-8283

THE REDMOND ALLIANCE,
INC.
Albuquerque, NM
(800) 326-1044

THE REDMOND ALLIANCE,
INC.
Huntington Beach, CA
(800) 326-1044

THE REDMOND ALLIANCE,
INC.
Los Angeles, CA
(800) 326-1044

SEMIOTIX
Aurora, CO
(303) 743-1400

SOFTEACH CORPORATION
Denver, CO
(303) 759-1127

SQLSOFT
Kirkland, WA
(206) 822-1287

TOMPKINS GROUP
Vista, CA
(619) 941-7294

WORDLINK
San Francisco, CA
(415) 392-5465

CANADA

MICROSOFT UNIVERSITY
Toronto, Ontario
(416) 568-0343, Ext. 4134

MEXICO

ASESORIA TECNOLÓGICA
EN REDES
Mexico D.F.
(525) 669-39-94 or 669-38-37

INTEGRACION
TECNOLOGICA VERTICAL
Mexico D.F.
(525) 687-63-47 or 543-10-57

To register, or for more
information on Microsoft
education and training, call us at
(206) 828-1507. MSU video
courses are also available through
resellers.

Online Services

The following are numbers for popular BBSs. These numbers will help you log on to the service from any city in the country. Listed below each BBS are the appropriate forums from which information or support for Windows for Workgroups is available.

Microsoft
Product Support Services Download Service (MSDL)
(206) 936-MSDL.

CompuServe
1-800-848-8990

GE^{nie}TM
1-800-638-9636

Windows Driver Library

The Windows Driver Library is for all users of Windows who own peripheral devices not supported in the Windows or Windows for Workgroups retail package.

The WDL will include the Windows 3.1 compatible device drivers that are not included in the Windows 3.1 or Windows for Workgroups 3.1 retail box, including Printer, Display, Network, Fastdisk and VDD and Grabbers for the display drivers. All drivers in the WDL are compatibility tested by Microsoft. The WDL will be updated regularly as new and improved drivers become available.

The WDL will be distributed on network services including CompuServe, GE^{nie}, ONLine, APCUG, and Microsoft PSS Download Service. The WDL is free

of charge when downloaded from these services. The WDL is also available on disk for a fee from Microsoft Customer Service at 1-800-426-9400 (Outside the U.S., contact your Microsoft subsidiary.)

The Windows Technical Support Library

This service will help users who are either PC support professionals or who support Windows for Workgroups users in their organization. This Library provides better and faster access to the information needed to solve Windows for Workgroups technical support problems. It will also help you update and customize the Windows for Workgroups Resource Kit with up-to-date technical information related to specific applications, hardware devices, and networks.

The Windows for Workgroups Technical Support Library will be available by referencing the appropriate forums on network services such as GE^{nie} and CompuServe or free of charge by calling Microsoft PSS Download Service. To have information faxed or mailed, call Microsoft Product Support Services.

Microsoft Developer Network

Join the Microsoft Developer Network to tap into the definitive source of programming information for Microsoft Windows.

The Microsoft Developer Network is a club for all developers who write applications for the Microsoft Windows operating system or who use Microsoft tools for development purposes. The Microsoft Developer Network has two main goals: to write and publish information on programming for Windows, and to establish two-way communication with the development community. By joining the Microsoft Developer Network, you will become a registered developer with Microsoft and receive technical and strategic information through three channels: the Microsoft Developer Network CD, the *Microsoft Developer Network News*, and the Microsoft Developer Network Forum on CompuServe.

The Developer Network CD

The Microsoft Developer Network CD is a comprehensive source of information for developers of Windows-based applications. The CD provides new, in-depth articles on programming created in response to developers' inquiries.

To create the content for the Developer Network CD, we

assembled a team of programmers, each experienced in specific areas of Windows (GDI, User, Kernel, Win32™, multimedia, and so on), and asked them to document the architecture of Windows, addressing known areas of complexity. The first CD contains over 43,000 pages of programming information, including:

Dozens of newly created technical articles containing in-depth explanations of specific programming topics

- Complete documentation sets for Microsoft development products
- *Microsoft Systems Journal* articles
- Developer Knowledge Base articles from Microsoft Product Support Services
- Sample code, tools, and utilities
- Microsoft Press® books, such as *Programming Windows* by Charles Petzold
- White papers and specifications

The Developer Network CD uses a powerful, Windows-based search engine that lets you search the contents of the entire CD by source, by subject, or by keyword. You can create precise queries, print topics, and import source code into your application.

The Newspaper

The *Microsoft Developer Network News* is a quarterly newspaper with helpful, timely information for all programmers. The newspaper provides the latest news on Microsoft development tools, operating systems, and Windows functions, as

well as programming tips, strategic information, and key Microsoft phone numbers.

The Forum

The Microsoft Developer Network posts all new technical articles and sample code on CompuServe, in the Microsoft Developer Network Forum. You can access this area by typing GO MSDNLB at the CompuServe ! prompt. You can communicate with the Microsoft Developer Network through forum messages and electronic mail.

Enrollment Information

For more information, call the Microsoft Developer Services Team at 1-800-227-4679, ext. 11771, between 6:30 A.M. and 5:30 P.M. Pacific time. Outside the U.S., contact your Microsoft subsidiary or call (206) 936-8661 for local contact information.

Mail: Microsoft Developer Network
One Microsoft Way
Redmond, WA 98052-6399

(206) 936-7329, Attn: Developer Network

Internet: devnetwk@Microsoft.com

CompuServe:

>INTERNET:devnetwk@Microsoft.com



Microsoft.
Developer
Network

Microsoft Mail Configuration Guide

This appendix contains information that can be used to identify components that are necessary to integrate Microsoft Mail into multiple postoffice or heterogeneous mail environment. The *Microsoft Mail and Schedule+ Extensions for Windows for Workgroups* product upgrades a Windows for Workgroups postoffice to offer the full capabilities available with the *Microsoft Mail version 3.x for PC Networks* product. It is not necessary to modify any of the client software when upgrading to a Microsoft Mail 3.x postoffice. Once a Windows for Workgroups postoffice is upgraded, the Microsoft Mail gateways can be used to communicate with other mail systems. This appendix describes the connectivity options that are available and identifies the components that are necessary for using Microsoft Mail to communicate with other mail systems.

For more about the connectivity options available for use with Microsoft Mail, or the name of an authorized reseller, call Microsoft Customer Service at 1-800-227-4679 (outside the U.S., contact your Microsoft subsidiary).

Note *Microsoft Mail and Schedule+ Extensions for Windows for Workgroups* must first be used to upgrade your Windows for Workgroups postoffice to a Microsoft Mail 3.x postoffice before connectivity to other postoffices, connecting to remote users, or the gateways listed in this appendix can be used.

Related information

- *Windows for Workgroups Resource Kit: Chapter 12, "Mail;" Chapter 13, "Schedule+."*

Contents of This Configuration Guide

Introduction.....	B-3
How to Use This Appendix.....	B-7
Master Connectivity Map	B-10
One Postoffice, No External, Multiple Mail Clients	B-13
One LAN with One Postoffice and Remote Users.....	B-15
One LAN with One FAX Gateway.....	B-17
One LAN with Multiple Postoffice Configuration	B-19
Microsoft Mail Connection.....	B-20
Multiple LANs Joined Asynchronously.....	B-22
Hub Postoffice for WANs	B-23

Hub Postoffice for Gateway Access	B-24
Wide-Area Networking Through a Gateway	B-26
X.400 Gateway	B-28
X.400 Gateway to AT&T or Other ADMD	B-30
X.400 Gateway Backbone	B-32
Gateway to PROFS	B-34
PROFS Distribution Manager	B-36
PROFS Gateway Backboning	B-38
SNADS Gateway	B-40
SMTP Gateway	B-42
MHS Gateway	B-44
MCI Gateway	B-46
3+Mail Gateway	B-48
Configuration Guide Glossary	B-49

Introduction

To better understand Microsoft Mail for PC Networks, you should first become acquainted with the key components of the product. These components are included in both the *Microsoft Mail and Schedule+ Extensions for Windows for Workgroups* and the *Microsoft Mail Server* packages.

Microsoft Mail 3.x was designed to establish messaging as a transport mechanism that operates as an enabling technology for other applications, rather than simply as an end in itself. As a first step, Microsoft designed the Microsoft Mail Windows client to be transport-independent from the server. This provides flexibility to combine core messaging modules from a variety of vendors.

The flexibility of the Windows client makes all the existing server technology at the customer's site available on each user's desktop and fully-integrates that technology with Windows-based applications. This is accomplished through messaging drivers that enable the Windows client to serve as the "front-end" for the existing server products, in which the customer is likely to have a considerable corporate investment.

In addition to the new Windows client, in this release of Microsoft Mail we have enhanced our Mail server technology and our full suite of gateways. We have made it even easier for administrators to manage postoffices, directories, and gateways.

The Client Architecture

Along with the Windows client, Microsoft Mail provides three other interfaces—MS-DOS, Presentation Manager, and Apple Macintosh. All three currently run as mail clients for the Microsoft Mail server. All clients are interoperable with each other, so customers can install any or all of the interfaces they choose. All four clients are packaged together in the Microsoft Mail Server product and because users are licensed per workstation without respect to client type, customers retain the flexibility to change mail clients as their operating environments evolve.

All clients share a number of unique features. For example, they all have access to shared folders, personal address books, and private folders.

The Server Architecture

The Microsoft Mail database is known as the *postoffice*. Each postoffice supports up to 500 users, and an organization can start with a single postoffice and expand whenever needed. When linked by the External program, many postoffices can run simultaneously on a single LAN. The postoffice and its four clients can coexist with

many other postoffices on the network and can communicate with other mail systems, both within the organization and throughout the world.

Gateway Architecture

Gateways link Microsoft Mail with other mail systems. Microsoft Mail offers the most extensive array of LAN-based gateways. They include an IBM PROFS/OV gateway, an IBM SNADS gateway, an IBM Distribution Manager gateway, an X.400 gateway, SMTP gateway, an MHS gateway, an MCI gateway, a FAX gateway, and a 3+Mail gateway.

These gateways enable Microsoft mail users to send and receive mail from foreign mail systems. For example, the Microsoft Mail X.400 gateway enables users on any Microsoft Mail postoffice to exchange messages with users on any other X.400 system, whether they use a mainframe, minicomputer, or PC-based system. By using the Microsoft Mail Global Address Lists, the differences among the systems are transparent to the users, who chooses any gateway address (including FAX, SMTP, SNADS, and PROFS) just as they choose any other address within the local postoffice. The Microsoft Mail X.400 gateway is the best-selling X.400 gateway in the world. In fact, Microsoft Mail for PC Networks is the only LAN-based electronic mail system that offers its own complete X.400 gateway. All others must rely on third-party solutions.

As you will see in several of scenarios in this appendix, foreign electronic mail systems can act as hubs for Microsoft Mail because of the powerful “message encapsulation” capabilities. Messages with complex body types (for example, OLE object in the message body) can be sent from one Microsoft Mail user to another through a gateway without losing any of their richness.

Each gateway (with the exception of the PROFS gateway) has two components. The first is the gateway itself, which converts the proprietary format of PC Mail to the format of the foreign mail system. The second is the Access component, which is the addressing template that allows PC Mail users to address mail in the format of the foreign mail system. Both components are required for the gateway Postoffice, but only the Access component is required for the downstream postoffices.

A unique feature of the Microsoft Mail architecture is that both the gateways and the mail system use the same administration program.

Address Lists and Routing

A Microsoft Mail address consists of a mailbox name, a postoffice name, and a network name. This three-level addressing scheme makes it possible to use duplicate mailbox names on different postoffices, and duplicate postoffice names on different networks. This enables each department or site within an organization to adopt meaningful naming conventions that offer direct relevance to people and

places within the organization rather than obscure codes mandated from above. The importance of this addressing scheme is even more apparent when you consider that Microsoft Mail is independent of the mail transport system and exchanges messages with numerous mail systems, each with its own naming conventions.

The routing information assigned to each user is established when the user is added to their local postoffice. This logical address is attached to the user name and is associated with all the lists the user is attached to, such as groups and aliases. If the user's routing information changes, the address is modified and all the other related lists and groups automatically reflect the change.

One important feature that distinguishes Microsoft Mail from its competitors is the use of custom directory templates. Administrators can create custom templates for each user that can include information such as telephone numbers, department number, and so on, and when users select "details" from the address menu, this information will appear.

The Global Address List

The structure of the Microsoft Mail directory is as flexible as you need it to be. To make all mail addresses available from a single list, all the postoffice and gateway address lists can be combined in a single entity known as the Global Address List (GAL). Each address list can also be segregated and appear as a separate list, or the administrator can choose a combination of the two and designate specific address lists for the Global Address List.

The GAL consists of a flat file structure which provides the ability to search for key character strings. This design enables very fast address retrieval, which has been tested on lists consisting of over a half-million names.

The GAL is available as the primary address list in the Windows, MS-DOS, Macintosh and Presentation Manager clients or users can choose the Personal Address List (PAL) as the default.

The Global Address List includes a feature that automatically rebuilds the list whenever directory synchronization occurs. This means that any changes in individual postoffices are automatically reflected in the GAL.

Directory Synchronization

The most time consuming aspect of updating ever-changing directory information is incorporating changes and passing them to all the servers on the system. The old, inefficient way is to transmit the entire directory to every server every time there is a change. This is like exchanging telephone directories on a yearly basis, where the

entire printed directory is distributed, and users are encouraged to replace the obsolete directory.

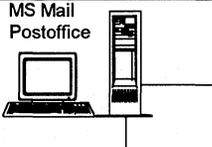
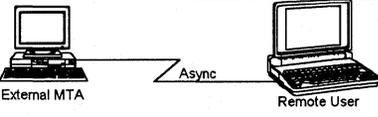
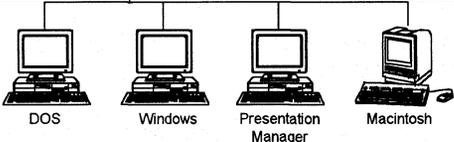
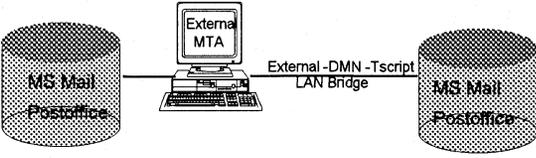
Microsoft Mail avoids this time-consuming problem with directory synchronization, which automatically updates all the postoffices with only the changes and enhances, rather than replaces, the existing GAL. Directory synchronization is accomplished by designating each postoffice as a server or requester of directory synchronization information. Each postoffice keeps a record of the changes that are made to its directory during the day, and then at a time specified by the system administrator, these changes are sent as a mail message to its designated server postoffice. Because only the changes are sent, this information is processed quickly and consolidated by the server postoffice, which in turn sends out messages containing the changes to all its requester postoffices. The requester postoffices automatically incorporate the changes in their existing Global Address Lists. A fault tolerant mechanism ensures that all updates are received and tracked through the system; any distribution errors are automatically reported to the server postoffice. Other networks and mail systems can participate in the directory synchronization process by sending and receiving directory synchronization messages in the same format as Microsoft Mail.

This section introduced the features that form a robust framework for Microsoft Mail. Now with the basic structure of the product established, we can move on to detailed configuration examples.

How to Use This Appendix

This appendix consists of an introduction to the Microsoft Mail 3.x architecture, followed by 20 sections that discuss possible configurations.

The following symbols are used in most of the illustrations contained in this appendix. The text next to the illustration describes how the symbols are used in the illustrations.

 <p>MS Mail Postoffice</p>	<p>This symbol represents the postoffice database. To simplify some of the illustrations, it is shown without clients attached; however, clients are always part of any configuration that includes the postoffice database.</p>
 <p>External MTA Async Remote User</p>	<p>This illustration shows the External MTA connected to a remote user. Links between postoffices can be any connection unless otherwise specified.</p>
 <p>DOS Windows Presentation Manager Macintosh</p>	<p>The client symbols are represented by workstation symbols on a LAN.</p>
 <p>MS Mail Postoffice External MTA External -DMN -Tscript LAN Bridge MS Mail Postoffice</p>	<p>Common External Command line options:</p> <ul style="list-style-type: none"> • D is the drive designation of the postoffices. • T is used to specify a script file. • H specifies the home postoffice drive option for dynamic drive administration. • W is the Wide-area network Drive option.

Ordering Information

Each configuration diagram requires at a minimum the Server component for Microsoft Mail. The Server component is in both the **Mail and Schedule+ Extensions for Windows for Workgroups** and in the **Microsoft Mail Server** packages. This configuration guide will focus on the components needed if you are using a Windows for Workgroups postoffice. Additional Mail users can be added by ordering either additional Windows for Workgroups packages or ordering **Microsoft Add Paks (MAPs)**. The Microsoft Mail part numbers and ordering information are described as follows:

- 068-499V300 **Microsoft Mail and Schedule+ Extensions for Windows for Workgroups** (also called **Extensions**; includes single postoffice, External, dial-in support, user agent software, TSRs, utilities, and Schedule+ administration and user agent software)
- 068-099V300 **Microsoft Mail Server** (includes single postoffice, External, dial-in support, user agent software, TSRs, utilities, and a license for 10 users)

These server packages includes user agent software for MS-DOS, Windows, Presentation Manager, and Apple Macintosh.

To add users, order either Windows for Workgroups or one or more of the following **Microsoft Mail Add Paks (MAPs)**:

- 068-399V300 5 Workstation MAP
- 0684099V300 20 Workstation MAP
- 0684199V300 100 Workstation MAP
- 0684299V300 500 Workstation MAP

The purchase of one or more MAP licenses the customer to support a maximum user count. These users may use any mix of client interfaces. The license is not tied to a specific postoffice and, in multiple postoffice configurations, the users included in the count may be spread over several postoffices. Up to 500 users can be placed on a file server; however, disk space and the number of simultaneous network connections may also be limiting factors. Refer to your LAN Administrator's manual.

Hardware Information

Microsoft Mail for PC Networks is a network application that runs on Windows for Workgroups, LAN Manager, IBM, Novell, 3Com, Banyan, or other PC-oriented file-sharing networks. As long as a user's workstation (DOS, Windows, OS/2 PM, or MAC) can redirect a drive to a shared file, he or she can use PC Mail. Unless otherwise stated, the LAN is the only hardware that is required to run PC Mail.

As previously stated, the Windows client for PC Mail is transport-independent. It does not in fact require a network but can run as an independent client for other Mail transports. This appendix doesn't cover these instances.

Master Connectivity Map

The following table identifies the Microsoft Mail gateways that can be used to integrate Mail in a heterogeneous mail environment. By using one of the gateways available from Microsoft, Mail can communicate with a variety of mail systems available throughout the world.

The table includes these columns:

Mail System—identifies the mail system that Microsoft Mail can communicate with.

Gateways Available to Access—identifies the Microsoft Mail gateways that can be used to communicate with the mail environment identified in the Mail System column.

Message & File Support—indicates whether messages and file attachments can be sent through the Microsoft Mail gateways for communicate with the Mail System.

Addressing Forms—indicates whether an integrated addressing form is available to facilitate the creation of an address to send a mail message to a mail system user through the gateway.

Directory Exchange—identifies whether Microsoft Mail can exchange and synchronize directory information with the mail system through the Microsoft Mail gateway.

Gateway for Microsoft Mail—indicates whether Microsoft Mail can use the identified gateway to enable two Microsoft Mail postoffices to communicate indirectly by passing messages through the associated gateway. This provides a backboning capability through the use of an intermediary communication mechanism such as an X.400 service.

Mail System	Gateways Available to Access	Message & File Support	Addressing Forms	Directory Exchange	Gateway for Microsoft Mail
3+Mail®	PC Mail 3+Mail	●	●		
ARCOM 400 Swiss PTT	X.400	●	●		●
AT&T Mail	X.400	●	●		●
Atlas 400	X.400	●	●		●
Atlas-Minitel	by Atlas 400 Services	●	●		
Banyan Mail	MHS	●	●		●
	X.400	●	●		●
Beyond Mail	MHS	●	●		●
cc:Mail™	MHS (both ends)	●	●		
	SNADS+ Soft-Switch	●	●		
	X.400 (MS) X.400 Server (Retix/Touch)	●	●		
Coordinator	MHS	●	●		●
CompuServe	SMTP	●	●		
	MHS	●	●		●
Data General® CEO	X.400	●	●		●

(continued)

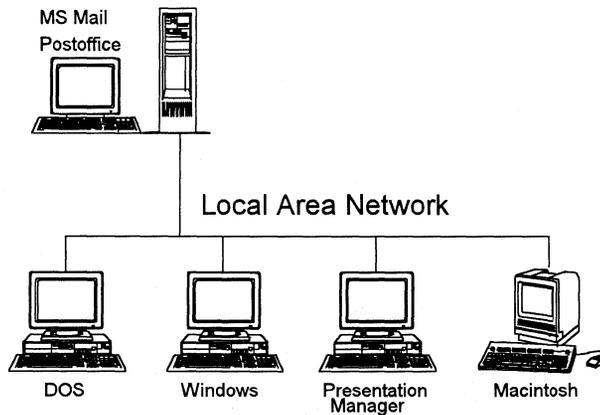
Mail System	Gateways Available to Access	Message & File Support	Addressing Forms	Directory Exchange	Gateway for Microsoft Mail
DaVinci™	MHS	●	●		●
DEC All in 1	X.400 + DEC Message Router X.400	●	●		
	PROFS+Dist. Mgr., and DEC's MRP	●	●		●
	SNADS+ Soft-Switch™	●	●	●	●
DEC VMS®	X.400 + DEC Message Router X.400	●	●		●
	SNADS+ Soft-Switch	●	●	●	●
Dutch PTT	X.400	●	●		●
Easylink®	X.400 (available 1Q93)	●	●		●
Envoy 100/ Gemdes	X.400	●	●		●
FAX	Fax	●	●		
Fischer Int'l EMC2	X.400	●	●		●
	SNADS	●	●		●
GEIS	QuickComm gateway from GE	●	●		
Gold 400 UK	X.400	●	●		●
Higgins	MHS	●	●		●
HP Desk	Walker, Richer, Quinn HP Desk Gateway	●	●		
IBM AS/400® Office	SNADS	●	●	●	●
IBM System 36	PROFS + Dist. Manager	●	●	●	●
IBM CMS NOTES	PROFS	●	●		●
IBM DISOSS	SNADS/DISOSS	●	●		●
IBM OfficeVision™ MVS	SNADS	●	●		●
	PROFS Dist. Mgr.	●	●	●	●
IBM PROFS® IBM Office Vision/VM	PROFS	●	●	●	●
	SNADS+ Soft-Switch	●	●	●	●
INFONET	X.400	●	●		●
Internet	SMTP	●	●		●
Lotus Notes®	MHS (both ends)	●	●		
MCI FAX	MCI MAIL ®	●	●		
MCI Telex	MCI MAIL	●	●		
MCI Paper Mail	MCI MAIL	●	●		

(continued)

Mail System	Gateways Available to Access	Message & File Support	Addressing Forms	Directory Exchange	Gateway for Microsoft Mail
MCI Mail	MCI	●	●		
	X.400	●	●		●
Microsoft Mail for AppleTalk	Microsoft Mail Connection	●	●	●	
Microsoft Mail to Microsoft Mail	LAN Connection, Async, dial-up X.25 via PC Mail's MTA/External Program	●	●	●	●
NCR—Cooperation	X.400	●	●		●
Novell MHS	MHS	●	●		●
Retix®	X.400	●	●		
Softswitch Central	SNADS	●	●	●	●
	X.400	●	●		●
Sprint Teleelectronic mail	X.400	●	●		●
Standalone users	PC Mail Remote User	●	●	●	●
Telebox 400 German PTT	X.400	●	●		●
Telex	from Management Software Ltd. - UK	●	●		
Touch	X.400	●	●		
Unisys	X.400	●	●		●
UNIX/SMTP	SMTP	●	●		●
UUCP	SMTP via UUCP host	●	●		
Verimation Memo	SNADS	●	●		●
	SNADS+ Soft-Switch	●	●		●
WANG® Office	MHS - Lightspeed	●	●		
	X.400 - Wang X.400	●	●		
	PROFS+Dist. Mgr.	●	●	●	●
X.25	X.400	●	●		●

Note The Mail 3.x Directory Synchronization engine permits synchronization with any of the foreign systems listed by writing an agent, which may reside either on the External PC or the host system.

One Postoffice, No External, Multiple Mail Clients



Configuration

This configuration, found typically in small businesses, is a single LAN with no connections to other networks and no remote users. The electronic mail system is used for internal office communications, but there is no need to exchange mail with other postoffices or to support dial-in access.

Windows for Workgroups is the product that supports this configuration if the LAN includes only Windows and MS-DOS clients. If Macintosh or OS/2 clients are necessary, the customer will need to purchase either **Extensions** or **Microsoft Mail for PC Networks Server**.

Extensions is ideal for Windows for Workgroups upgrades because it enhances the Workgroup postoffice (the server package installs a new postoffice) and includes the full Schedule+ product (which includes more features than the Windows for Workgroups Schedule+ software).

PC versus Macintosh Workstations

Workstations for this configuration and all other in this appendix may be either IBM-compatible or Macintosh computers.

For concentrations of Macintoshes, link them with AppleTalk and use **Microsoft Mail for AppleTalk Networks**.

For isolated Macintoshes on a PC LAN, use the Macintosh user agent to **Microsoft Mail for PC Networks**. The Macintosh user agent, together with user agents for MS-DOS, Windows, and OS/2 Presentation Manager, are all sold as part of the **Microsoft Mail Server package** and can coexist and interoperate with each other.

Customers can install any or all of the user interfaces and retain the flexibility to change user agents as their operating environments evolve.

Microsoft Mail Server automatically includes a license for 10 users. You can also purchase the **Microsoft Add Pak** can be purchased in quantities of five, 20, 100, and 500 users. The **Microsoft Add Pak** is a license to copy existing user agent software provided with **Microsoft Mail Server** and does not include software.

For ordering and hardware requirements information, see the preceding “Ordering Information” and “Hardware Information” sections.

Practical Limitations

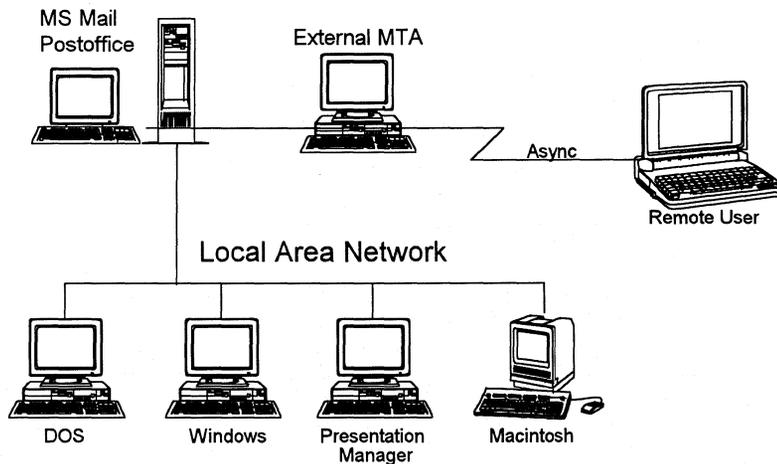
The maximum number of users per postoffice is 500, but depends greatly on volume of traffic, stored mail, and the size of the server.

If all the client databases are to be stored on the server including the Windows client mail database the recommended server disk-space per user is 10 MB.

If the Windows client mail database is to be stored on the local drive the recommended server disk-space per user is 3 MB.

These figures may vary, depending on the way specific workgroups use mail.

One LAN with One Postoffice and Remote Users



Configuration

This configuration is for a small business with employees who dial in for messages. For this configuration, you'll need to install **Microsoft Mail External (External)** on a dedicated PC for remote dial-in support. (**External** is included as part of the Server package.) You'll also need a copy of **Microsoft Mail Remote** for each remote workstation requiring dial-in access. Note that these remote users should not be included in the **Microsoft Add Pak (MAP)** user count unless they also access mail locally.

The **Microsoft Mail Remote** software runs under MS-DOS ; however, workstations on the PC LAN may be IBM-compatibles running MS-DOS, Windows, or OS/2 Presentation Manager, or they may be Macintoshes running in the native Macintosh environment. (A Windows version of **Microsoft Mail Remote** will be available in first quarter of 1993.)

Order

- 068-499V300 Microsoft Mail and Schedule+ Extensions for Windows for Workgroups (includes External)
- 068-299V300 Remote Workstation; one copy per remote workstation
- Microsoft Add Paks as needed

Hardware

- A dedicated AT-compatible computer with 512K RAM is required to run the **External Mail** program; MS-DOS or PC-DOS 3.1 or later; 2400-baud or faster modem
- PC class machine (8088 or later) to run the **Remote Workstation** software
- Modem or internal modem board for each external connection

Practical Limitations

The limitations for “One Postoffice, No External, Multiple Mail Clients” apply also for this configuration.

Multiple remote workstations can dial a single External MTA machine. One External MTA machine can route mail for remote users on different postoffices.

Modem speeds of 1200, 2400 and 9600 baud are supported. A customizable script file is included for support of unique modems and different modem speeds. An IBM-AT compatible should be used for modem speeds greater than 2400 baud.

One LAN with One FAX Gateway



Configuration

For a workgroup that makes extensive use of fax facilities, add **FAX Gateway** to the LAN. This enables any Microsoft Mail user to send faxes directly from their workstation. **FAX Gateway** allows transmission of mail messages, text documents, and bit-mapped graphics directly, without the need to produce printouts to feed through a fax machine. **FAX Gateway** checks the gateway postoffice for outgoing faxes, initiates calls, and sends faxes.

FAX Gateway also delivers incoming faxes into the electronic mail system, so they can be routed directly into a user's inbox without the need to print them. By installing the **FAX Access** product on a downstream postoffice, users with mailboxes on that postoffice can send faxes as easily as those on the gateway postoffice. The faxes are routed through the gateway postoffice where they are serviced by **FAX Gateway**.

Order

- 068-499V300 Microsoft Mail and Schedule+ Extensions for Windows for Workgroups (includes External)
- 068-099-120 FAX Gateway version 3.x(includes Access component for gateway postoffice)
- 068-099-121 FAX Access version 3.x(one copy per downstream postoffice)
- Microsoft Add Paks as needed

Hardware

- A dedicated IBM-compatible computer with 512K RAM to run **External Mail** and support the modem connection; MS-DOS or PC-DOS 3.1 or later.
- A dedicated IBM-compatible computer with 512KB RAM and MS-DOS 3.3 or later to run **FAX Gateway**
- Intel Connection coprocessor and software, Intel SatisFAXion® board and software, GammaFAX board or other CAS-compatible FAX board and software
- LAN network adapter and redirection software

Services

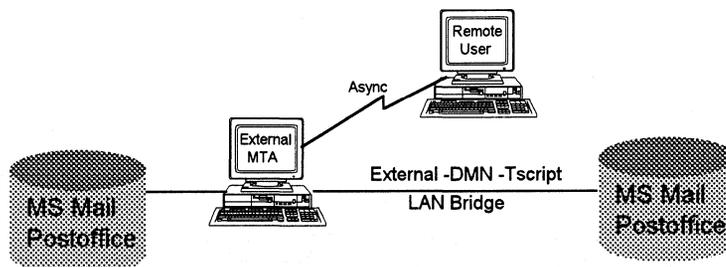
- Business telephone line to connect to the **FAX Gateway computer**

Practical Limitations

The limitations for “One Postoffice, No External, Multiple Mail Clients” apply also for this configuration.

The theoretical limit to the number of postoffices **External** can connect is 50 postoffices; the practical limit is 15 postoffices.

One LAN with Multiple Postoffice Configuration



Configuration

Where a customer has a LAN with multiple file servers, one or more postoffices can be configured on each server. Install **Microsoft Mail Server** for each postoffice. You'll also need to install **External** on a dedicated PC, to route mail between postoffices.

In the diagram above, the External Mail Transfer Agent (MTA) command line uses **-DMN** to define the drive location of the postoffices and **-TSCRIPT** to specify a modem script for the asynchronous connection.

Order

- 068-099V300 Server (includes External Mail)
- 068-299V300 Remote Workstation; one copy per remote workstation

Hardware

- A dedicated IBM-compatible computer with 512K RAM to run **External**; MS-DOS or PC-DOS 3.1 or later; 2400-baud or faster modem
- PC (8088 or later) to run the Remote Workstation
- Modem or internal modem board for each external connection

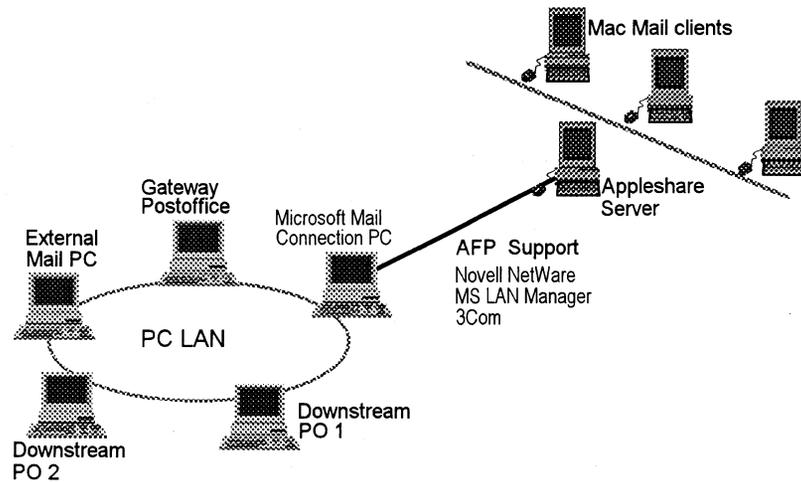
Practical Limitations

The limitations for "One Postoffice, No External, Multiple Mail Clients" apply also for this configuration.

The theoretical limit to the number of postoffices **External** can connect is 50 postoffices; the practical limit is 15 postoffices.

Modem speeds of 1200, 2400, and 9600 baud are supported. A customizable script file is included for support of unique modems and different modem speeds. An AT-compatible computer should be used for modem speeds greater than 2400 baud.

Microsoft Mail Connection



Configuration

This configuration includes multiple LANs and a concentration of Macintoshes. Use **Microsoft Mail for AppleTalk Networks** to provide electronic mail for Macintosh users. Where there are isolated Macintoshes on a PC LAN, use the Macintosh user agent to **Microsoft Mail for PC Networks**. The Macintosh user agent, together with user agents for MS-DOS, Windows, and OS/2 Presentation Manager, are all sold as part of the **Microsoft Mail Server package** and can coexist and interoperate with each other.

Use **Microsoft Mail Connection for PC & AppleTalk Networks** to link AppleTalk network and PC LAN for message exchange between the two mail systems. The physical connection between the networks may require a hardware bridge (see the following "Hardware" section). Unlike the other gateway packages, the **Microsoft Mail Connection for PC & AppleTalk Networks** doesn't require an access component for downstream postoffices.

Order

- 068-499V300 Microsoft Mail and Schedule+ Extensions for Windows for Workgroups (includes External)
- 067-096V300 Microsoft Mail for AppleTalk Networks Server

To add users to the AppleTalk LAN, order one or more of the following Microsoft Add Paks (MAPs):

- 0671796V300 Macintosh Workstation 5-pack MAP
- 0674096V300 Macintosh Workstation 20-pack MAP

- 068-099-063 Microsoft Mail Connection for PC and AppleTalk Networks

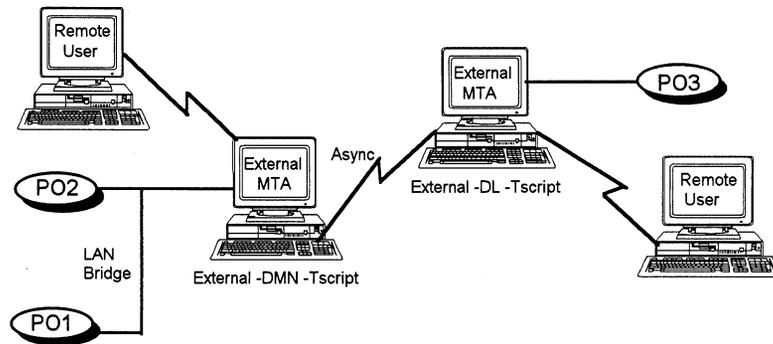
Hardware

- A dedicated IBM-compatible computer with 512K RAM is required to run **External Mail**; MS-DOS or PC-DOS 3.1 or later
- A dedicated IBM-compatible computer with 512K RAM to run the **Microsoft Mail Connection**; MS-DOS or PC-DOS 3.1 or later
- Modem or internal modem board for each external connection

Practical Limitations

The limitations for “One Postoffice, No External, Multiple Mail Clients” apply also for this configuration.

Multiple LANs Joined Asynchronously



Configuration

For a configuration with one or more offices in geographically distant locations, **Microsoft Mail for PC Networks** can provide economical electronic mail support between locations by using modems and regular phone lines to communicate. The software provides the ability to automatically connect with specified remote postoffices by storing phone numbers and, where appropriate, a calling schedule. The indirect routing capability of Microsoft Mail, together with the Script features, can be used to cut costs by using alternate telecom carriers.

In this diagram, the **External** Message Transfer Agent (MTA) command line uses the **-DMN** option to drive location of Postoffice 1 and Postoffice 2, the **-TSCRIPT** option to specify the script file used to communicate asynchronously between the remote user and Postoffice 3, and **-DL** to specify the drive location of Postoffice 3.

The limitations for “One LAN with Multiple Postoffice Configuration” apply also for this configuration.

Order

- 068-499V300 Microsoft Mail and Schedule+ Extensions for Windows for Workgroups (includes External)
- 068-299V300 Remote Workstation; one copy per remote workstation

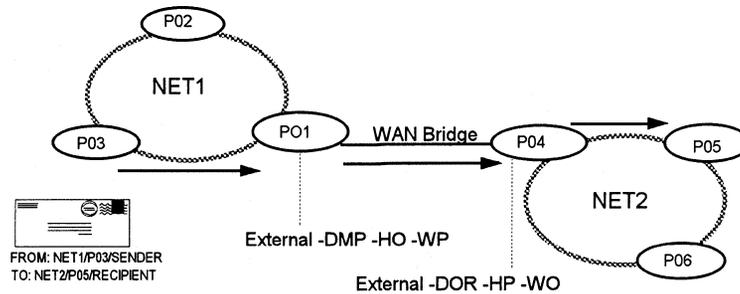
Hardware

- A dedicated IBM-compatible computer with 512K RAM to run **External Mail**; MS-DOS or PC-DOS 3.1 or later.
- A IBM-compatible computer to run Remote Workstation
- Modem or internal modem board for each external connection.

Services

- Regular business telephone line or leased line for modem connection.

Hub Postoffice for WANs



Configuration

In this configuration, the customer needs to connect multiple postoffices over a wide range of geographical locations which are however already linked together over a wide-area network (WAN). This configuration diagram shows the flow of mail from the sender on Postoffice 3 (PO3 on NET1) to the recipient on Postoffice 5 (PO5 on NET2). Instead of having each postoffice sending and receiving mail with each other postoffice on the WAN, PO1 and PO4 are defined as hub postoffices. This means that PO1 receives all the mail from PO3 and PO2. PO4 receives all the mail from PO5 and PO6. Only PO1 and PO4 need to make a connection; there is no need for connections between any of the others. This reduces both the complexity of the mail network and the amount of traffic that needs to pass along it.

In this configuration diagram the External MTA command line options denote the following:

Option	Function
-DMP	Drive location of PO2 and PO4
-HO	Drive location of the Home postoffice, PO1
-WP	WAN connection to PO4
-DOR	Drive location of PO1 and PO6
-HO	Drive location of the Home postoffice, PO4
-WO	WAN connection to PO1

The limitations for “One LAN with Multiple Postoffice Configuration” apply also for this configuration.

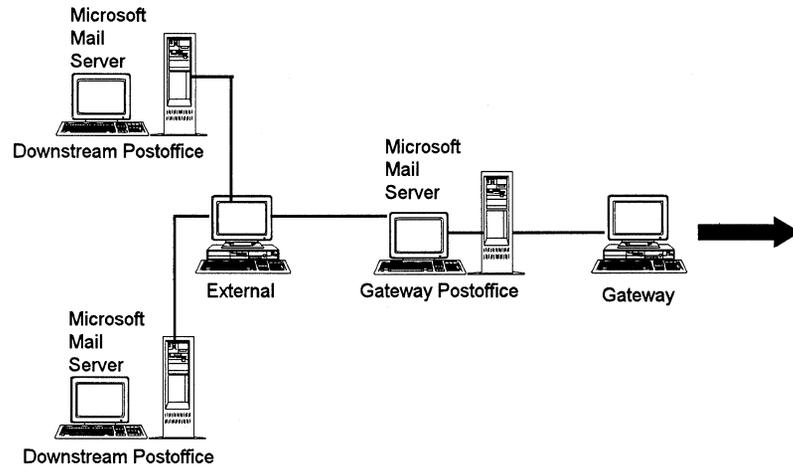
Hardware

- A dedicated IBM-compatible computer with 512K RAM to run **External Mail** on each network and support the modem connection; MS-DOS or PC-DOS 3.1 or later
- Modem or internal modem board for each external connection

Services

- Regular business telephone line or leased line for modem connection.

Hub Postoffice for Gateway Access



Configuration

This configuration uses any of the of the many gateway options available through **Microsoft Mail for PC Networks**. It also enables users to send and receive mail with a foreign mail system using a gateway such as X.400 available to all the postoffices in the Microsoft Mail environment.

One postoffice is defined as the hub postoffice on which the relevant gateway hardware and software is installed. All other postoffices become downstream postoffices from that hub. The downstream postoffices only need to install the Access component of the gateway software and don't require the full gateway hardware and software.

Order

- 068-499V300 Microsoft Mail and Schedule+ Extensions for Windows for Workgroups (includes External)

Several gateways can be used in this configuration. Using the X.400 gateway as an example:

- 068-099-128 X.400 Gateway (includes Access component)
- 068-099-129 X.400 Access; one copy per downstream postoffice

Hardware

- A dedicated IBM-compatible computer with 512K RAM to run the **External Mail**; MS-DOS or PC-DOS 3.1 or later

- Modem or internal modem board for each external connection
- A dedicated IBM-compatible computer with 640K RAM, running MS-DOS 3.1 or later, to run the gateway program
- Eicon X.25 Network Adapter and Eicon Network Interface Software.

Services

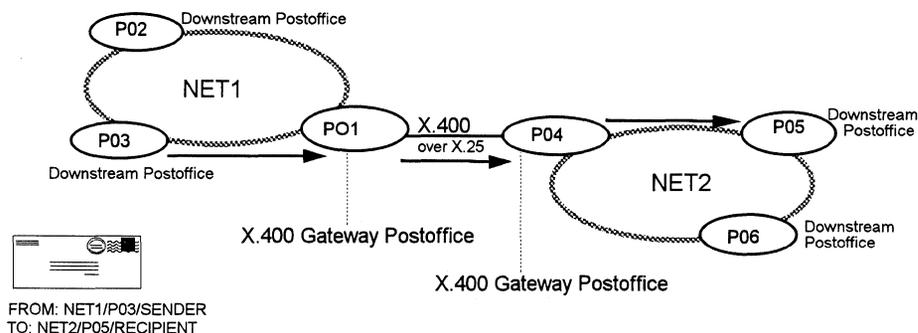
- Regular business telephone line or leased line for modem connection.
- A private X.400 link simply requires an X.25 connection between the gateway computer and an X.400 host. A connection to a public X.400 network, such as Sprint, requires an X.25 line and modem from an X.25 service company.

Practical Limitations

The limitations for “One Postoffice, No External, Multiple Mail Clients” apply also for this configuration.

The theoretical limit to the number of postoffices **External** can connect is 50 postoffices; the practical limit is 15 postoffices.

Wide-Area Networking Through a Gateway



Configuration

This configuration connects multiple postoffices over a wide range of geographical locations which are linked together over an X.25 wide-area network (WAN). In this example, the customer uses the **Microsoft Mail X.400** and a public X.400 service as the Mail hub.

This diagram illustrates how mail flows from the sender on Postoffice 3 (PO3 on NET1) through the X.400 Mail hub to the recipient on Postoffice 5 (PO5 on NET2). Instead of each postoffice sending and receiving with each other postoffice on the WAN, PO1 and PO4 are the gateway postoffices and the others are downstream postoffices. PO1 and PO4 need to make the WAN connection in order for all six postoffices to exchange mail. This reduces both the complexity of the mail network and the amount of traffic that needs to pass along it.

The connections between the postoffices on both NET 1 and/or NET2 can be either a physical link or asynchronous link between the postoffices. Once mail arrives at the Gateway postoffices, the External Mail Transfer Agent (MTA) can route the mail as in earlier examples.

Note that gateway postoffices are regular PC Mail postoffices and can have regular mail accounts associated with them like any other postoffice. (Also see gateway backbone scenarios in this appendix, such as the section called "X.400 Gateway Backbone.")

Order

- 068-499V300 Microsoft Mail and Schedule+ Extensions for Windows for Workgroups (includes External)
- 068-099-051 X.400 Gateway (includes Access component)
- 068-099-056 X.400 Access; one copy per downstream postoffice

Hardware

- A dedicated IBM-compatible computer with 512K RAM to run **External Mail** on each network; MS-DOS or PC-DOS 3.1 or later. (This may also be configured as the bridge connection between two networks, in which case a single copy of **External Mail** can support both networks.)
- A dedicated IBM-compatible computer with minimum of 640K RAM, running MS-DOS 3.1 or later, to run **Gateway**.
- Eicon X.25 Network Adapter and Eicon Network Interface Software.

Services

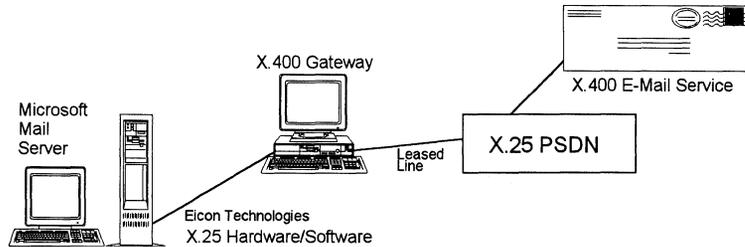
- A private X.400 link simply requires an X.25 connection between the gateway computer and an X.400 host. A connection to a public X.400 network, such as Sprint, requires an X.25 line and modem from an X.25 service company.

Practical Limitations

The limitations for “One Postoffice, No External, Multiple Mail Clients” apply also for this configuration.

The theoretical limit to the number of postoffices **External** can connect is 50 postoffices; the practical limit is 15 postoffices.

X.400 Gateway



Configuration

The configuration enables users to exchange messages with X.400-compatible foreign mail systems. The **X.400 Gateway** allows users to send mail to remote X.400 mail users as easily as to other Microsoft Mail users. Similarly X.400 mail is received from foreign mail systems through the gateway computer where the distribution list is examined and the mail is directed to the appropriate mailbox(es). The postoffice designated as the X.400 Gateway postoffice can also be a hub and can be used to route mail to a number of downstream postoffices.

Use the **X.400 Access** for each downstream postoffice requiring access to the gateway.

The **X.400 Gateway** is also used as a means of exchanging mail with proprietary electronic mail systems such as DEC's All-in-1, HP Desk, and Data General's CEO. Each of these products also supports an X.400 gateway, and the two gateways can exchange messages.

The **X.400 Gateway** is the most popular Microsoft Mail gateway. It enables customers to exchange messages with many different types of mail systems all around the world.

Order

- 068-499V300 Microsoft Mail and Schedule+ Extensions for Windows for Workgroups (includes External)
- 068-099-128 X.400 Gateway; includes Access component for gateway postoffice
- 068-099-129 X.400 Access; one copy per downstream postoffice

Hardware

- A dedicated IBM-compatible computer with 512K RAM is required to run the **External Mail** program; MS-DOS or PC-DOS 3.1 or later
- Modem or internal modem board for each external connection
- A dedicated IBM-compatible computer with 640K RAM, running MS-DOS 3.1 or later, to run Gateway
- Eicon X.25 Network Adapter and Eicon Network Interface Software.

Services

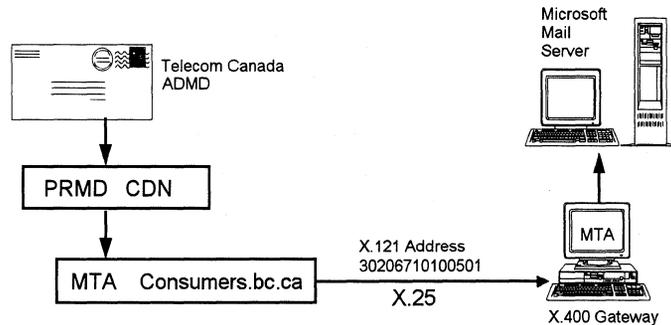
- Regular business telephone line or leased line for modem connection.
- A private X.400 link simply requires an X.25 connection between the gateway computer and an X.400 host. A connection to a public X.400 network, such as Sprint, requires an X.25 line and modem from an X.25 service company.

Practical Limitations

The limitations for “One Postoffice, No External, Multiple Mail Clients” apply also for this configuration.

The theoretical limit to the number of postoffices **External** can connect is 50 postoffices; the practical limit is 15 postoffices.

X.400 Gateway to AT&T or Other ADMD



Configuration

The configuration lets the customer register as an X.400 Private Management Domain (PRMD) on a public service provider that is providing an Administration Management Domain (ADMD) facility. Telecommunications companies (such as AT&T, BT, MCI, and Sprint) are typical ADMD service providers.

When a company registers as a PRMD on an ADMD, that company can send and receive X.400 messages without having to maintain a continuous online access. The ADMD acts as a central repository for messages to avoid tie-ups resulting from continuous PRMD-to-PRMD message transfers. The sending PRMD can leave its messages at the ADMD and the receiving PRMD can then collect the message by using a X.25 line for access.

X.400 addressing is very comprehensive to cater to multiple addresses. However this is hidden from the Microsoft Mail user by the friendly user names that are available from the Global Address List.

Order

- 068-499V300 Microsoft Mail and Schedule+ Extensions for Windows for Workgroups (includes External)
- 068-099-128 X.400 Gateway; includes Access component for gateway postoffice
- 068-099-129 X.400 Access; one copy per downstream postoffice

Hardware

- A dedicated IBM-compatible computer with 512K RAM to run **External Mail** on each network; MS-DOS or PC-DOS 3.1 or later. (This may also be configured as the bridge connection between two networks, in which case a single copy of **External Mail** can support both networks.)
- A dedicated IBM-compatible computer with minimum of 640K RAM, running MS-DOS 3.1 or later, to run the gateway program.
- Eicon X.25 Network Adapter and Eicon Network Interface Software.

Services

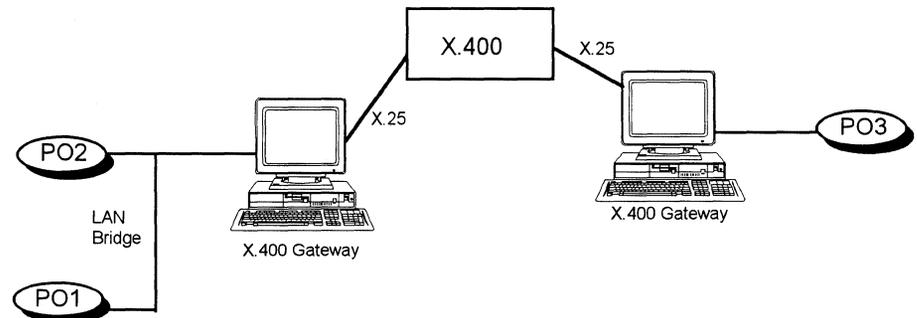
- A private X.400 link simply requires an X.25 connection between the gateway computer and an X.400 host. A connection to a public X.400 network, such as Sprint, requires an X.25 line and modem from an X.25 service company.

Practical Limitations

The limitations for “One Postoffice, No External, Multiple Mail Clients” apply also for this configuration.

The theoretical limit to the number of postoffices **External** can connect is 50 postoffices; the practical limit is 15 postoffices.

X.400 Gateway Backbone



Configuration

This configuration uses the **X.400 Gateway** interconnects electronic mail through an existing X.400 system.

For example, this diagram illustrates a configuration using DEC-oriented with computers running All-in-One and VMS. The existing message-router network can be used to connect two remote PC LANs and provide mail exchange with the DEC electronic mail services.

The **X.400 Gateway** enables users to send mail to remote X.400 mail users as easily as to other Microsoft Mail users. Similarly, X.400 mail is received through the gateway computer where it is reformatted and placed in the appropriate mailbox. The postoffice designated as the X.400 Gateway postoffice can be used to route mail to a number of downstream postoffices. The **X.400 Access** product must be purchased for each downstream postoffice requiring access to the gateway.

Connectivity provided by the X.400 gateway makes **Microsoft Mail for PC Networks** a practical and economical solution for companies that choose X.400 messaging as an open systems strategic direction. See the “Master Connectivity Map” section earlier in this appendix for more information.

The Windows client for Mail 3.x supports a number of advanced features such as OLE and intelligent attachments. In this “backboning” scenario, the gateway has been enhanced to support these new features. The gateway supports message “encapsulation,” which allows a message with complex message parts to be sent from one PC Mail user to another PC Mail user through the gateway. Message encapsulation ensures that the integrity of the message, with its complex message parts, remains fully intact as it passes through the host system.

Order

- 068-499V300 Microsoft Mail and Schedule+ Extensions for Windows for Workgroups (includes External)
- 068-099-051 X.400 Gateway; (includes Access component for gateway postoffice)
- 068-099-056 X.400 Access; one copy per downstream postoffice

Hardware

- A dedicated IBM-compatible computer with 512K RAM to run **External Mail** on each network; MS-DOS or PC-DOS 3.1 or later. (This may also be configured as the bridge connection between two networks, in which case a single copy of **External Mail** can support both networks.)
- A dedicated IBM-compatible computer with 640K RAM, running MS-DOS 3.1 or later, to run the gateway program.
- Eicon X.25 Network Adapter and Eicon Network Interface Software.

Services

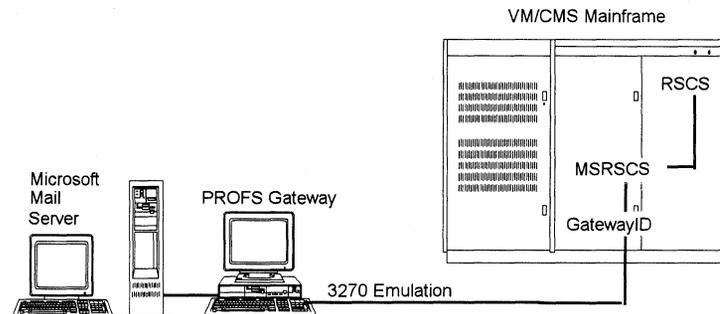
- A private X.400 link simply requires an X.25 connection between the gateway computer and an X.400 host. A connection to a public X.400 network, such as Sprint, requires an X.25 line and modem from an X.25 service company.

Practical Limitations

The limitations for “One Postoffice, No External, Multiple Mail Clients” apply also for this configuration.

The theoretical limit to the number of postoffices **External** can connect is 50 postoffices; the practical limit is 15 postoffices.

Gateway to PROFS



Configuration

This configuration includes two networks—a network of PROFS users and a PC-oriented network running **Microsoft Mail for PC Networks**.

The **PROFS Gateway** enables users on any of the networks to exchange messages using the interface with which they are familiar. The gateway comprises two parts:

- **PROFS Gateway** is gateway and access software for the LAN. This software runs on a dedicated personal computer that also includes the mechanism for 3270 file transfer.
- **PROFS Host Access** runs on the IBM host under CMS. This software runs with two service computers—one is the actual gateway with a class “G” CMS identification, and the other is a Remote Spooling Communications Subsystem (RSCS) virtual machine.

This software provides seamless communication with remote messaging systems. The **PROFS Gateway** is also applicable where PROFS isn't present but connectivity to CMS users is required.

A single gateway on the LAN provides mail access to users with mailboxes on the downstream postoffices as well as those on the gateway postoffice. Each downstream postoffice that requires indirect access through the gateway must install the access component of the **PROFS Gateway**.

Order

- 068-499V300 Microsoft Mail and Schedule+ Extensions for Windows for Workgroups (includes External)
- 068-099-132 PROFS Host Access 3.2; one license per mainframe CPU
- 068-099-130 PROFS Gateway 3.0; includes the access component and software to support all connection types
- 068-099-131 PROFS Access 3.0; one copy per downstream postoffice

Hardware

- A dedicated IBM-compatible computer with 512K RAM running MS-DOS 3.1 or later to run the **External Mail**. If two LANs are bridged using a PC as a bridging computer, one copy of **External Mail** running on this will serve both networks.
- A dedicated IBM-compatible computer with 512K RAM running MS-DOS 3.1 or later to support LAN gateway. The gateway computer can be connected to the host in one of several ways, including the following:
 - o 3270 emulation card with direct 4-wire connection to cluster controller
 - o 3270 emulation card plus a Token-Ring attachment
 - o SDLC card and synchronous modem

Host Software

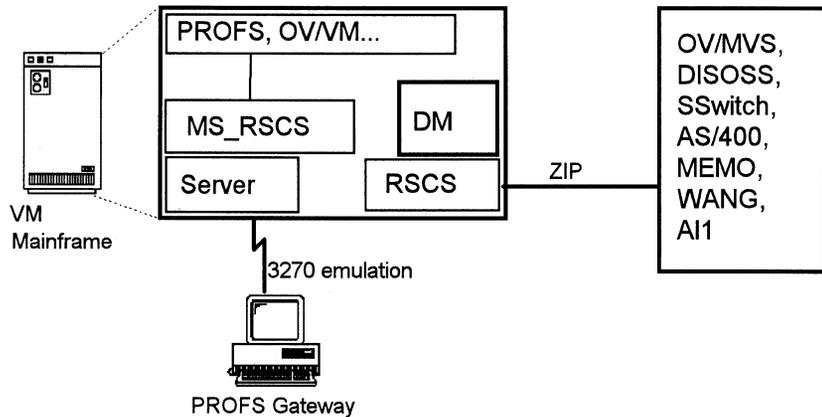
- IBM VM/SP Release 4.0 or later, or VM/XA, VM/HPO, or VM/ESA.
- IBM RSCS Release 2.2 or later.
- IBM File Transfer Program 5664-281 (IND\$FILE) or compatible program.
- IBM PROFS (optional)
- IBM OfficeVision/VM (optional)

Practical Limitations

The limitations for “One Postoffice, No External, Multiple Mail Clients” apply also for this configuration.

The theoretical limit to the number of postoffices **External** can connect is 50 postoffices; the practical limit is 15 postoffices.

PROFS Distribution Manager



Configuration

This configuration includes a number of host-based electronic mail systems in addition to the PC networks that support **Microsoft Mail for PC Networks**. **PROFS Distribution Manager (DM)** extends the **PROFS Gateway** so that mail can be exchanged between PC Mail users and any electronic mail system that supports the IBM Zip message format. These systems include IBM PS-CICS, IBM OfficeVision/MVS, IBM DISOSS, IBM System 36 and 38, HP Desk, Soft-Switch Central, IBM OfficeVision/400, Verimation MEMO, WANG Office, DEC All-in-One. **DM** is a VM host application and is installed with the **PROFS Host Access** component. **DM** supports DCA, RFT and all PROFS document types.

Order

- 068-499V300 Microsoft Mail and Schedule+ Extensions for Windows for Workgroups (includes External)
- 068-099-161 Distribution Manager 3.2
- 068-099-132 PROFS Host Access 3.2; one license per mainframe CPU
- 068-099-130 PROFS Gateway 3.0; includes the access component and software to support all connection types
- 068-099-131 PROFS Access 3.0; one copy per downstream postoffice

Hardware

- A dedicated IBM-compatible computer with 512K RAM running MS-DOS 3.1 or later is required to run **External Mail**. If two LANs are bridged using a personal computer as a bridge, one copy of **External Mail** running on this will serve both networks.
- A dedicated IBM-compatible computer with 512K RAM running MS-DOS 3.1 or later to support LAN gateway. The gateway computer may be connected to the host in many ways, including the following:
 - o 3270 emulation card with direct 4-wire connection to cluster controller.
 - o 3270 emulation card plus a Token-Ring attachment.
 - o SDLC card and synchronous modem.

Host Software

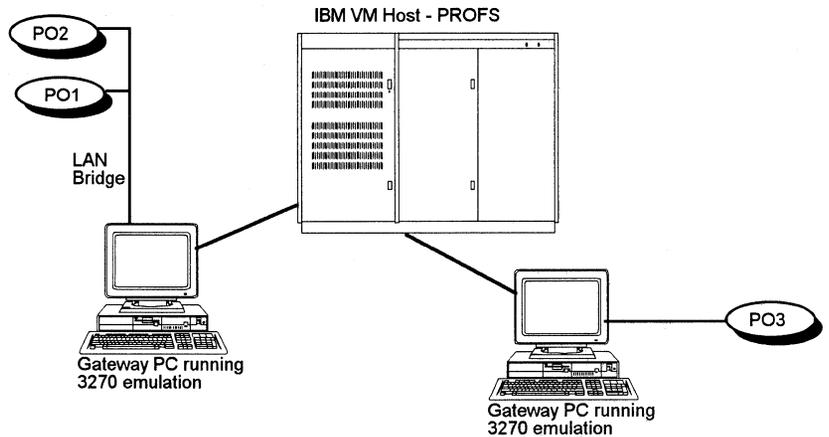
- IBM VM/SP Release 4.0 or later, or VM/XA, VM/HPO, or VM/ESA.
- IBM RSCS Release 2.2 or later.
- IBM File Transfer Program 5664-281 (IND\$FILE) or compatible program.
- IBM PROFS (optional)
- IBM OfficeVision/VM (optional)

Practical Limitations

The limitations for “One Postoffice, No External, Multiple Mail Clients” apply also for this configuration.

The theoretical limit to the number of postoffices **External** can connect is 50 postoffices; the practical limit is 15 postoffices.

PROFS Gateway Backboning



Configuration

The **PROFS Gateway** provides interconnect PC-oriented electronic mail systems through an existing PROFS network. PROFS has the largest installed base of any mail system, so making use of an existing PROFS network to connect PC-oriented LANs provides an efficient and economical solution in many accounts. The **PROFS Gateway** software allows Microsoft Mail users to exchange messages with PROFS users as easily as with one another. From the PROFS perspective, Microsoft Mail users simply appear to be on another PROFS node.

A single gateway on each PC-oriented LAN provides mail access to users with mailboxes on the downstream postoffices in addition to those on the gateway postoffice. Each downstream postoffice that requires indirect access through the gateway must install the Access component of the **PROFS Gateway**.

The Windows client for Mail 3.x supports a number of advanced features such as OLE and intelligent attachments. In this "backboning" scenario, the gateway has been enhanced to support these new features. The gateway supports message "encapsulation," which allows a message with complex message parts to be sent from one PC Mail user to another PC Mail user through the gateway. Message encapsulation ensures that the integrity of the message, with its complex message parts, remains fully intact as it passes through the host system.

Order

- 068-499V300 Microsoft Mail and Schedule+ Extensions for Windows for Workgroups (includes External)
- 068-099-132 PROFS Host Access 3.2; one license per mainframe CPU
- 068-099-130 PROFS Gateway 3.0; includes the access component and software to support all connection types.
- 068-099-131 PROFS Access 3.0; one copy per downstream postoffice

Hardware

- A dedicated IBM-compatible computer with 512K RAM running MS-DOS 3.1 or later is required to run **External Mail**. If two LANs are bridged using a PC as a bridging machine, one copy of **External Mail** running on this will serve both networks.
- A dedicated IBM-compatible computer with 512K RAM running MS-DOS 3.1 or later to support LAN gateway. The gateway computer may be connected to the Host in many ways, including the following:
 - 3270 emulation card with direct 4-wire connection to cluster controller.
 - 3270 emulation card plus a Token-Ring attachment.
- SDLC card and synchronous modem.

Host Software

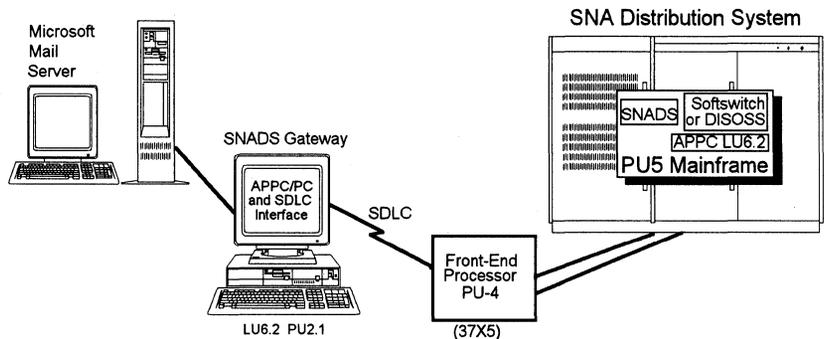
- IBM VM/SP Release 4.0 or later, or VM/XA, VM/HPO, or VM/ESA.
- IBM RSCS Release 2.2 or later.
- IBM File Transfer Program 5664-281 (IND\$FILE) or compatible program.
- IBM PROFS (optional)
- IBM OfficeVision/VM (optional)

Practical Limitations

The limitations for “One Postoffice, No External, Multiple Mail Clients” apply also for this configuration.

The theoretical limit to the number of postoffices **External** can connect is 50 postoffices; the practical limit is 15 postoffices.

SNADS Gateway



Configuration

The configuration in this example enables message exchanges with DISOSS users on an SNA network. An IBM AT (286 processor), or better, is dedicated to running the **SNADS Gateway** between the MS-Net compatible local area network and the SNA Distribution Service. The gateway enables users with mailboxes on the LAN to exchange mail and files with users of any SNADS-based mail system.

A single gateway on the LAN provides mail access to users with mailboxes on the downstream postoffices in addition to those on the gateway postoffice. The **SNADS Message Service** product must be purchased for each downstream postoffice requiring access to the gateway.

Foreign electronic mail systems that PC Mail users can exchange mail with include IBM's AS400 Office, Verimation's Memo, Soft-Switch Central, and any other SNADS based mail systems.

Order

- 068-499V300 Microsoft Mail and Schedule+ Extensions for Windows for Workgroups (includes External)
- 068-099-060 SNADS Message Service; one copy per downstream postoffice
- SNADS Gateway for MS-DOS *or* SNADS Gateway for OS/2, from LinkAge Office Information Solutions
- SNADS Gateway for MS-DOS or Soft-Switch Central from Soft-Switch

Hardware

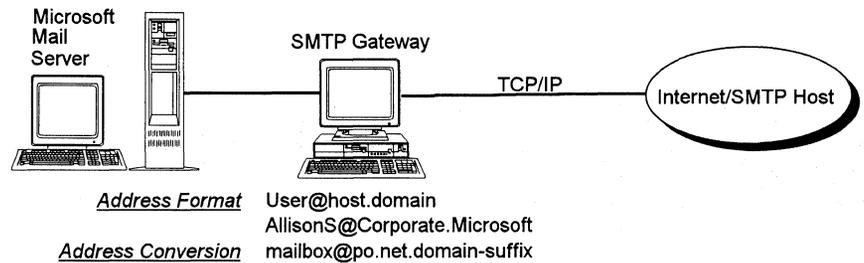
- A dedicated IBM-compatible computer with 512K RAM running MS-DOS 3.1 or later is required to run **External Mail**. If two LANs are bridged using a PC as a bridging machine, one copy of **External Mail** running on this will serve both networks.
- A dedicated 286 or better to support the LAN gateway.
- SDLC card and synchronous modem to connect to SNADS.
- APPC/PC to create an LU6.2 connection.

Practical Limitations

The limitations for “One Postoffice, No External, Multiple Mail Clients” apply also for this configuration.

The theoretical limit to the number of postoffices **External** can connect is 50 postoffices; the practical limit is 15 postoffices.

SMTP Gateway



Configuration

This configuration enables data and message exchange with remote UNIX or non-UNIX hosts using the Simple Mail Transfer Protocol. In this example, the **SMTP Gateway** converts outgoing messages from the PC Mail format to the Internet format before transmitting them to a *smart SMTP host* for routing. Messages from remote sites may be directly routed to the gateway for conversion to PC Mail format and delivery to the recipient. Where the SMTP gateway is used to connect PC Mail directly to a DEC VAX, a smart SMTP host may not be required.

The dedicated gateway computer and the SMTP host are linked by a TCP/IP connection, which may co-exist with the network protocol on the LAN. In routing to the Internet, the gateway computer can't do its own routing and relies on a smart host for this service.

Users with mailboxes on the gateway postoffice or downstream postoffices can send and receive SMTP mail. The **SMTP Access** component is installed on each postoffice requiring the service, enabling these users to exchange mail with remote SMTP users as easily as they exchange mail internally.

Order

- 068-499V300 Microsoft Mail and Schedule+ Extensions for Windows for Workgroups (includes External)
- 068-099-126 SMTP Gateway 3.0
- 068-099-127 SMTP Access 3.0

Hardware

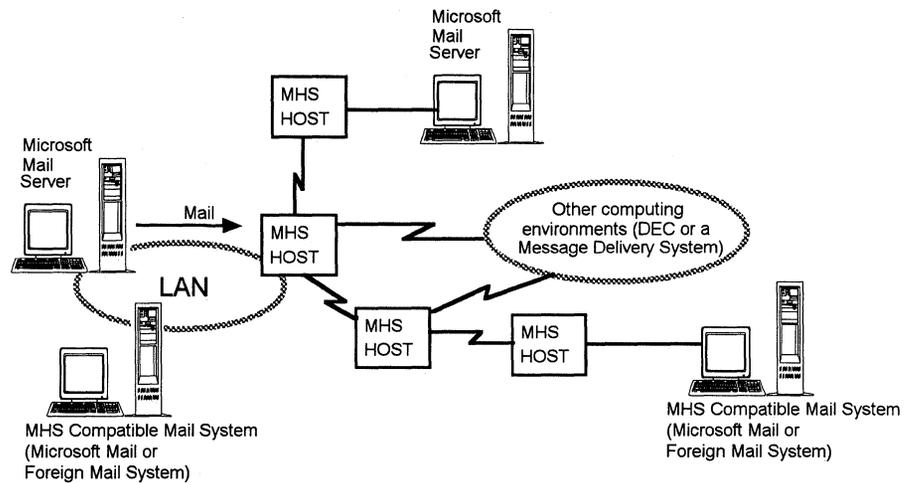
- A dedicated IBM-compatible computer with 512K RAM running MS-DOS 3.1 or later is required to run **External Mail**. If two LANs are bridged using a PC as a bridge, one copy of **External Mail** running on this will serve both networks.
- Smart SMTP host (UNIX, XENIX, and so on) for routing mail to final SMTP destinations.
- A dedicated IBM-compatible computer with 512K RAM running MS-DOS 3.1 or later and one of the following hardware/software combinations to support TCP/IP:
 - o FTP Software Inc.'s PC/TCP Kernel and any network adapter card which supports it.
 - o The Wollongong Group's WIN/TCP and any network adapter card which supports it.
 - o Excelan® EXOS 205T Ethernet card and the EXOS TCP/IP transport system.

Practical Limitations

The limitations for “One Postoffice, No External, Multiple Mail Clients” apply also for this configuration.

The theoretical limit to the number of postoffices **External** can connect is 50 postoffices; the practical limit is 15 postoffices.

MHS Gateway



Configuration

This configuration enables message exchange with a Novell NetWare LAN running an MHS-based electronic mail system, such as DaVinci electronic mail. The **MHS Gateway** provides a seamless interface between the systems, enabling Microsoft Mail users to send mail to MHS users as easily as to one another. Incoming MHS mail is sent to the gateway computer where it is re-formatted and placed in the appropriate mailbox. One gateway postoffice can be used to forward mail to a number of downstream postoffices. The **MHS Access** product must be purchased for each downstream postoffice requiring access to the gateway. The MHS gateway can be installed on the MHS Connectivity Manager, so a dedicated PC may not be required.

Order

- 068-499V300 Microsoft Mail and Schedule+ Extensions for Windows for Workgroups (includes External)
- 068-099-124 MHS Gateway 3.0
- 068-099-125 MHS Access 3.0

Hardware

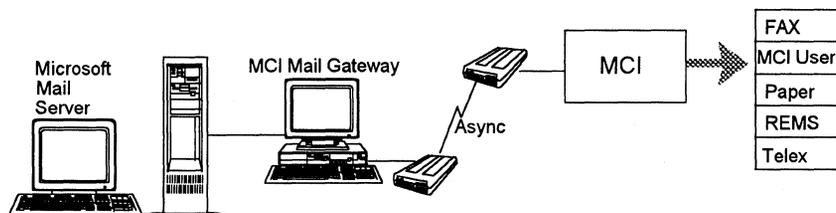
- A dedicated IBM-compatible computer with 512K RAM to run **External Mail** on the network; MS-DOS or PC-DOS 3.1 or later. This may also be configured as the bridge connection between two networks, in which case a single copy of **External Mail** can support both networks.
- Optionally, a dedicated IBM-compatible computer with 512K RAM and running MS-DOS 3.1 or later to run the gateway program. The gateway software can be installed on the MHS Connectivity Manager.
- LAN Network Adapter and redirection software.
- Novell network should run MHS 1.1 or later.

Practical Limitations

The limitations for “One Postoffice, No External, Multiple Mail Clients” apply also for this configuration.

The theoretical limit to the number of postoffices **External** can connect is 50 postoffices; the practical limit is 15 postoffices.

MCI Gateway



Configuration

Adding the **MCI Mail Gateway** to a network running **Microsoft Mail for PC Networks** allows Microsoft Mail users easy access to the MCI Mail features. (The AT&T Gateway, available in first quarter of 1993, will operate the same way.)

In addition to exchanging messages with MCI Mail users worldwide, the gateway provides the ability to send a fax or telex without requiring a fax or telex machine. The gateway provides access to the MCI Mail *Paper Mail* interface, which generates a first-class letter as geographically close as possible to the recipient. The MCI Mail REMS (Remote Electronic Mail Systems) account also provides an interface to businesses with private electronic mail systems.

Users of Microsoft Mail with mailboxes either on the gateway postoffice or on a downstream postoffice, can route a message to the appropriate service. The **MCI Mail Gateway** has a unique MCI Mail identification, and all incoming MCI Mail is addressed to this identification. The gateway routes the mail directly to a mailbox on the gateway postoffice or indirectly to a downstream postoffice.

Order

- 068-499V300 Microsoft Mail and Schedule+ Extensions for Windows for Workgroups (includes External)
- 068-099-053 MCI Mail Gateway (includes Access component for gateway postoffice)
- 068-099-052 MCI Mail Access; one copy per downstream postoffice

Hardware

- A dedicated IBM-compatible computer with 512K RAM to run **External Mail** on the network; MS-DOS or PC-DOS 3.1 or later.
(The **MCI Mail Gateway** program doesn't require a dedicated IBM-compatible computer. The **MCI Mail Gateway** can be spawned by **External Mail**.)
- Modem or internal modem board in gateway computer.

Services

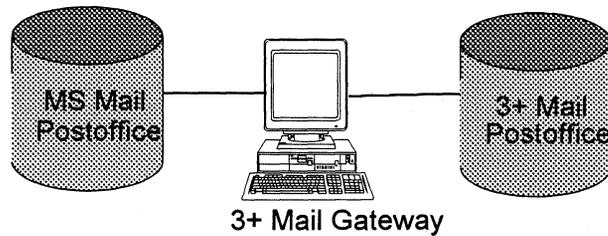
- An MCI Mail REMS account and business telephone line.

Practical Limitations

The limitations for “One Postoffice, No External, Multiple Mail Clients” apply also for this configuration.

The theoretical limit to the number of postoffices **External** can connect is 50 postoffices; the practical limit is 15 postoffices.

3+Mail Gateway



Configuration

This configuration includes both PC Mail and 3Com's 3+Mail products. The gateway enables both mail systems to coexist and communicate seamlessly on your current network, permitting a staged, server-to-server conversation with minimal disruption for current mail users.

See the earlier section called "Order Information" for ordering options.

Hardware

- A dedicated IBM-compatible computer with 512K RAM to run **3+Mail Gateway**; MS-DOS or PC-DOS 3.1 or later
- A network adapter card and compatible software to access network that contains the Microsoft Mail files
- The same network adapter card or another card to access the 3+Share/3+Open XNS system and deliver mail to the 3+ Server

Practical Limitations

The limitations for "One Postoffice, No External, Multiple Mail Clients" apply also for this configuration.

The theoretical limit to the number of postoffices **External** can connect is 50 postoffices; the practical limit is 15 postoffices.

Configuration Guide Glossary

ADMD Administrative Domain attribute of X.400 O/R address. A management domain managed by an administration. In practice, the name of the nation-wide carrier through which communications are routed, for example *Telecom Canada*.

APPC/PC An Application Program Interface (API) used by the OS/2 Communications Manager.

AS/400 IBM minicomputer, which can exchange mail with Microsoft Mail for PC Networks, by using the SNADS Gateway.

Backboning In the context of Microsoft Mail, backboning means supporting the interconnection of multiple PC LANs through an external messaging system. Examples are the use of a PROFS or an X.400 network to link Microsoft Mail users on two or more LANs.

Bit-Mapped Display Each pixel (dot) on the screen is individually addressable.

CAS Fax term for Communication Application Specification.

CCITT International Telegraphy and Telephony Consultative Committee. The CCITT recommends standards for data communications interfaces—for example X.25, X.400, X.500.

Client The part of an electronic mail system that resides on the user's workstation.

CUA Common User Access. One of four components of IBM's System Application Architecture (SAA), CUA specifies conventions for the dialog between a user and the computer.

CUT Control Unit Terminal (3270 Emulation support).

DFT Distributed Function Terminal (3270 emulation support).

DISOSS **DIS**tributed **O**ffice **S**upport **S**ystem. An electronic office system implemented on IBM mainframes under the MVS operating system.

External Also known as the External Mail program, this program is responsible for transferring mail and directory information between mail servers. **External** requires a dedicated computer.

Gateway In the context of electronic mail, a gateway is a connection to a foreign mail system.

Gateway Postoffice In an electronic mail system, a gateway postoffice serves as the routing point or hub for all gateway-bound mail. The gateway SKU is installed on it.

HLLAPI High Level Language API; term used with 3270 emulation support.

ISDN Integrated Services Digital Network. An all-digital telephone connection which offers the ability to transmit digital data without a modem.

LLAPI Low Level API. A term used with 3270 emulation support.

Mail Backbone The messaging infrastructure of a company. Refers to the messaging systems and the network environment.

MHS Message Handling System. Developed by Action Technologies Inc., MHS comprises a set of electronic mail protocols that are implemented by many electronic mail vendors.

MTA Message Transfer Agent. The software in an electronic mail system which handles routing, forwarding, and message bundling. Implemented in Microsoft Mail in both the user agent *Mail* program and in **External** for inter-postoffice transfer.

NBP NetBIOS Protocol.

Network Courier The name of Microsoft Mail for PC Networks when it was marketed by Consumer's Software Inc.

ODI Open Datalink Interface.

OfficeVision IBM's standard for delivering the full range of office services through icon-controlled screens. The three distinct products are: OfficeVision/VM, a PROFS upgrade; OfficeVision/MVS, a DISOSS upgrade; and OfficeVision LAN, which runs on top of OS/2.

OS/2 A multitasking operating system developed by Microsoft for AT and PS/2 personal computers.

OS/2 EE IBM's OS/2 Extended Edition is a proprietary operating system for the PS/2. It features integrated communications and database facilities and is required for OfficeVision.

OSI Open Systems Interconnection. A seven-layer network architecture standard developed by ISO to enable OSI-compliant devices to exchange data.

PDS Packet Driver Specification.

Postoffice A Microsoft Mail "server." A postoffice consists of a collection of files which contain user mailboxes, folders, and the directory.

Presentation Manager The graphical user interface for OS/2. Presentation Manager complies with IBM's SAA standard for CUA.

PRMD Private Domain Name attribute of X.400 O/R address. Management Domain managed by an organization.

PROFS Professional Office System. An IBM electronic office product implemented on mainframes.

RSCS Remote Spooler Communications Subsystem.

SAA Systems Application Architecture. IBM's framework for developing applications in the MVS, VM, AS/400, and OS/2 environments. SAA consists of conventions and protocols for software development.

Server **Microsoft Mail Server** is the software product that includes all programs required to configure an electronic mail postoffice. In the hardware sense, a server is a processor that handles various duties in a network including database and file management.

SMTP Simple Mail Transfer Protocol. A basic electronic mail service that is included with TCP/IP and is primarily used between UNIX hosts.

SNADS SNA Distribution Service. SNADS provides store-and-forward among multiple IBM hosts.

TCP/IP Transmission Control Protocol/Internet Protocol. A set of protocols developed for UNIX systems providing transport and network services.

Transport Independent Application that doesn't rely on a particular electronic mail transport—for example, PC Mail 3.x Windows Client.

UUCP The standard UNIX asynchronous (modem) communications protocol.

X.121 A standard for constructing numeric identifiers, for instance the unique 15 digit X.400 address attribute. This is an alternative to the composite text form.

X.400 An evolving CCITT Open Systems Interconnection standard protocol for exchanging and addressing electronic messages. There are two standard dates, 1984 and 1988. Most existing X.400 gateways comply with the 1984 standard. The standard is backward compatible.

X.500 An evolving CCITT standard for directory services.

XNS Xerox Network System.

Flowcharts for Troubleshooting Windows for Workgroups

This chapter contains flowcharts to help aid troubleshooting common problems that you may encounter with Windows for Workgroups. The flowcharts in this appendix were originally provided with the *Windows Resource Kit*, and have been updated to reflect Windows for Workgroups.

The troubleshooting flowcharts provide an orientation and quick introduction to problem-solving for Windows for Workgroups 3.1. Each chart presents a brief statement of the problem with a decision tree for determining the source and solution.

Related information

- *Windows for Workgroups Getting Started*. Chapter 6, "Troubleshooting."

The flowcharts are organized in the following sequence.

Flowchart Series 1: Troubleshooting Setup

Chart 1.1: System Requirements for Windows for Workgroups	C-3
Chart 1.2: Windows for Workgroups Hangs During Setup	C-4
Chart 1.3: Setup /n Fails	C-5
Chart 1.4: Known Problems with Computer BIOS	C-6
Chart 1.6: Reinstalling Windows for Workgroups Without Losing Settings	C-8
Chart 1.7: Expanding Files from the Windows for Workgroups Disks	C-9
Chart 1.8: Stacker	C-10

Flowchart Series 2: Troubleshooting Hardware Problems

Chart 2.1: Mouse Doesn't Work in Windows for Workgroups	C-11
Chart 2.2: Serial Port Problems (Communications and Terminal)	C-12
Chart 2.3: Display Adapter Problems (Distorted Video)	C-13
Chart 2.4: Cannot Access CD-ROM in 386 Enhanced Mode	C-14
Chart 2.5: Cannot Access Floppy Drives in 386 Enhanced Mode	C-15
Chart 2.6: Cannot Access Hard Disk in 386 Enhanced Mode	C-17
Chart 2.7: Common IRQ Usage for PC-AT and 80386 Computers	C-18
Chart 2.8: Mouse Doesn't Work in Windowed MS-DOS-based Applications	C-19

Flowchart Series 3: Troubleshooting Fonts

Chart 3.1: TrueType Fonts.....	C-20
Chart 3.2: Disabling Adobe Type Manager.....	C-21
Chart 3.3: Disabling Bitstream FaceLift.....	C-22
Chart 3.4: Disabling HP Intellifont.....	C-23

Flowchart Series 4: Troubleshooting Printing

Chart 4.1: Cannot Print on Standalone PC.....	C-24
Chart 4.2: Cannot Print on Network.....	C-25
Chart 4.3: Garbled Output.....	C-26
Chart 4.4: Garbled Output on Novell Network.....	C-27

Flowchart Series 5: Troubleshooting Operating Problems

Chart 5.1: Cannot Run 386 Enhanced Mode.....	C-28
Chart 5.3: General Protection Faults.....	C-31
Chart 5.4: Application Execution Error.....	C-33
Chart 5.5: System Integrity Violation in 386 Enhanced Mode.....	C-34
Chart 5.6: Disabling Norton Desktop for Windows for Workgroups.....	C-35
Chart 5.7: Disabling PC-Tools Version 7.....	C-36
Chart 5.8: Out-of-Memory Errors in Windows for Workgroups.....	C-37
Chart 5.9: Cannot Run Windows for Workgroups in Standard Mode.....	C-38
Chart 5.10: Problems with Object Linking and Embedding.....	C-39

Flowchart Series 6: Troubleshooting Multimedia

Chart 6.1: Multimedia Drivers.....	C-40
Chart 6.2: Media Player.....	C-41
Chart 6.3: MIDI Mapper.....	C-42
Chart 6.4: Music Box.....	C-43
Chart 6.5: Sound.....	C-44
Chart 6.6: Sound Recorder.....	C-45

Chart 1.1 System Requirements Windows for Workgroups

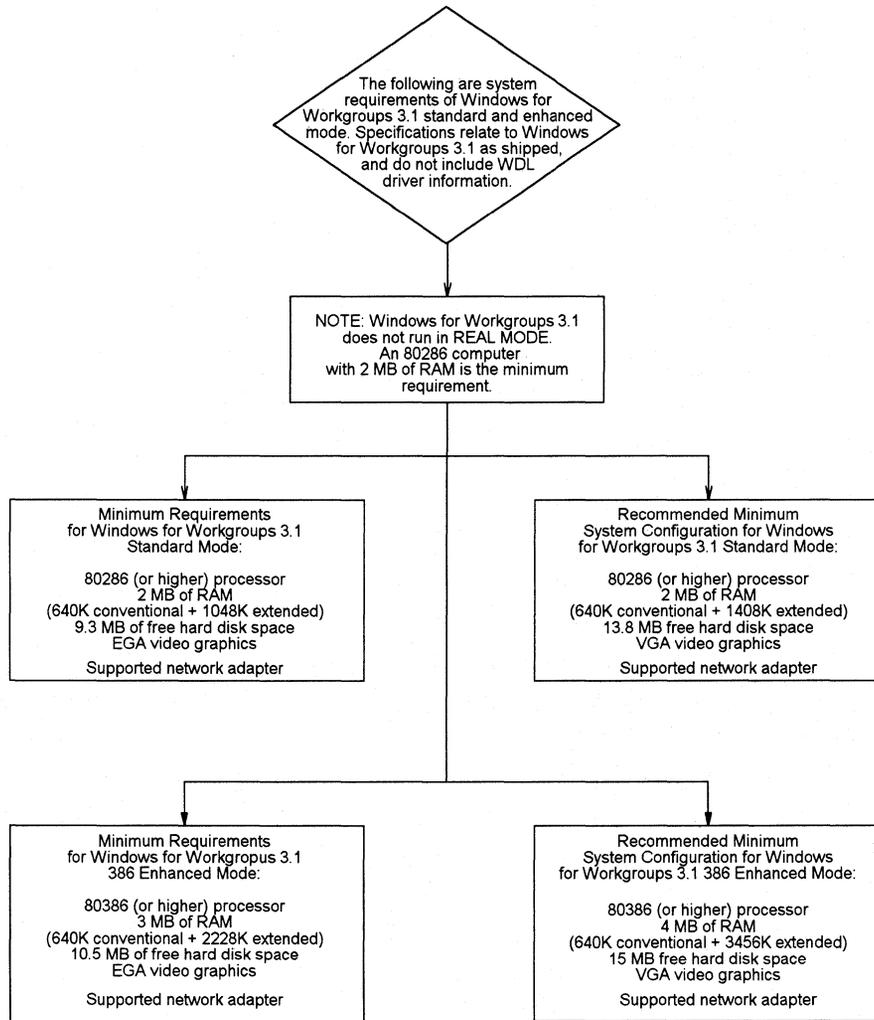


Chart 1.2 Windows for Workgroups Hangs During Setup

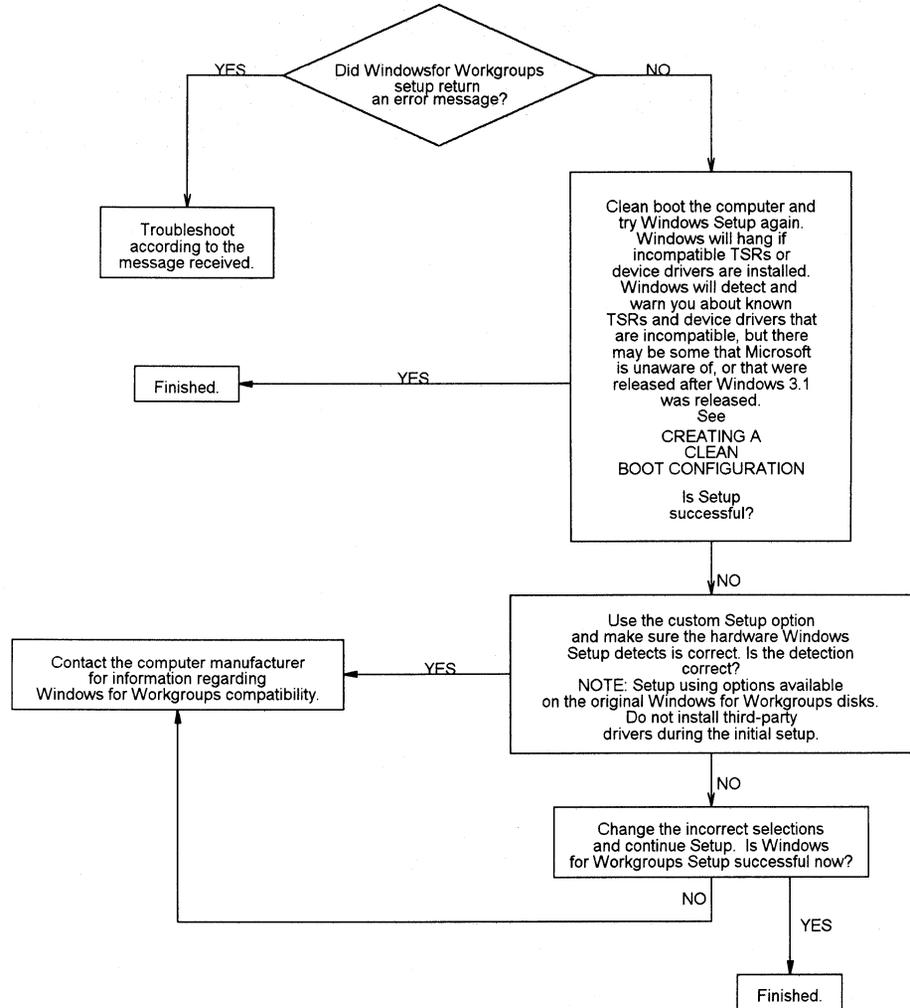


Chart 1.3 Setup /n Fails

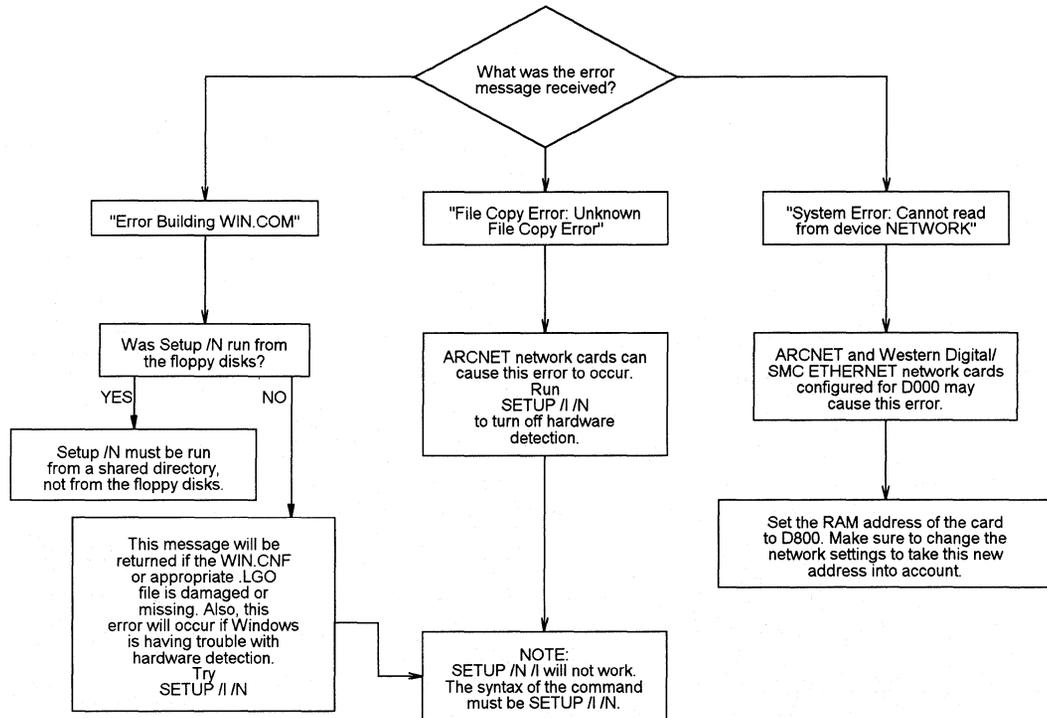


Chart 1.4 Known Problems with Computer BIOS

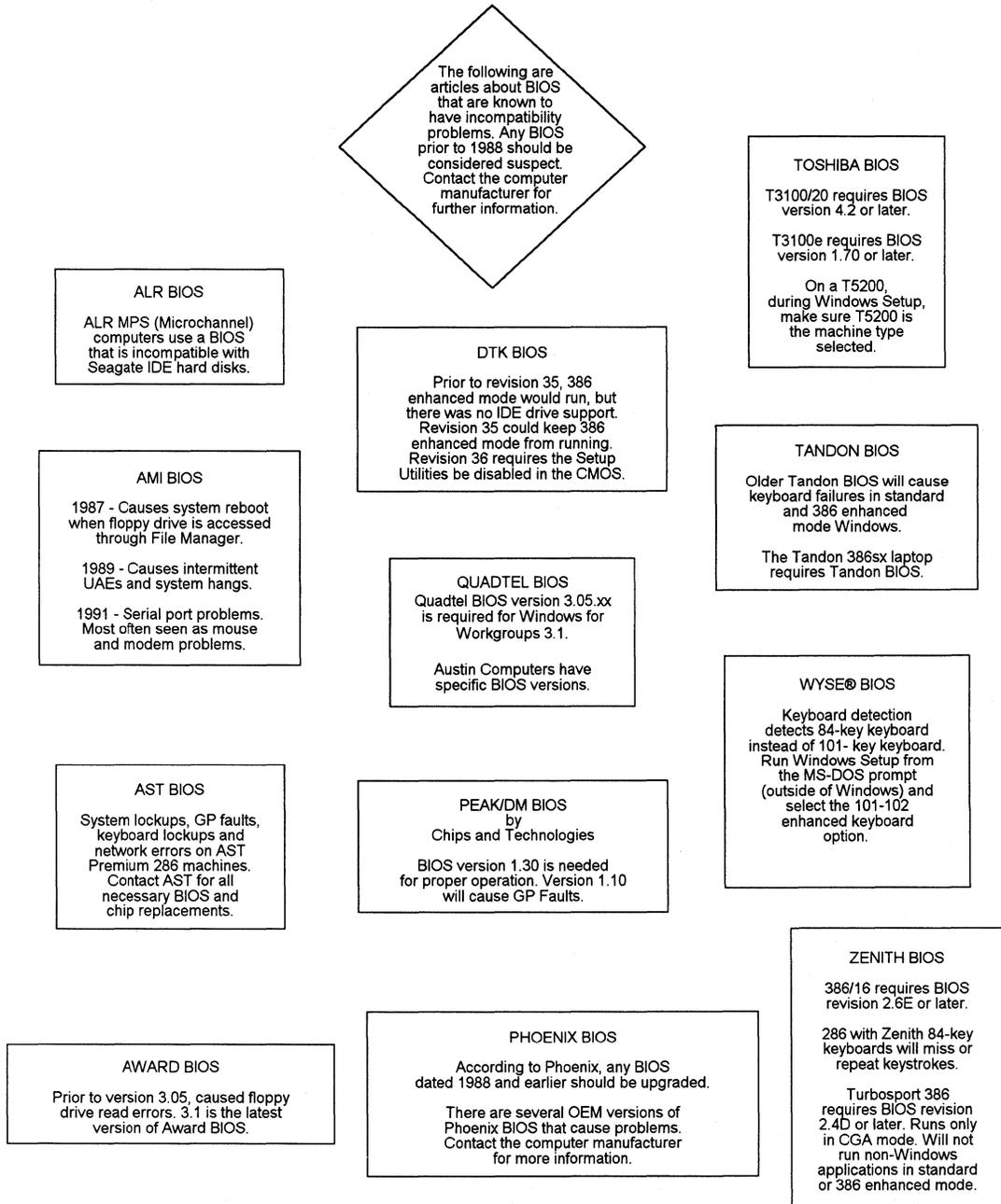


Chart 1.5

This page left intentionally blank

Chart 1.6 **Reinstalling Windows for Workgroups Without Losing Settings**

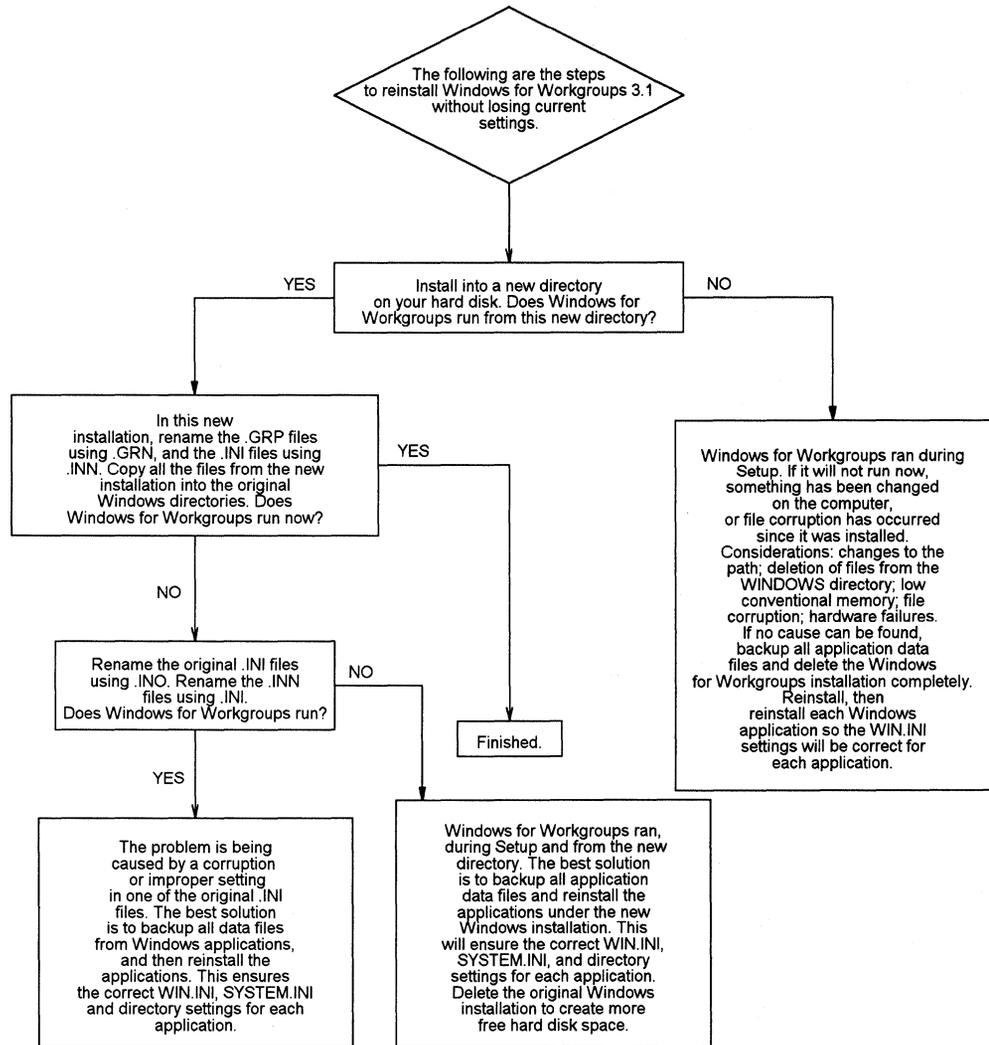


Chart 1.7 Expanding Files from the Windows for Workgroups Disks

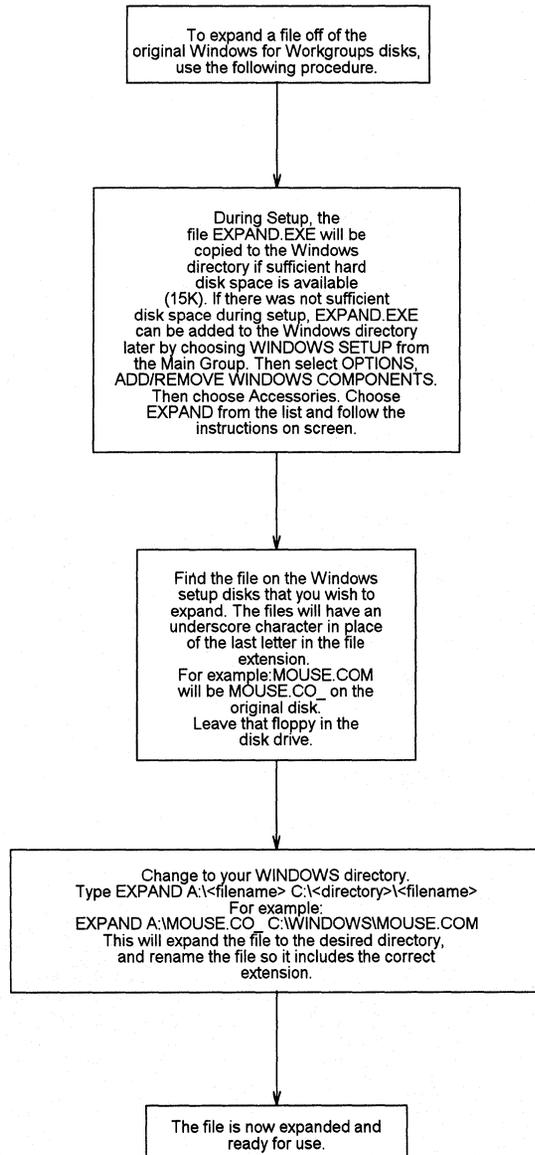


Chart 1.8 Stacker™

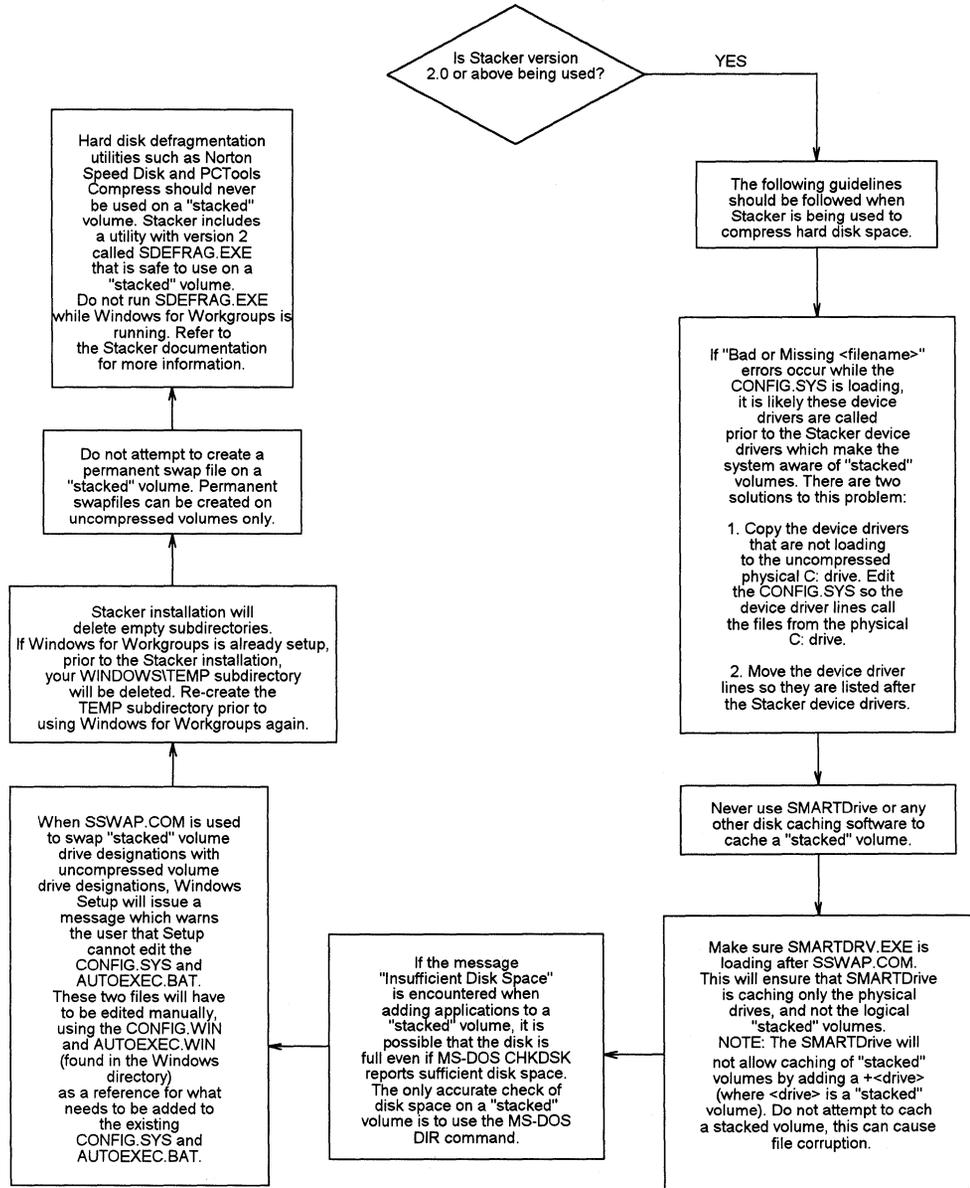


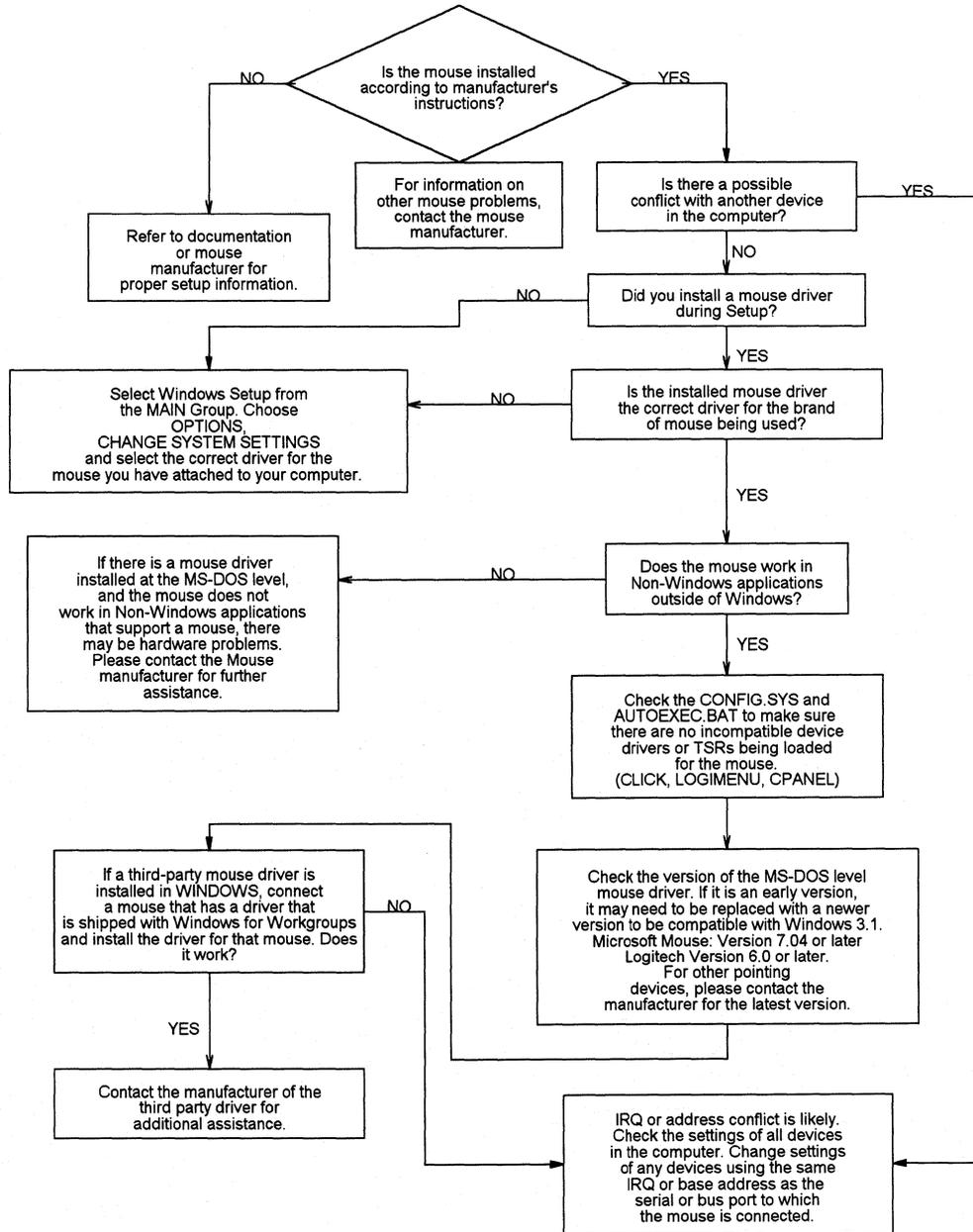
Chart 2.1 Mouse Doesn't Work in Windows for Workgroups

Chart 2.2 Serial Port Problems (Communications and Terminal)

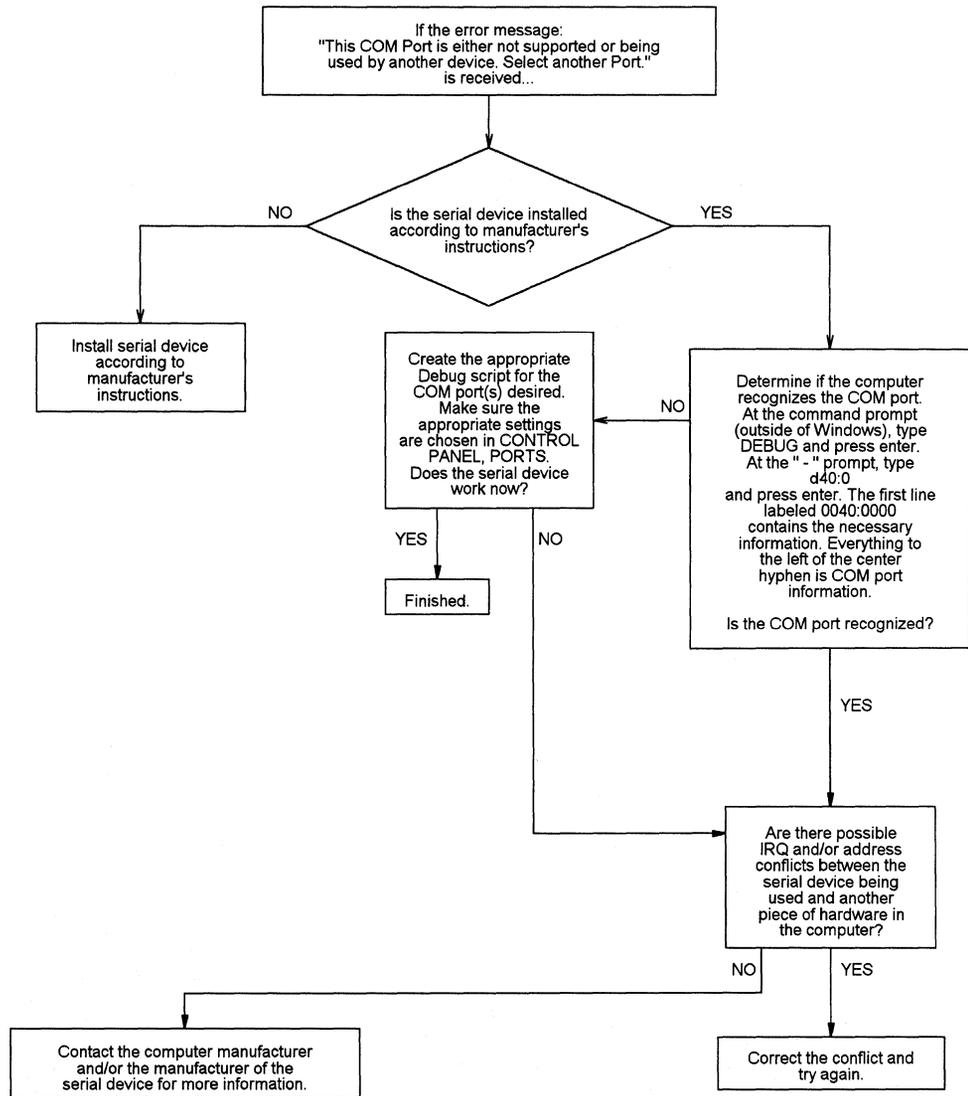


Chart 2.3 Display Adapter Problems (Distorted Video)

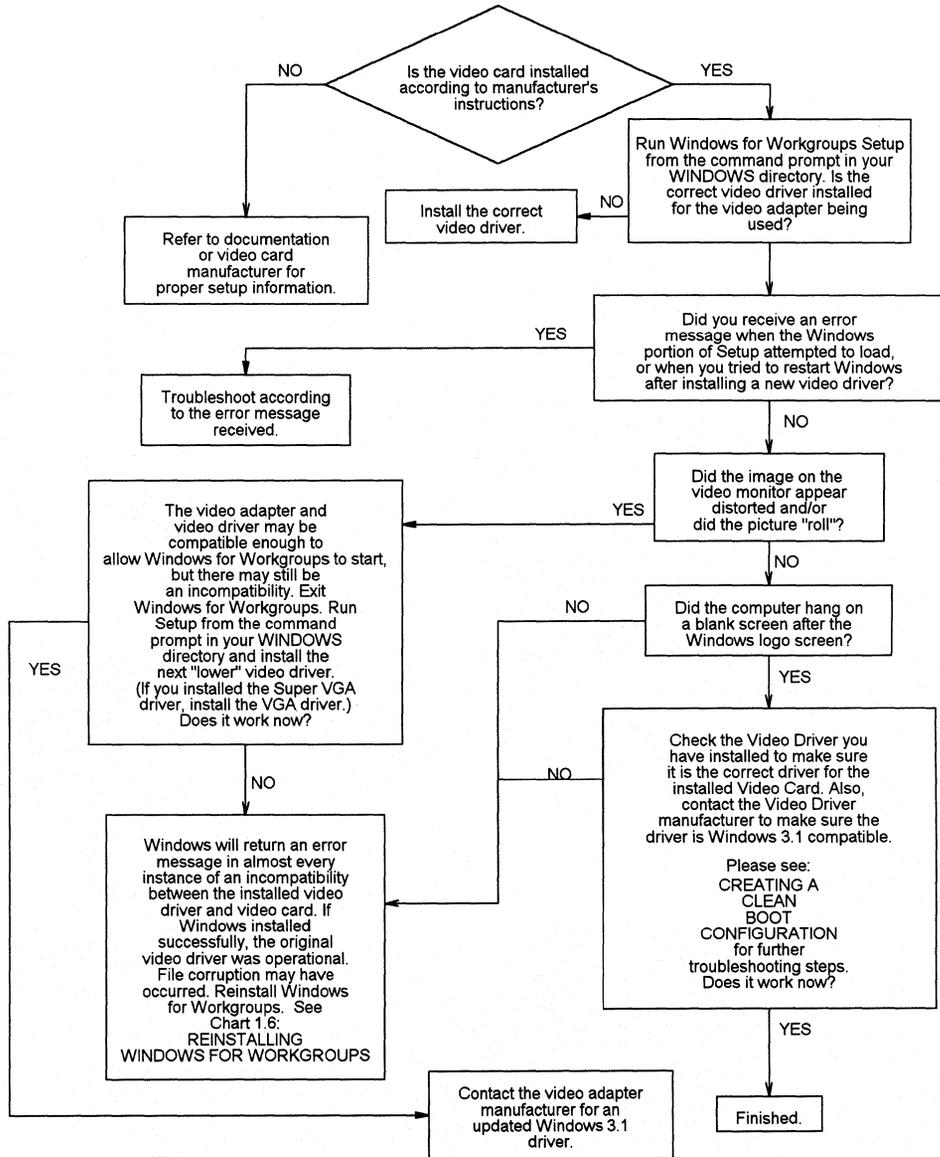
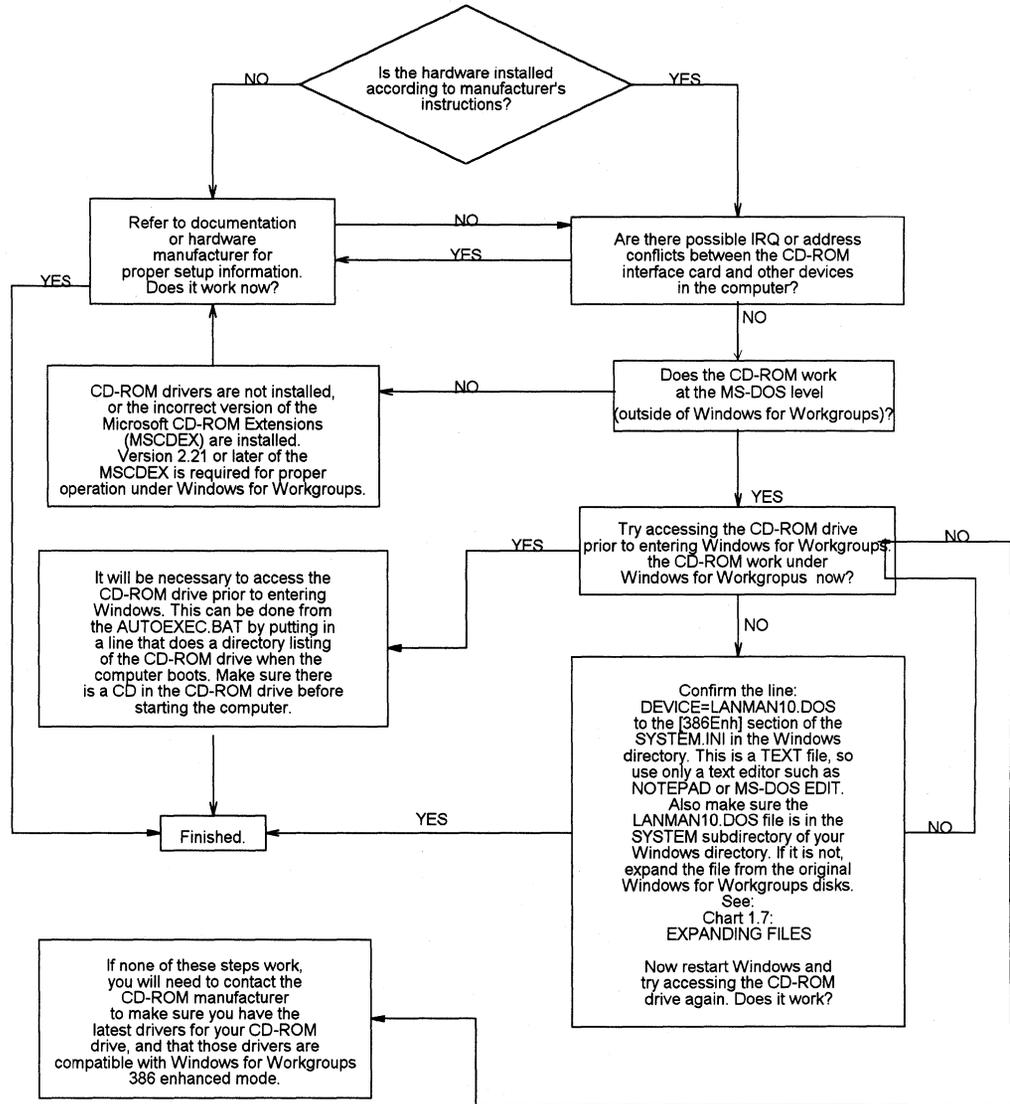


Chart 2.4 Cannot Access CD-ROM



Sharing a CD-ROM drive on a Windows for Workgroups network:

- 1) The MSCDEX must be version 2.21 or above. Windows for Workgroups installs this version on Setup. If not found in the WINDOWS directory: See Chart 1.7 EXPANDING FILES.
- 2) The "/S" switch must be added to the MSCDEX command line to enable sharing.
- 3) The CD-ROM directories have been properly shared in File Manager.
- 4) If using Novell NetWare the MSCDEX must be placed before MSIPX and after Net Start in the AUTOEXEC.BAT.

Chart 2.5A Cannot Access Floppy Drives in 386 Enhanced Mode

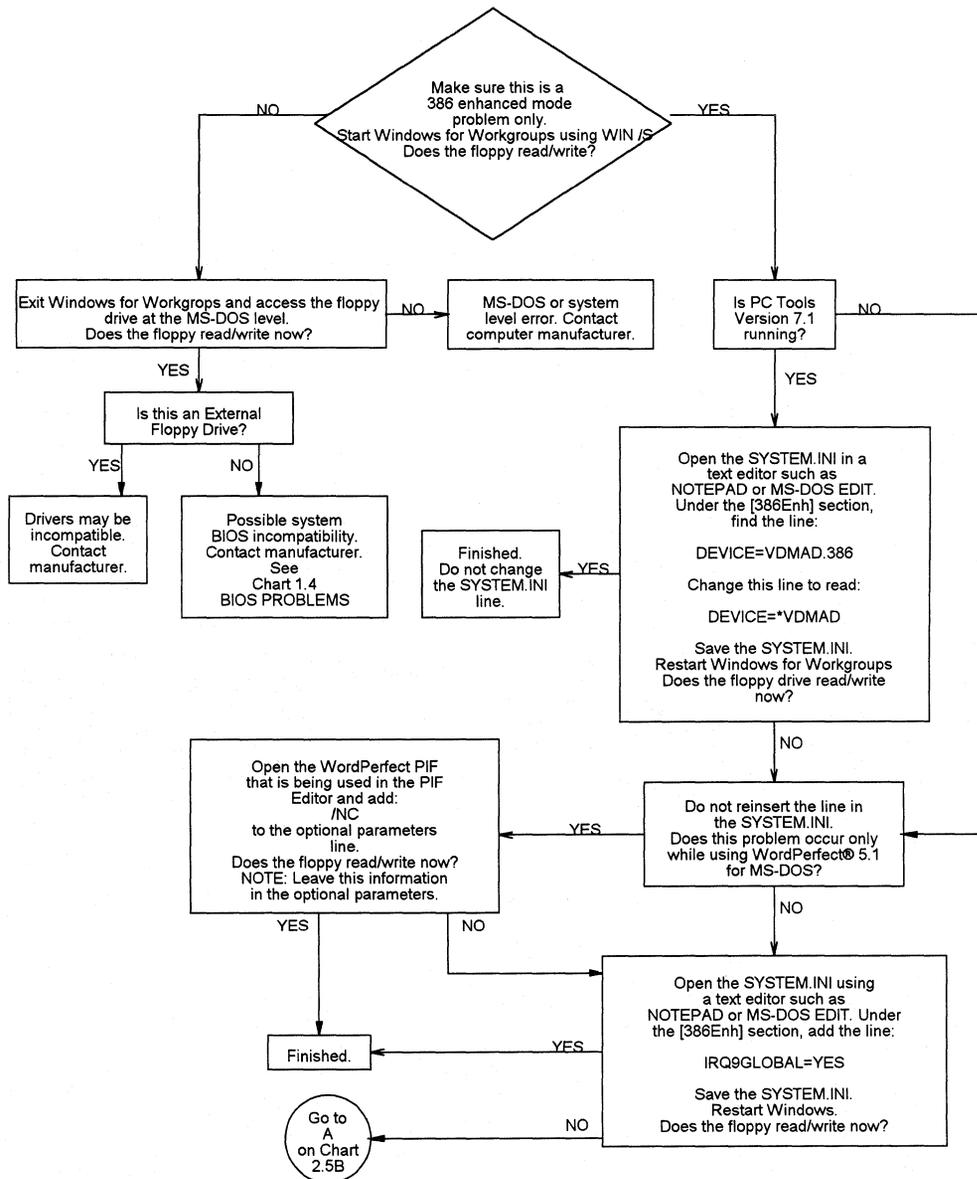


Chart 2.5B **Cannot Access Floppy Drives in 386 Enhanced Mode**

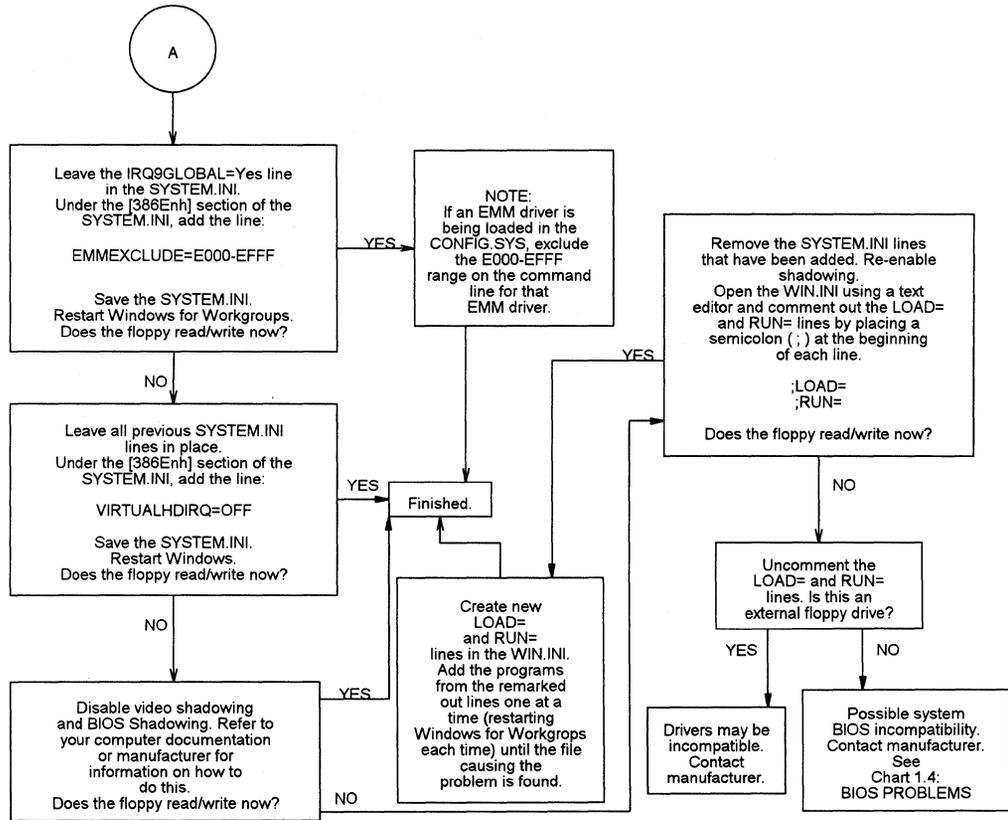


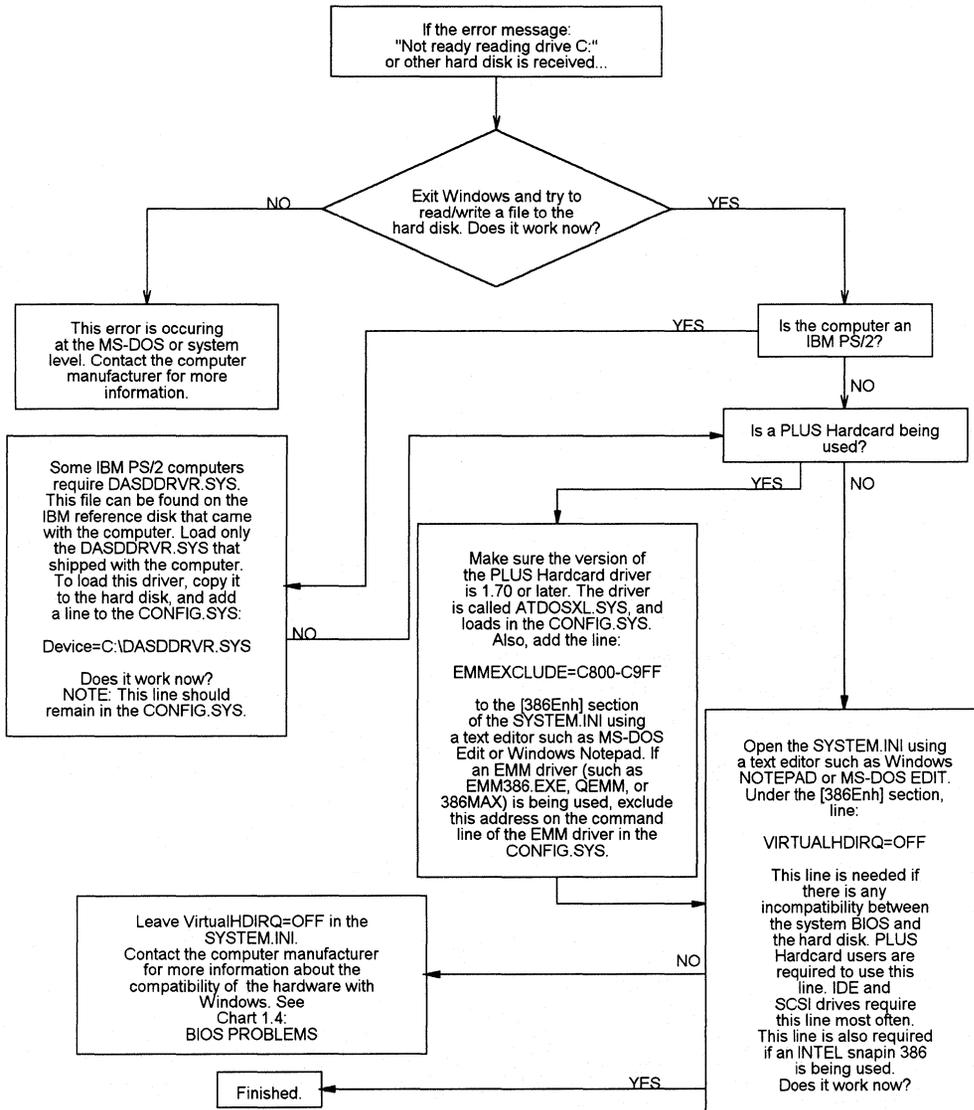
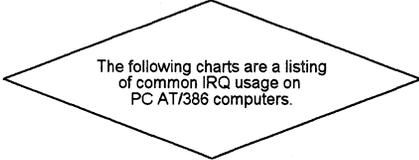
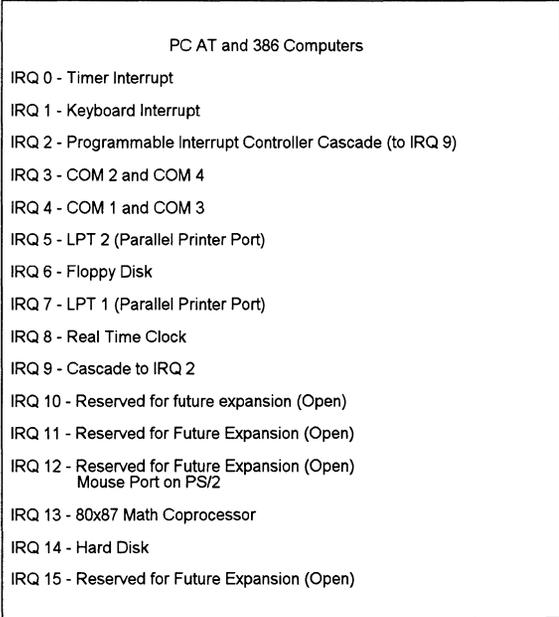
Chart 2.6 Cannot Access Hard Disk in 386 Enhanced Mode

Chart 2.7**Common IRQ Usage for
PC-AT and 80386 Computers**



The following charts are a listing
of common IRQ usage on
PC AT/386 computers.



PC AT and 386 Computers

IRQ 0 - Timer Interrupt
IRQ 1 - Keyboard Interrupt
IRQ 2 - Programmable Interrupt Controller Cascade (to IRQ 9)
IRQ 3 - COM 2 and COM 4
IRQ 4 - COM 1 and COM 3
IRQ 5 - LPT 2 (Parallel Printer Port)
IRQ 6 - Floppy Disk
IRQ 7 - LPT 1 (Parallel Printer Port)
IRQ 8 - Real Time Clock
IRQ 9 - Cascade to IRQ 2
IRQ 10 - Reserved for future expansion (Open)
IRQ 11 - Reserved for Future Expansion (Open)
IRQ 12 - Reserved for Future Expansion (Open)
Mouse Port on PS/2
IRQ 13 - 80x87 Math Coprocessor
IRQ 14 - Hard Disk
IRQ 15 - Reserved for Future Expansion (Open)

Chart 2.8 **Mouse Doesn't Work in Windowed MS-DOS-based Applications**

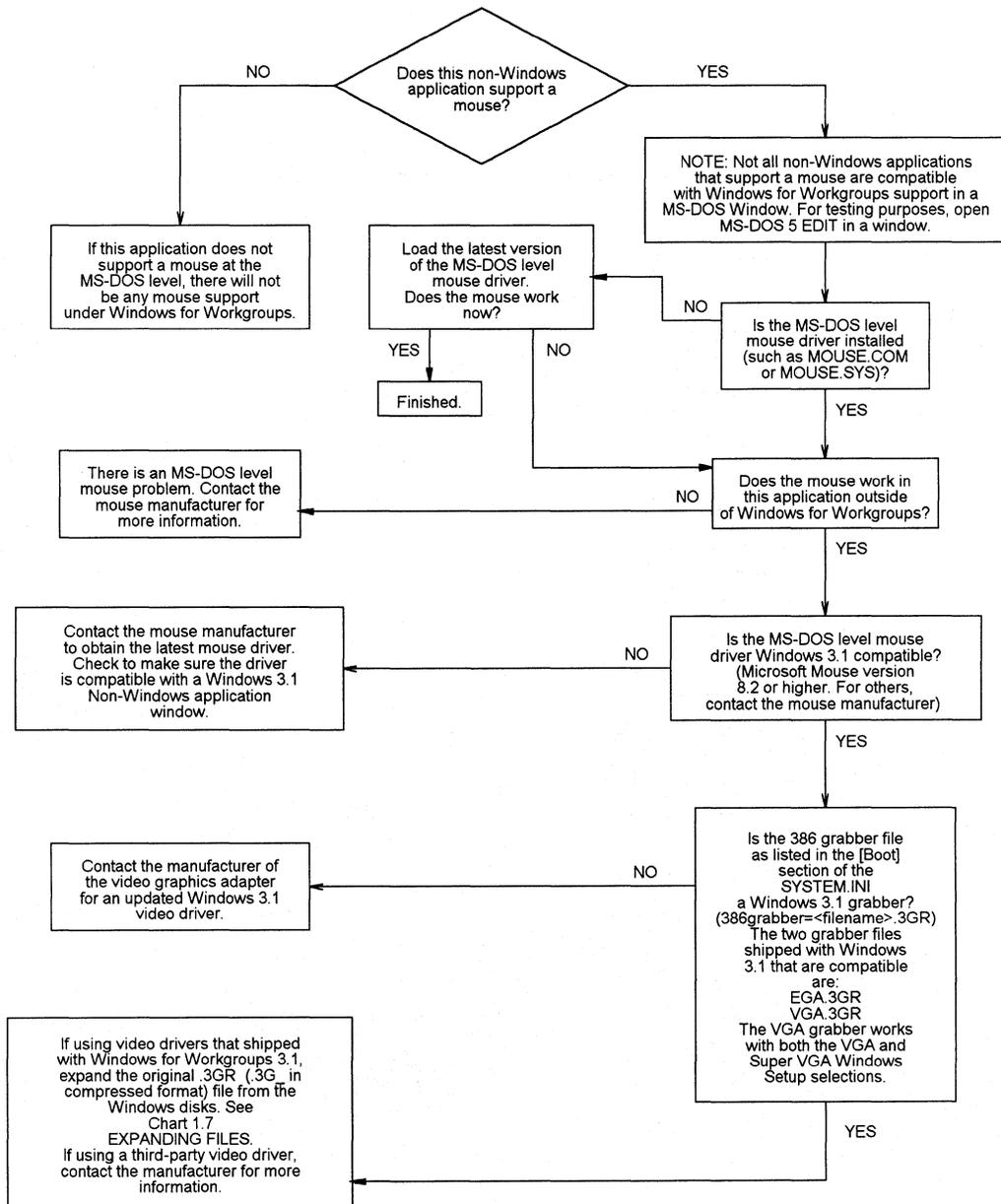


Chart 3.1 TrueType Fonts

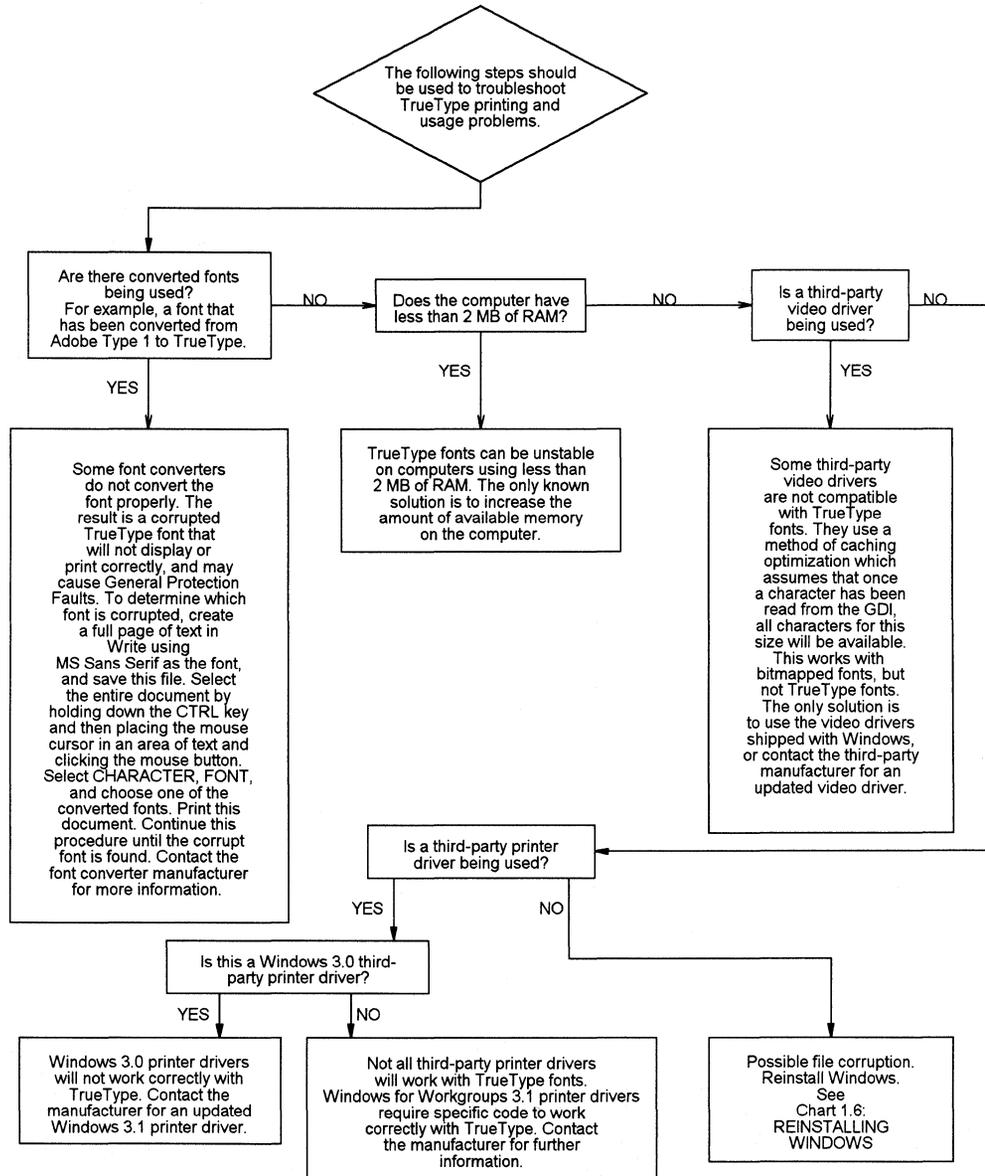


Chart 3.2 Disabling Adobe Type Manager

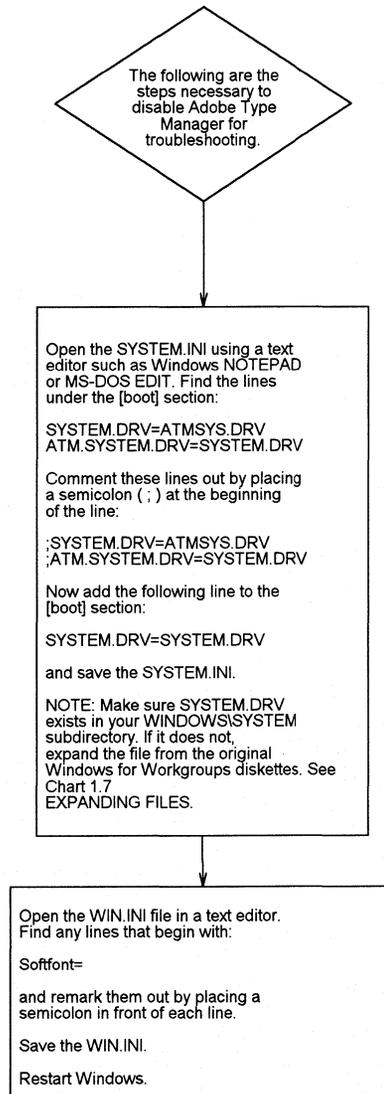


Chart 3.3 Disabling Bitstream FaceLift

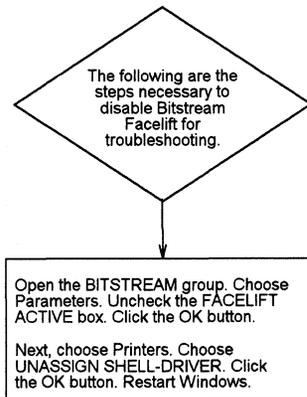


Chart 3.4 Disabling HP Intellifont

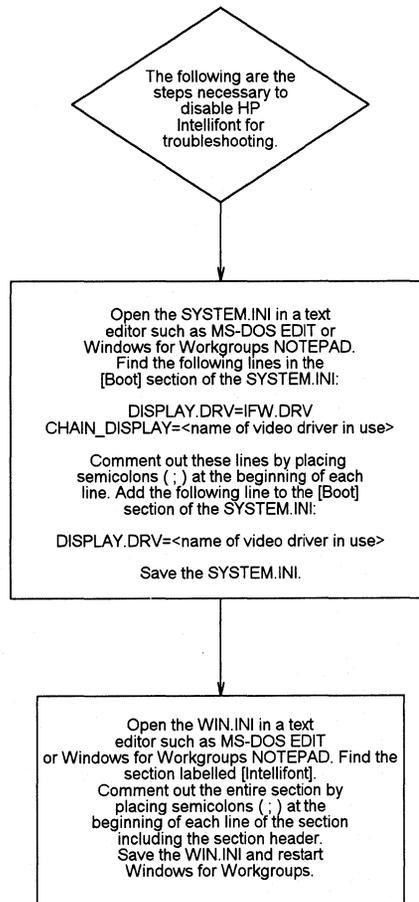


Chart 4.1 Cannot Print on Standalone PC

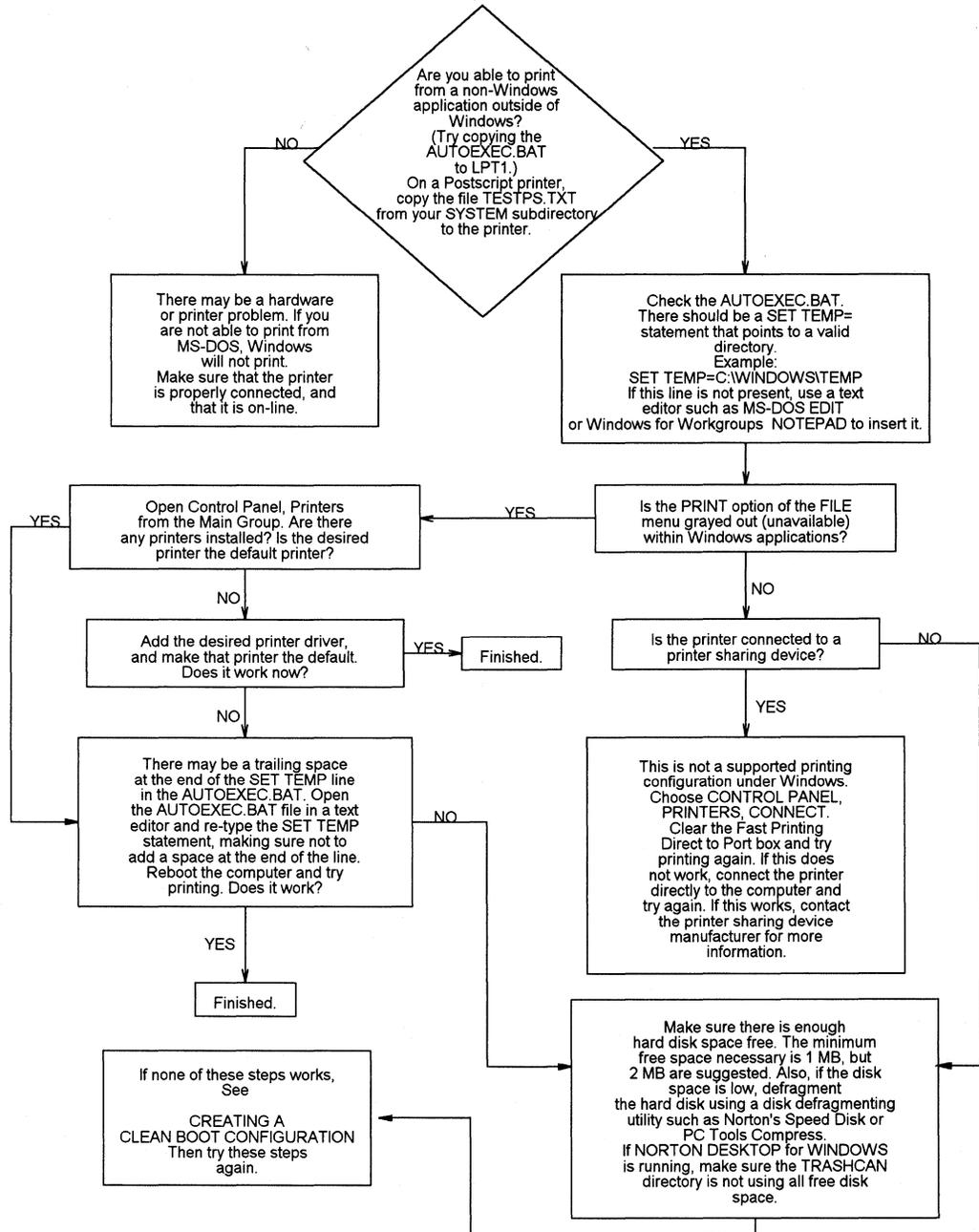


Chart 4.2 Cannot Print on Network

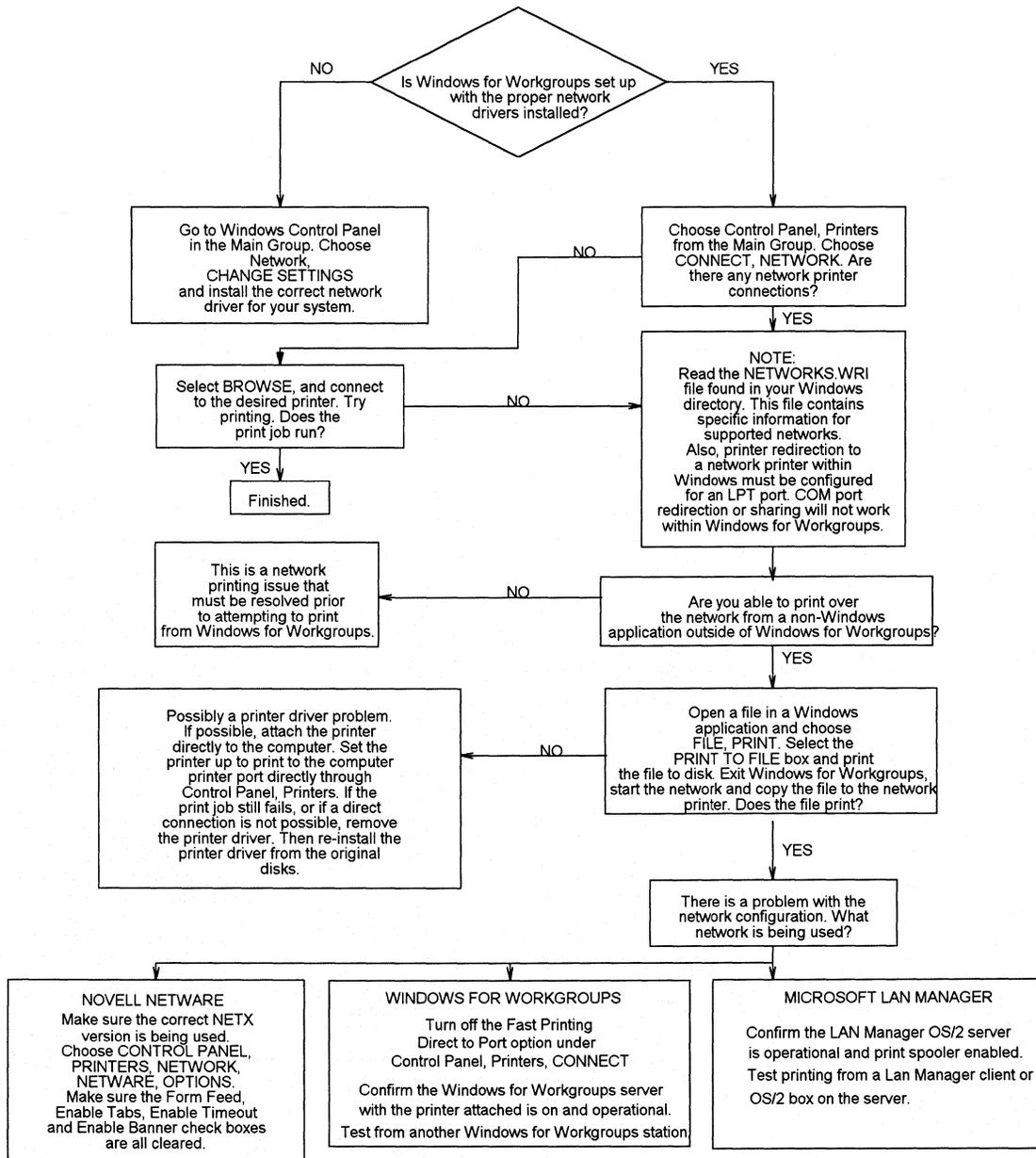


Chart 4.3 Garbled Output

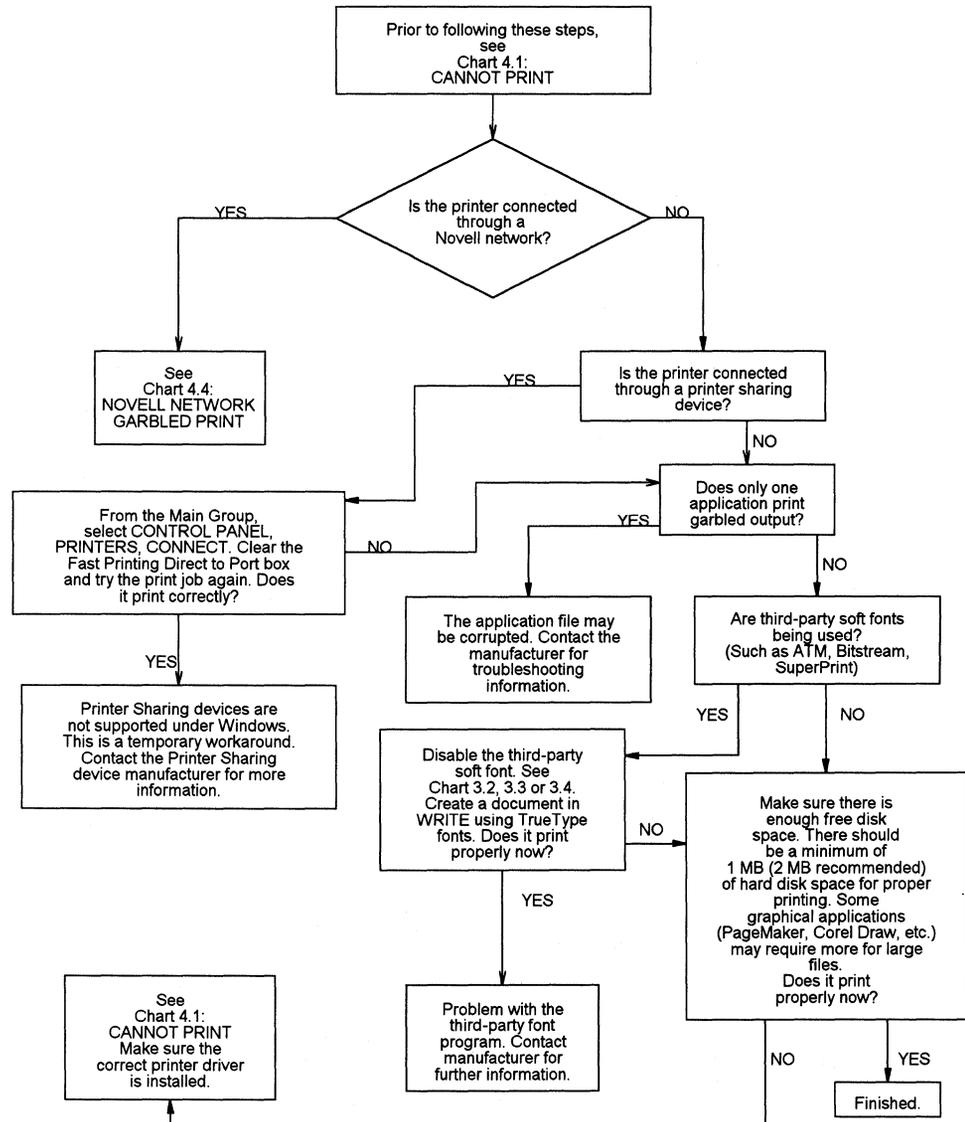


Chart 4.4 Garbled Output on Novell Network

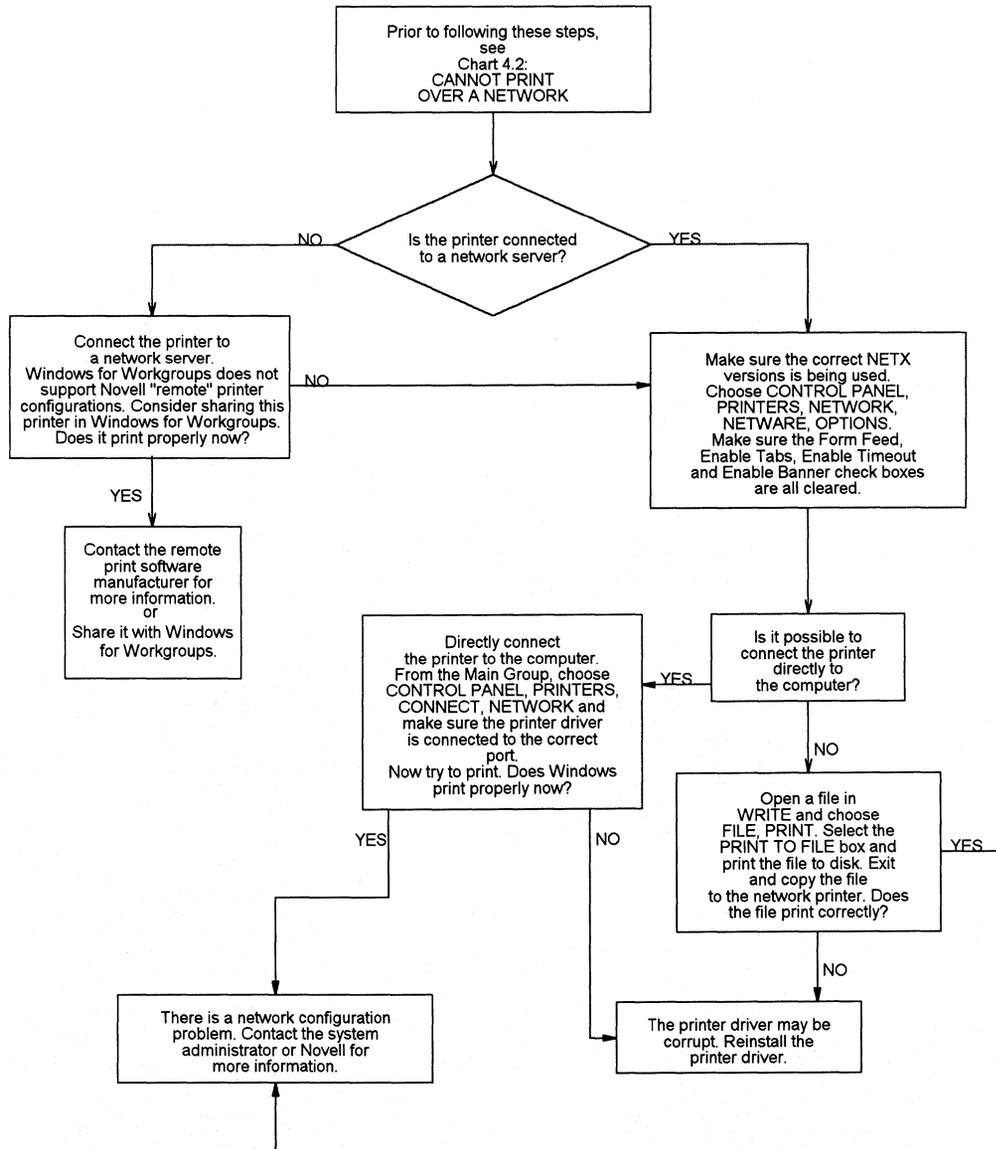


Chart 5.1A Cannot Run 386 Enhanced Mode

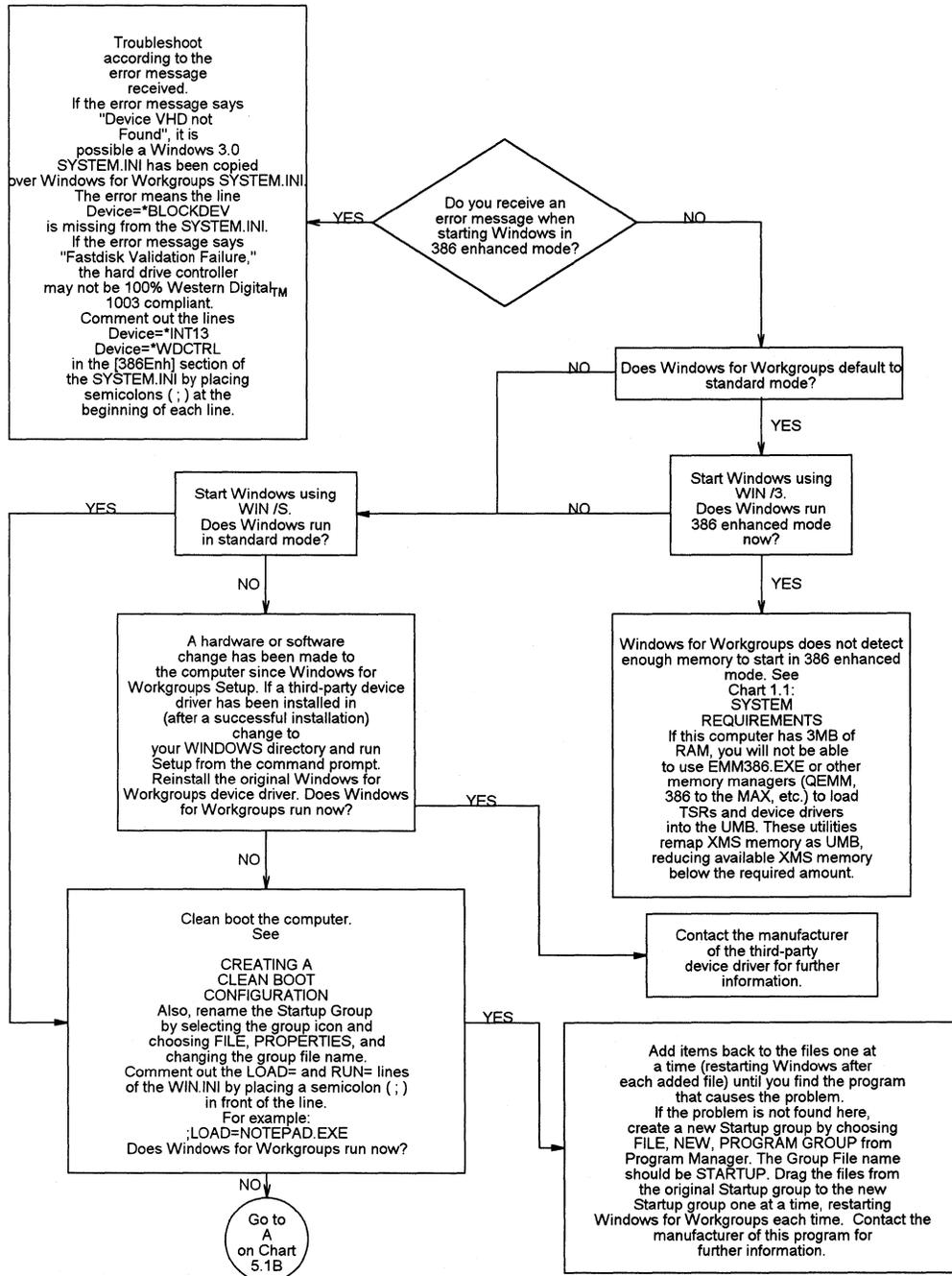


Chart 5.1B Cannot Run 386 Enhanced Mode

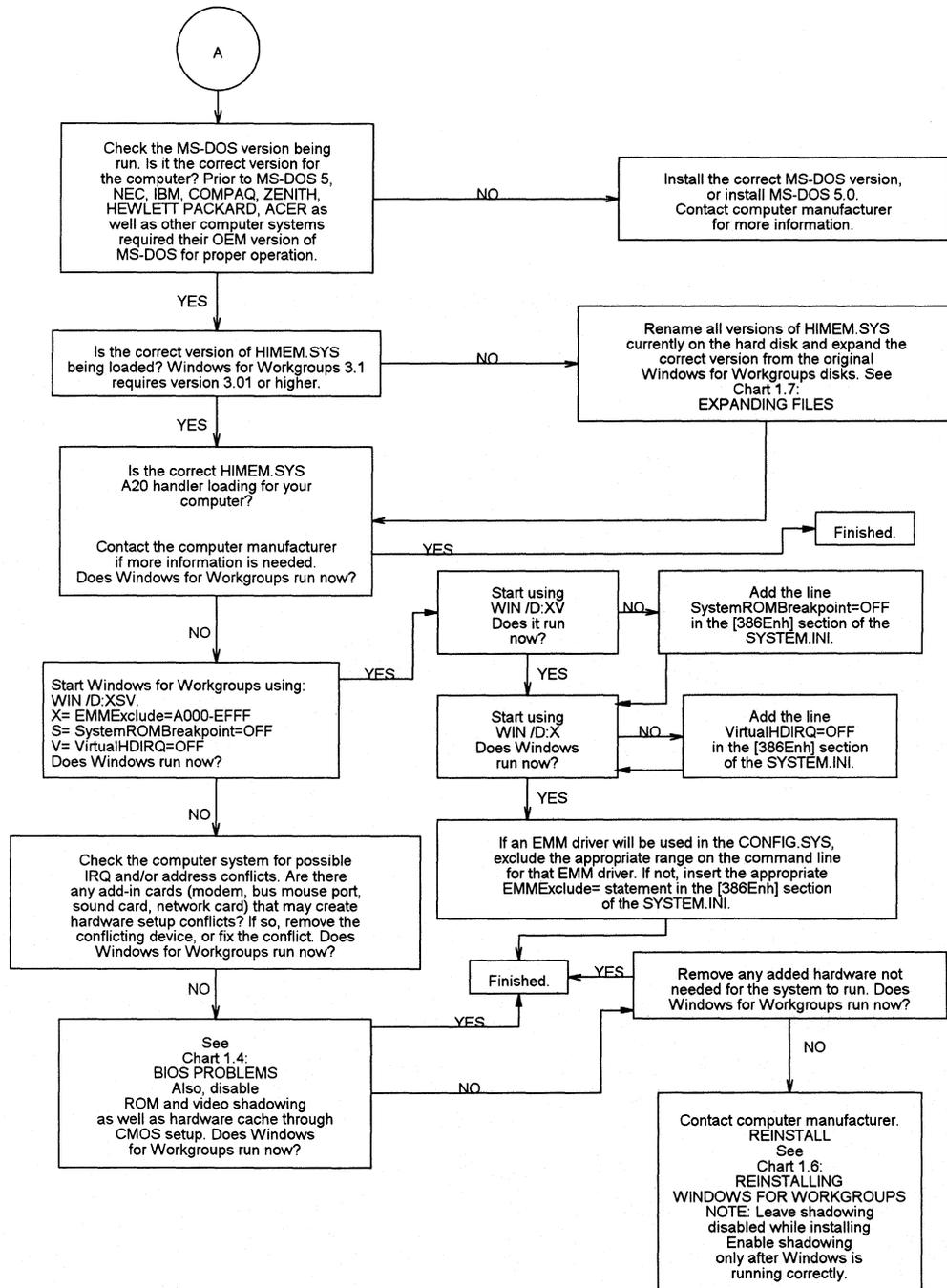


Chart 5.2

This page left intentionally blank

Chart 5.3A General Protection Faults

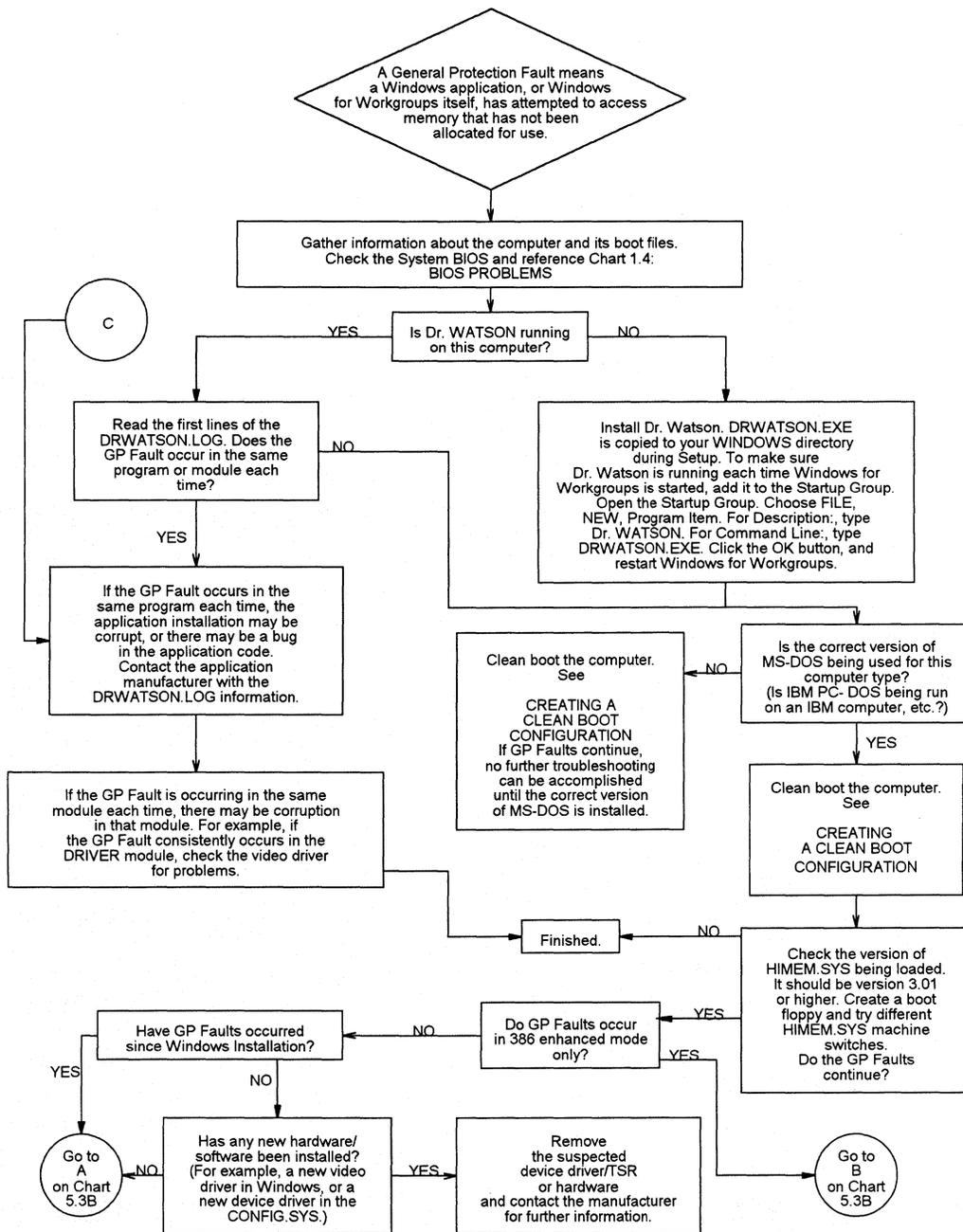


Chart 5.3B General Protection Faults

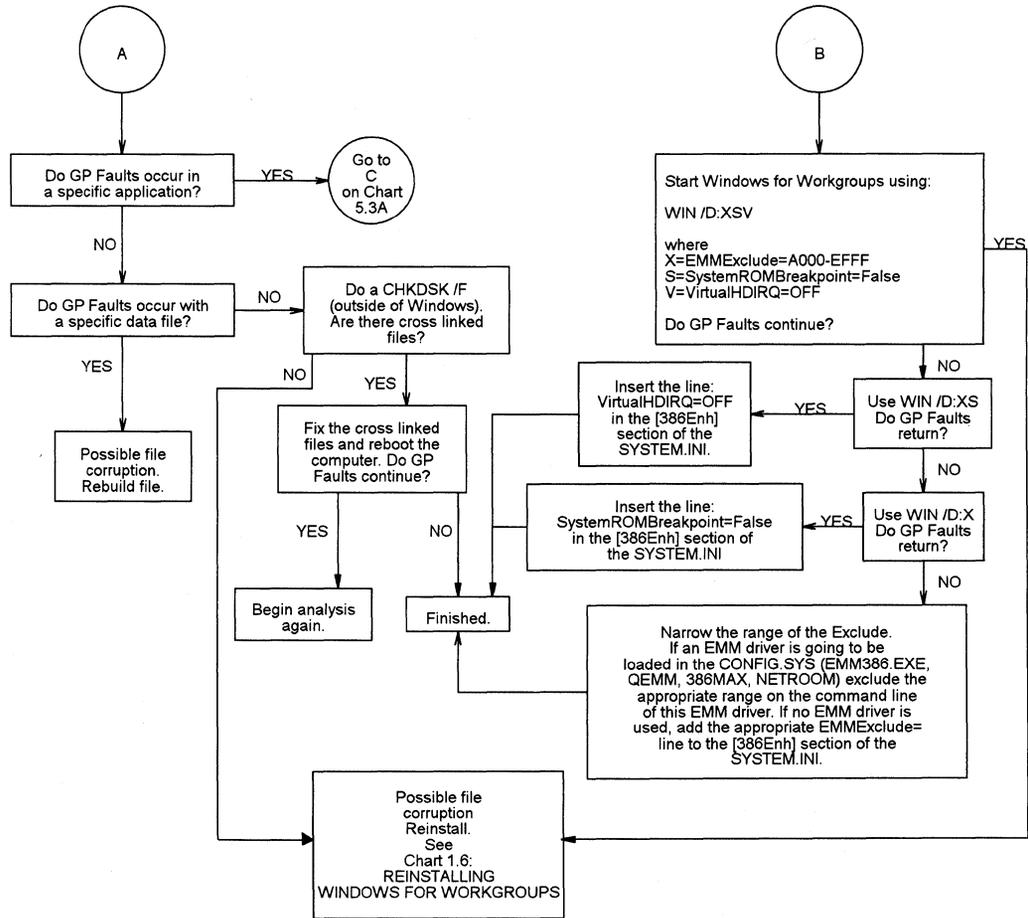


Chart 5.4 Application Execution Error

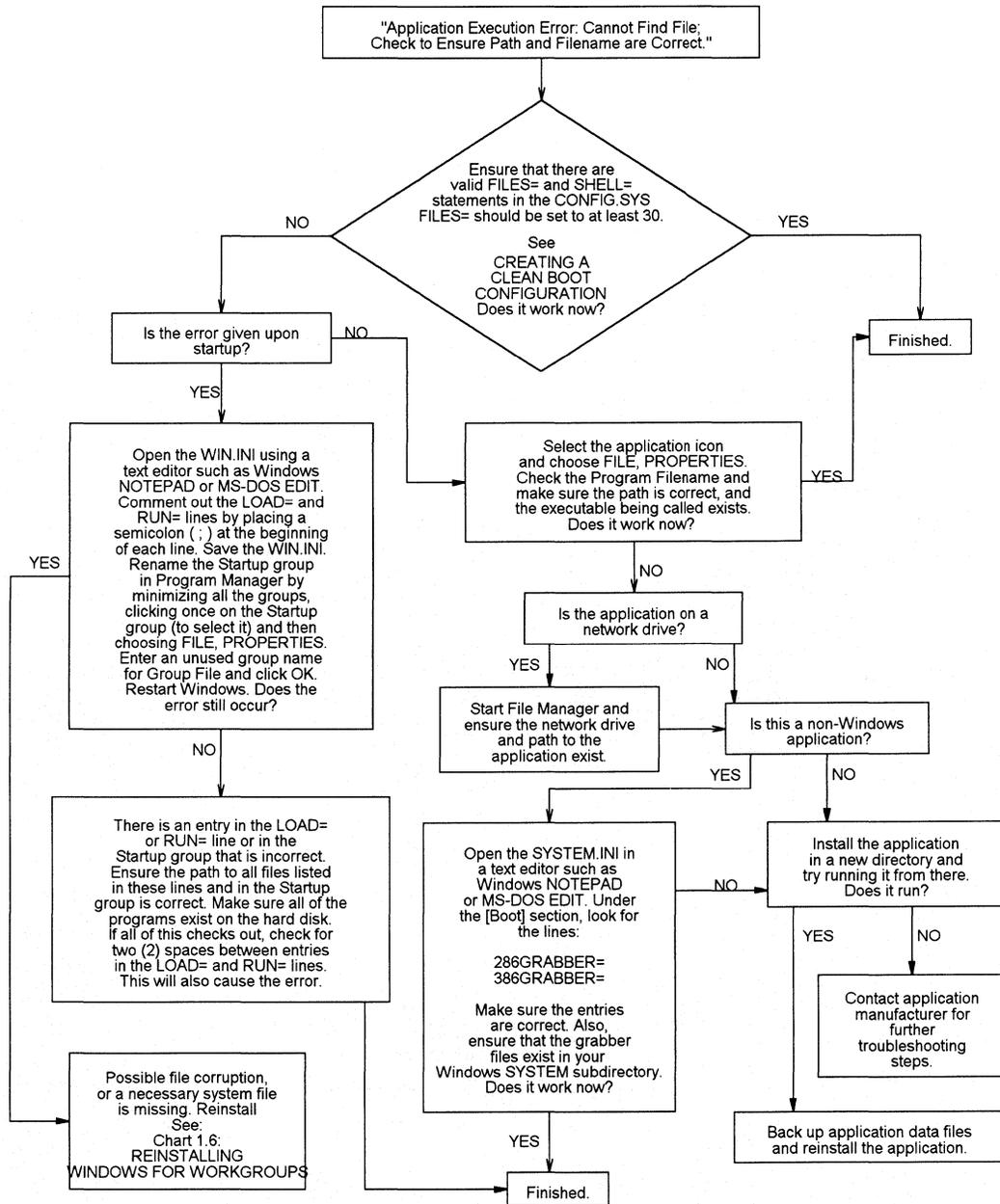


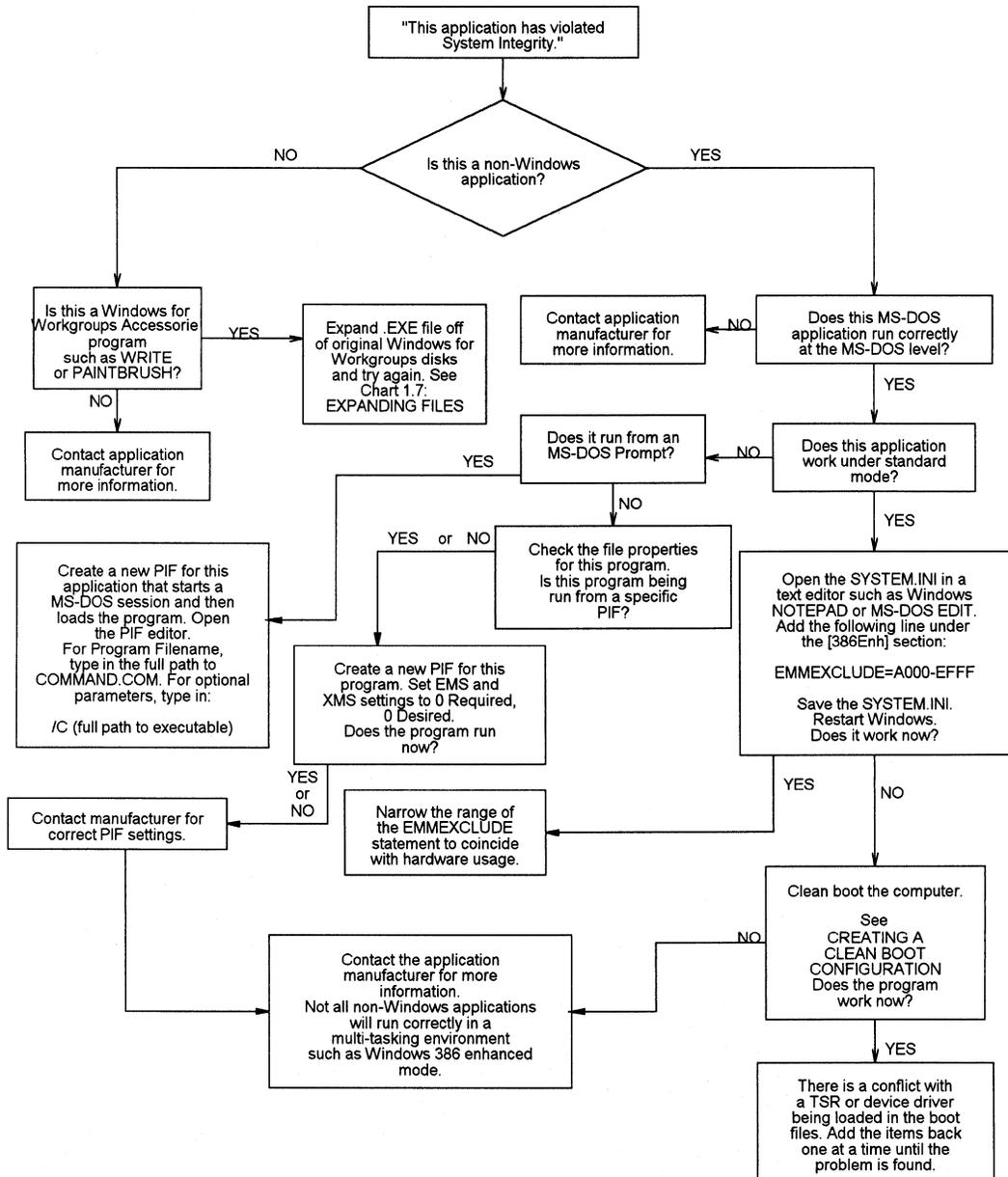
Chart 5.5 System Integrity Violation in 386 Enhanced Mode

Chart 5.6 Disabling Norton Desktop for Windows



NOTE: Check the dates of the NDW files. For Norton Desktop to operate correctly with Windows 3.1, 7/24 or 7/25 are required. If the dates on the files are earlier than 7/24, contact Symantec for an update.

Open the SYSTEM.INI using a text editor such as Windows NOTEPAD or MS-DOS EDIT. Find the line under the [boot] section:

```
SHELL=NDW.EXE
```

Comment this line out by placing a semicolon (;) at the beginning of the line:

```
;SHELL=NDW.EXE
```

Now add the following line to the [boot] section:

```
SHELL=PROGMAN.EXE
```

and save the SYSTEM.INI. Restart Windows.

Chart 5.7 Disabling PC-Tools Version 7

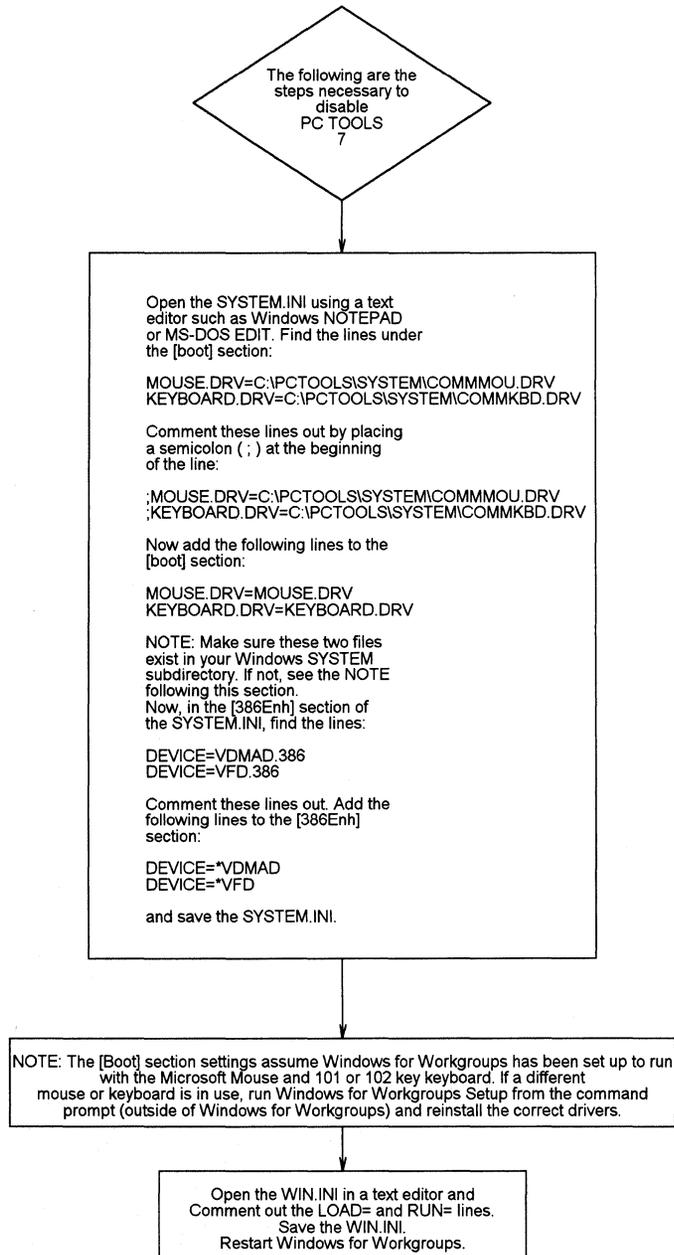


Chart 5.8 Out-of-Memory Errors in Windows for Workgroups

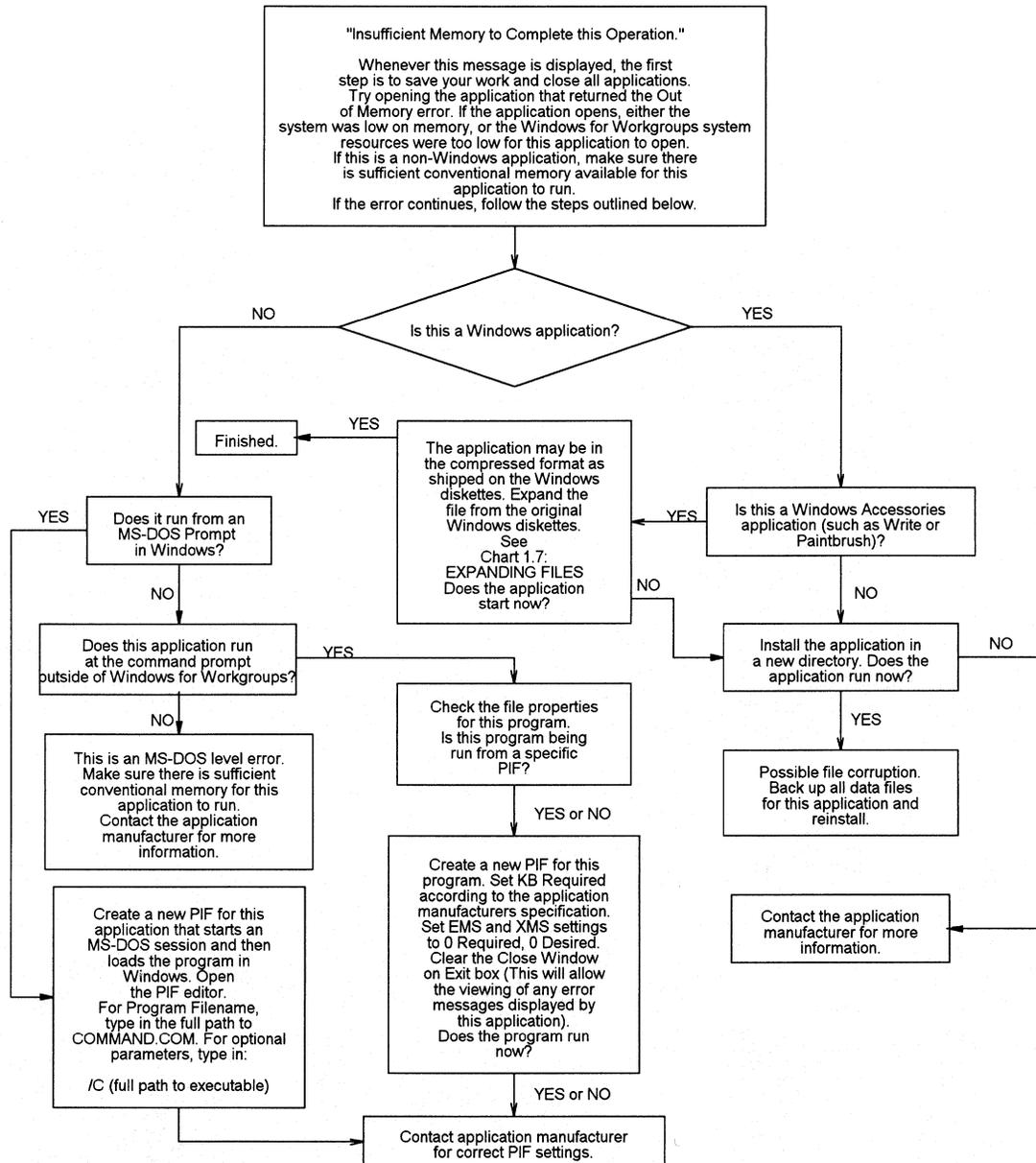


Chart 5.9 **Cannot Run Windows for Workgroups in Standard Mode**

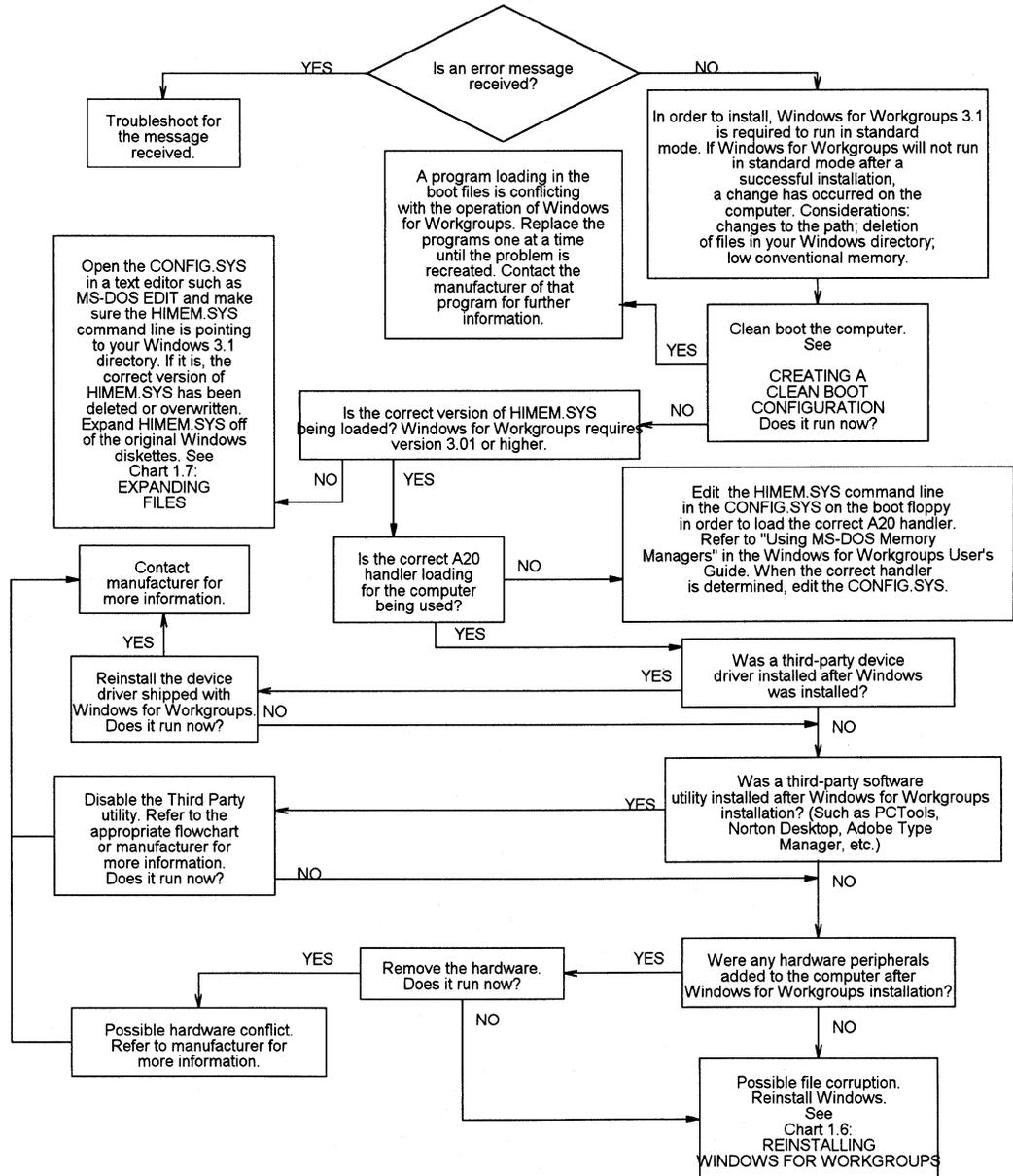


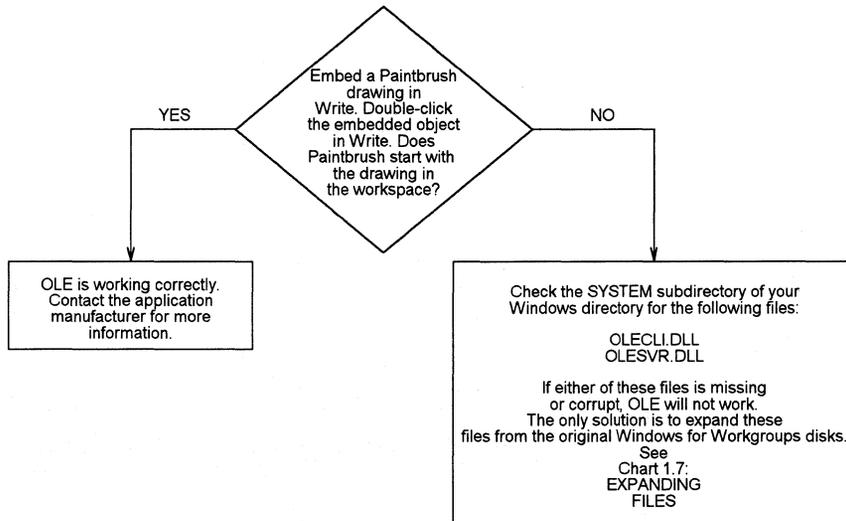
Chart 5.10 Problems with Object Linking and Embedding

Chart 6.1 Multimedia Drivers

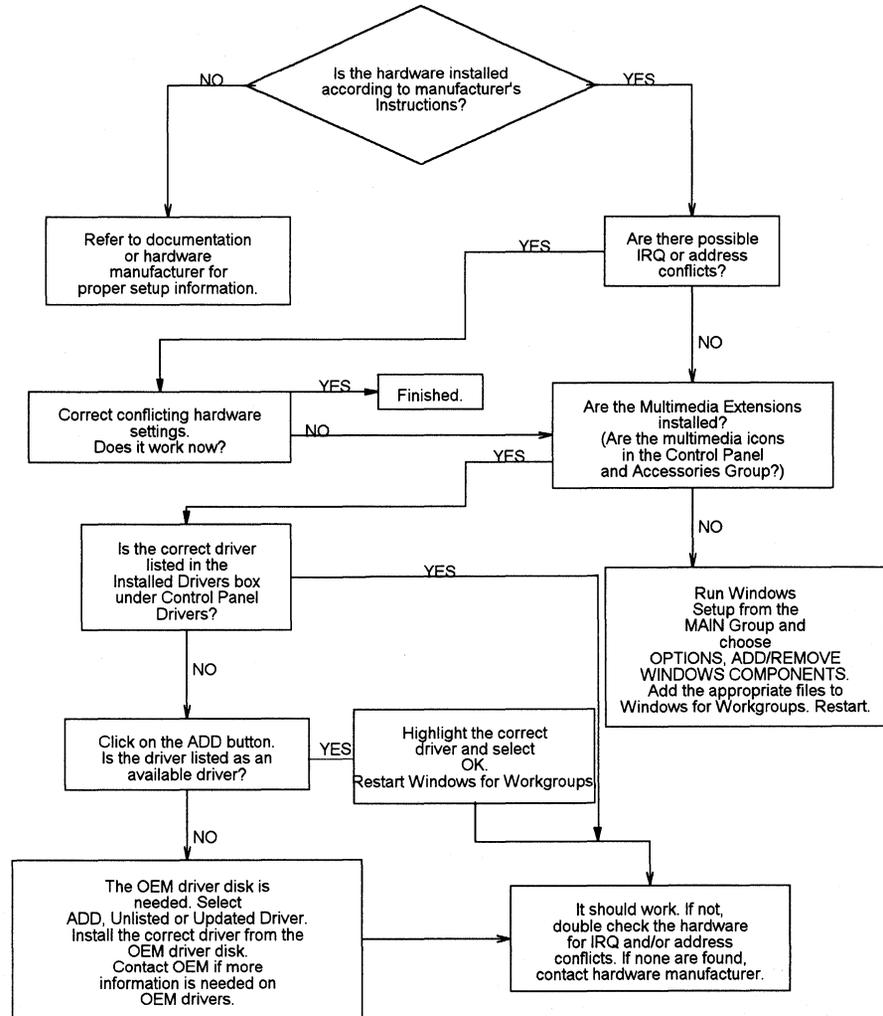


Chart 6.2 Media Player

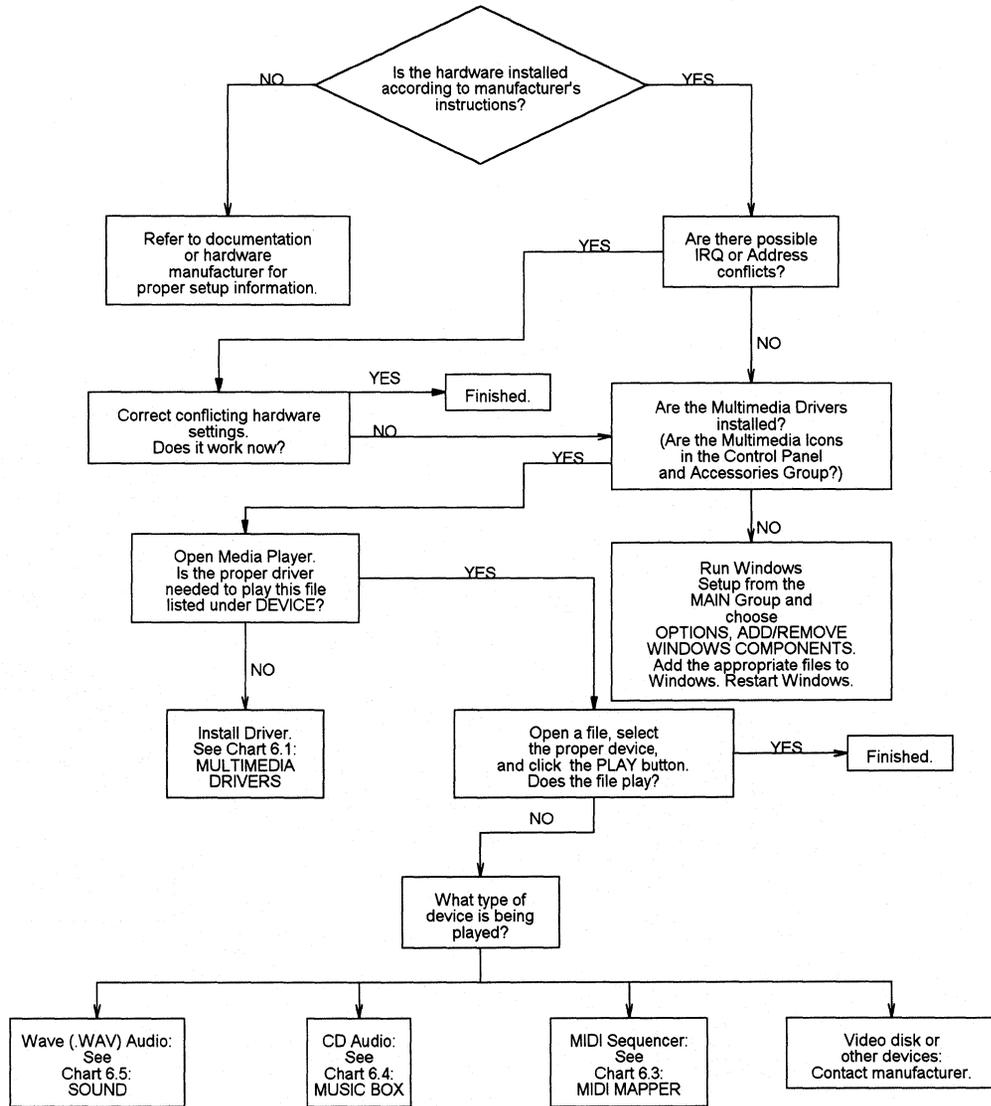


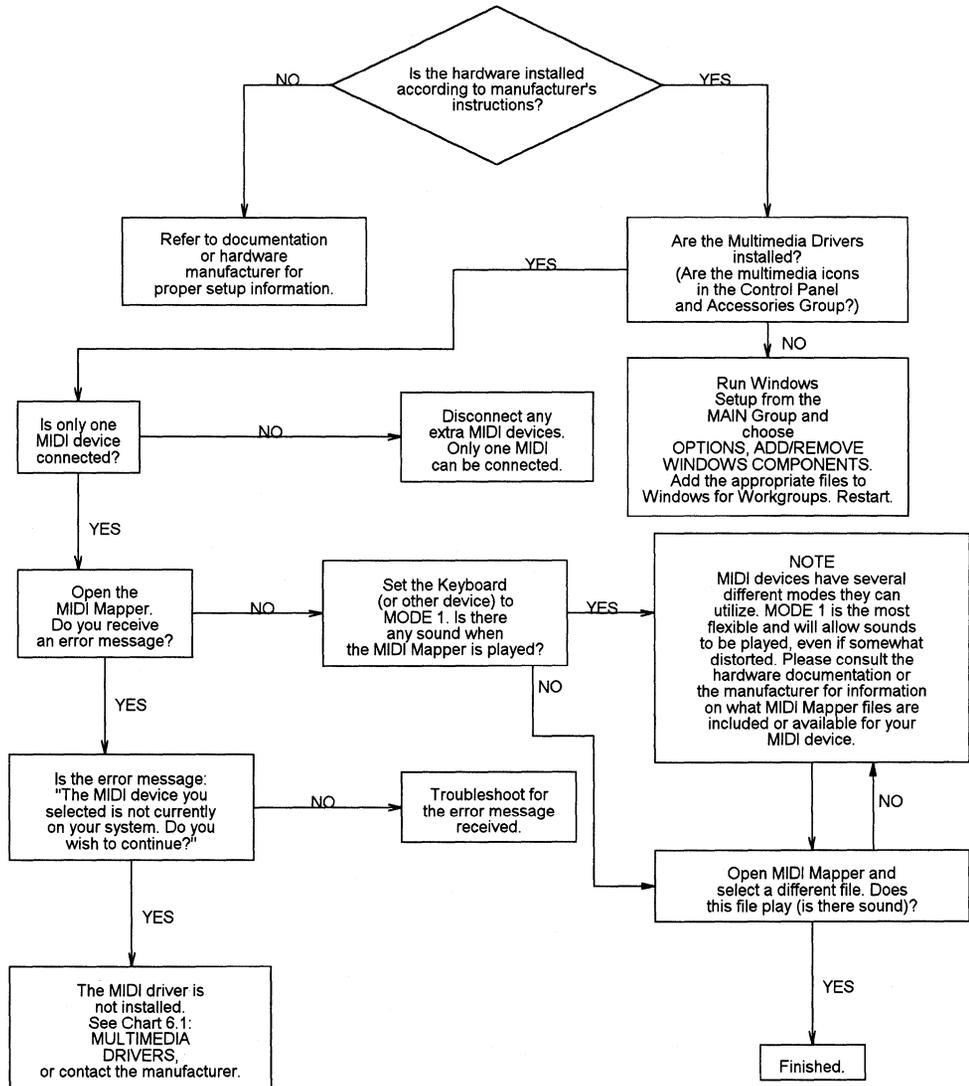
Chart 6.3 **MIDI Mapper**

Chart 6.4 Music Box

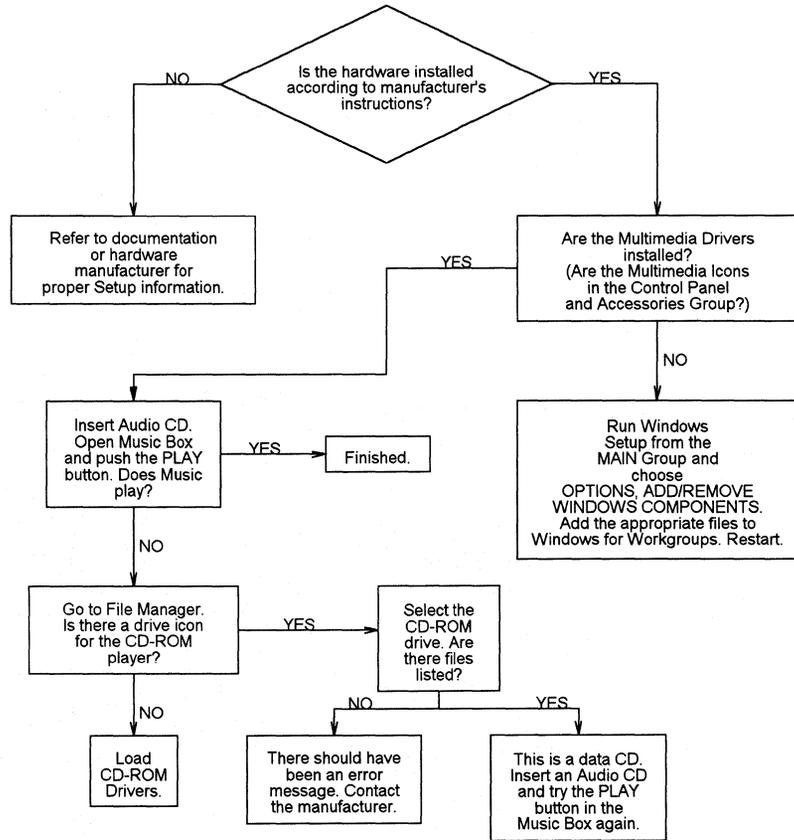
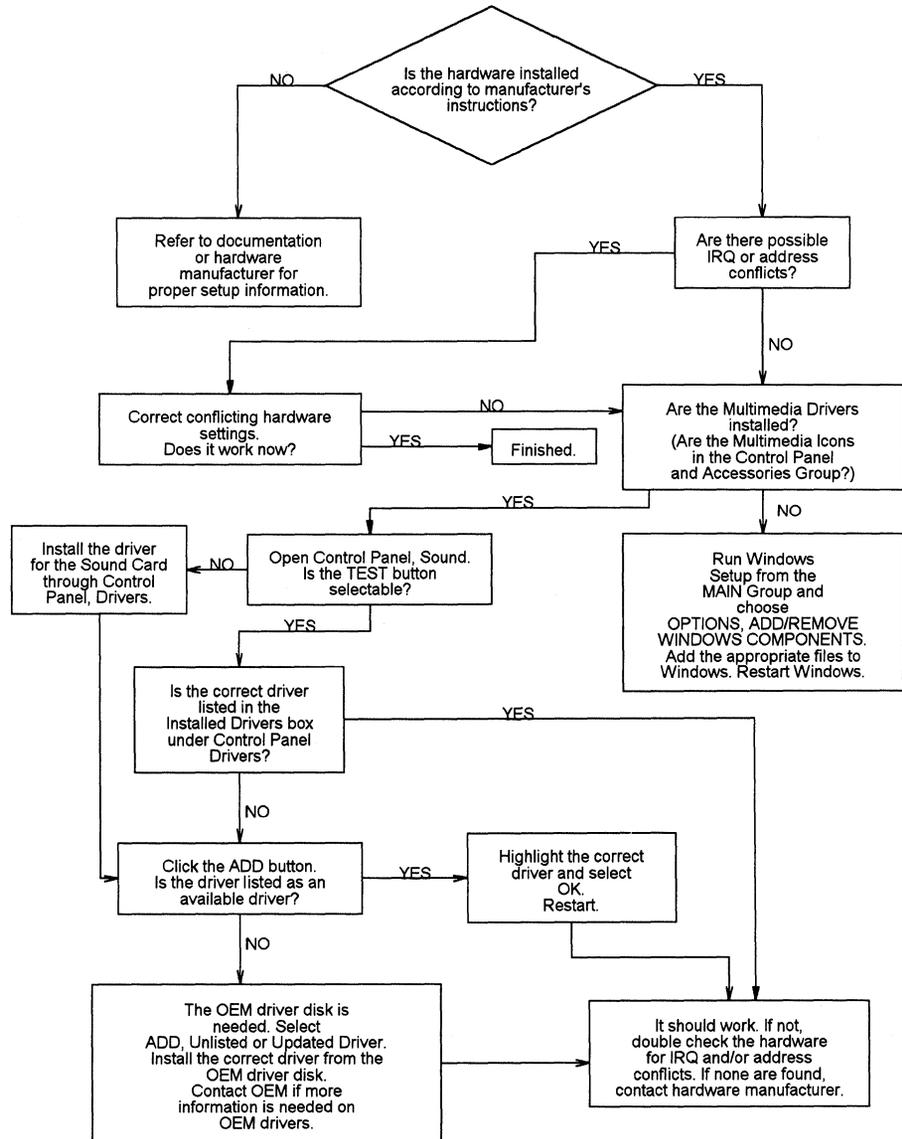


Chart 6.5 Sound



More About MS-DOS, Microsoft Windows for Workgroups and Personal Computers

Print References:

- *Windows for Workgroups User's Guide* and the *Getting Started* booklet in your Windows for Workgroups 3.1 package.
- *Microsoft MS-DOS 5.0 User's Guide and Reference* and the *Getting Started* booklet in your MS-DOS 5.0 package.
- *Computer Dictionary*. Microsoft Press, 1991.
- Gookin, Dan. *The Microsoft Guide to Managing Memory with DOS 5*. Microsoft Press, 1991.

Online References:

- Windows for Workgroups 3.1 online help contains instructions and definitions of terms related to the screen elements in Microsoft Windows for Workgroups. To get online help, press F1 or choose a command from the Help menu in Program Manager or File Manager.

Support Services and Training:

- See the information in your Windows for Workgroups package about how to contact Microsoft Product Support Services and how to use the Microsoft Forums on CompuServe.
- See Appendix A, "Windows for Workgroups Resource Directory," in the *Windows for Workgroups Resource Kit* for information about Microsoft University, and other sources of information and training.

Also, in Appendix A, "Windows for Workgroups Resource Directory," there is an extensive list of reference materials.

Accessibility for Individuals with Disabilities

Microsoft is committed to making its products and services easier to use—for everyone. This appendix provides information about the following products and services available for computer users with special needs:

- Support services for people who are deaf or hard-of-hearing
- Access Pack for Windows, a product that makes Windows more accessible for people with motion or hearing disabilities
- Keyboard layouts designed for people who type with a single hand
- Microsoft documentation on audio cassettes and floppy disks
- Types of products available for blind or low vision individuals
- Hints for customizing Windows for Workgroups
- How to get more information about products and services for people with disabilities

Note The information in this appendix applies only to Windows for Workgroups users in the United States. If you are outside the United States, your Windows for Workgroups package contains a subsidiary information card listing product-support telephone numbers and addresses. You can contact your subsidiary to find out whether the type of products and services described in this appendix are available in your area.

Contents of This Appendix

Microsoft Support Services for the Deaf and Hard-of-Hearing	D-2
Access Pack for Windows	D-2
Keyboard Layouts for Single-Handed Users	D-3
Documentation on Audio Cassettes and Floppy Disks	D-3
Products for Blind and Low Vision Individuals	D-4
Customizing Windows for Workgroups	D-4
Getting More Information	D-6

Microsoft Support Services for the Deaf and Hard-of-Hearing

Microsoft provides a wide variety of services for customers who are deaf or hard-of-hearing through a text telephone service. This service gives Microsoft deaf and hard-of-hearing customers access to the same technical product and customer support as hearing customers.

You can contact Microsoft with a text telephone (TT or TDD) by dialing 1-206-635-4948. Microsoft's product-support services are subject to Microsoft's prices, terms, and conditions in place at the time the service is used.

Access Pack for Windows

Microsoft distributes Access Pack for Windows, which provides users with movement or hearing disabilities better access to computers running Windows or Windows for Workgroups. Access Pack for Windows contains several features that do the following:

- Allows single-fingered typing of SHIFT, CTRL and ALT key combinations.
- Ignores accidental key presses.
- Adjusts or disables the rate at which a character is repeated on the screen when a user holds down a key.
- Prevents extra characters if the user unintentionally presses a key more than once.
- Allows the user to control the mouse cursor using the keyboard.
- Allows control of the computer's keyboard and mouse using an alternate input device.
- Provides a visual cue when a computer makes sounds.
- Enables you to turn off these services for users that do not need them.

Access Pack for Windows, which was developed by the Trace R&D Center at the University of Wisconsin-Madison, is available on the Windows Driver Library. If you have a modem, you can download Windows Driver Library components from network services, including the following:

- CompuServe
- GENie
- Microsoft OnLine
- Microsoft Download Service (MSDL), which you can reach by calling 1-206-936-MSDL (6735).
- Various user-group bulletin boards (such as the bulletin-board services on the Association of PC User Groups network)

Users within the United States who do not have a modem can order disks by calling Microsoft Customer Support Services at 1-800-426-9400 or 1-206-635-4948 (text telephone).

Keyboard Layouts for Single-Handed Users

Microsoft distributes data files that contain keyboard layouts for people who have difficulty using the standard QWERTY layout. You can use these files to change your keyboard configuration without having to adjust your physical keyboard.

Windows for Workgroups already supports the Dvorak keyboard layout, which can be useful for coping with or avoiding types of repetitive motion injuries associated with typing. The two new files support new layouts optimized for persons who type with the left hand only or with the right hand only. They can also be useful for people who type with a single finger or a wand.

These layouts are included on the Windows Driver Library. For information about obtaining the Windows Driver Library, see the preceding section.

Documentation on Audio Cassettes and Floppy Disks

People who cannot use printed documentation can obtain most of Microsoft's publications from Recording for the Blind, Inc. (RFB). RFB distributes these documents to registered members of their distribution service, either on audio cassettes or on floppy disks. RFB's collection contains more than 80,000 titles,

including Microsoft product documentation and books from Microsoft Press. You can contact Recording for the Blind at the following address:

Recording for the Blind, Inc.
20 Roszel Road
Princeton, NJ 08540

Phone: 1-800-221-4792
Fax: 1-609-987-8116

Products for Blind and Low Vision Individuals

There are numerous products available to let blind or low vision individuals use Windows. For persons with low vision there are several screen enlargement utilities, and for blind individuals there are screen readers which provide alternative output by synthesized voice or refreshable Braille displays.

For more information on the various products available, see the “Getting More Information” section presented later in this appendix. For more information about customizing Windows for Workgroups for low vision individuals, see the following section.

Customizing Windows for Workgroups

There are many ways you can adjust the appearance and behavior of Windows for Workgroups to suit varying eyesight and motor skills, without requiring any additional software or hardware.

For more information on the initialization file entries listed here, see Chapter 6, “Windows for Workgroups Initialization Files.” For more information on using the Windows Control Panel, see the *Windows for Workgroups User's Guide*.

You can use the following methods to adjust the appearance of Windows for Workgroups:

- If you have trouble differentiating the system colors, or if they cause eyestrain, you can adjust them using the Colors section of Control Panel and additional entries in the [colors] section of WIN.INI.

- If you have trouble reading the Windows for Workgroups system fonts, you can increase their size using the **FixedFon.Fon**, **Fonts.Fon**, and **OEMFonts.Fon** entries in the **[boot]** section of SYSTEM.INI. (However, you should only replace the default fonts with other system font files supplied with Windows for Workgroups, such as 8514FIX.FON, 8514SYS.FON and 8515OEM.FON. Using other fonts may cause some applications to display text incorrectly.)
- If you have trouble reading the titles of fonts, you can adjust the appearance of these titles using the **IconTitleFaceName** and **IconTitleSize** entries in the **[desktop]** section of WIN.INI.
- If you have trouble reading full-screen messages displayed by Windows, you can adjust the colors using the **MessageBackColor** and **MessageTextColor** entries in the **[386Enh]** section of SYSTEM.INI.
- If the desktop pattern or wallpaper causes eyestrain, you can adjust them using the Desktop section of Control Panel.
- If you have trouble differentiating the colors used by Window Help, you can adjust them using the **IFJumpColor**, **IFPopupColor**, **JumpColor**, **MacroColor** and **PopupColor** entries in the **[Windows Help]** section of WIN.INI.
- If you have trouble reading the text in Microsoft Mail, you can adjust the fonts using the **FixedFont**, **NormalFont** and **PrintFont** entries in the MSMAIL.INI file.
- If you have trouble reading the text in Schedule+, you can adjust the fonts and colors by selecting Display from the Options menu or by changing the values assigned to the **ApptBookColor**, **ApptBookLinesColor**, **LargeFont**, **OtherColor**, **PageBackgroundColor**, **PlannerColor**, **PlannerLinesColor** and **UserColor** entries in the SCHDPLUS.INI file.
- If you have trouble locating the text selection cursor (also known as the caret or insertion bar) you can adjust the rate at which it flashes using the Desktop section of Control Panel.

You can use the following methods to adjust the behavior of the mouse:

- If you have difficulty controlling the mouse you can adjust its speed using the Mouse section of Control Panel or the **MouseSpeed**, **MouseThreshold1** and **MouseThreshold2** entries in the **[Windows]** section of WIN.INI.
- If you have trouble seeing the mouse when it is being moved, you can try turning on the Mouse Trails option in the Mouse section of Control Panel or by changing the value of the **MouseTrails** entry the **[Windows]** section of WIN.INI.

- If you have trouble double clicking the mouse, you can adjust the sensitivity using the Mouse section of Control Panel or using the **DoubleClickHeight**, **DoubleClickWidth** and **DoubleClickSpeed** entries in the [Windows] section of WIN.INI.
- If you have trouble positioning the mouse on window frames in order to adjust the size of the windows, you can increase the window border width using the Desktop section of Control Panel.
- If you have trouble using the keyboard because of the speed at which it repeats or the delay time before it begins repeating, you can adjust these using the Keyboard section of Control Panel. (You can adjust them even further using Access Pack for Windows, described earlier in this appendix.)

For more features to help use Windows for Workgroups, see the section Access Pack for Windows earlier in this chapter.

Getting More Information

For more information on Microsoft products and services for people with disabilities, contact Microsoft Customer Support Services at 1-800-426-9400.

The Trace R&D Center at the University of Wisconsin-Madison produces a book and a CD that describe products that help people with disabilities use computers. The book is titled *ResourceBook*. It provides descriptions and photographs of about 2,000 aids in about 500 pages. The CD provides a database of more than 12,000 products, and other information for people with disabilities. It is issued twice a year, and also should be available in many public libraries by early 1993.

For general information and recommendations on how computers can help specific individuals, you should consult a trained evaluator who can best match the individual's needs with the available solutions. If you are in the United States, you can obtain information about resources in your area by calling the National Information System, an information and referral center for people with disabilities, at the following address:

National Information System (NIS)
Center for Developmental Disabilities
University of South Carolina, Benson Bldg.
Columbia, SC 29208

Phone: 1-800-922-9234 (voice/text telephone)
1-800-922-1107 (in South Carolina)

Fax: 1-800-777-6058

Integrating Additional Protocols with Windows for Workgroups

This appendix contains information that describes the procedures for installing and configuring the Microsoft TCP/IP, Microsoft ASYBEUI, Microsoft DLC, and AT&T StarLAN ISO protocols for use with Windows for Workgroups. A solid understanding of the network protocols that are described in this appendix is helpful, and will help to ensure a successful installation.

Note The information provided in this appendix is preliminary and was not completely tested by Microsoft at the time the *Windows for Workgroups Resource Kit* went to print (September 1992).

Windows for Workgroups fully supports connectivity to LAN Manager and Novell NetWare networks using NetBEUI and MSIPX protocols, respectively, as described in Chapter 8, "Network Integration with Microsoft LAN Manager and Novell NetWare." Other protocols and networks may work with Windows for Workgroups, however, limited support may be available for these from Microsoft Product Support Services (PSS).

Integrated setup procedures that are supported by PSS will be available at a later date—check the Microsoft Download Service (MSDL), CompuServe, and other electronic bulletin board systems for announcements of additional information. (See Appendix A for information on these resources.)

Related information

- *Windows for Workgroups Resource Kit*. Chapter 6, "Windows for Workgroups Initialization Files;" Chapter 7, "Additional Windows for Workgroups Information;" Chapter 14, "Troubleshooting Windows for Workgroups."
- *Microsoft LAN Manager Resource Kit*

Contents of This Appendix

About Additional Protocol Integration	E-2
NDIS Transport Protocol Integration	E-3
Configuring the Microsoft TCP/IP Protocol	E-3
Configuring Microsoft RAS 1.0	E-8
Configuring Microsoft DLC	E-14
Configuring the AT&T StarLAN ISO Stack	E-18
Integration with Banyan VINES	E-21
Integration with Other Network Operating Systems	E-21

About Additional Protocol Integration

This appendix provides information about using additional protocols with Windows for Workgroups. Instructions are provided to help the reader install and configure additional network protocols such as Microsoft TCP/IP, ASYBEUI, and DLC.

The instructions provided in this appendix are preliminary and have not been completely tested and are designed to serve as an aid until more information can be made available.

As these instructions are preliminary, support is not available from Microsoft Support Services to assist with problems that may arise from following these procedures.

Note Check the appropriate Readme file provided on the *Windows for Workgroups Resource Kit* disk for the latest information known about using additional protocols with Windows for Workgroups.

The following tips should be observed when working with these additional transport protocols:

- Make a backup copy of all files that you will be modifying before following the installation procedures described in this appendix. The files that you will need to backup include CONFIG.SYS, AUTOEXEC.BAT, PROTOCOL.INI, and SYSTEM.INI.
- Exclude the memory address range used by your network adapter card (and any other adapters installed in your computer) using the **x=** option on EMM386.EXE, or using the **emmxclude=** entry in the **[386Enh]** section of the SYSTEM.INI file. This will ensure that you don't have a conflict in the upper memory area.
- If you experience problems using other protocols when NetBEUI is configured on your workstation, try the **/n** switch on the WORKGRP.SYS driver (for example, **device=C:\WINDOWS\WORKGRP.SYS /n**) — this will prevent NetBEUI from loading and may allow you to successfully use the other protocol. More information about the use of the **/n** switch is provided in the “Disabling the NetBEUI Protocol” section of Chapter 7.
- If after following the procedures described in this appendix you receive errors when your computer starts up, or you experience hangs when Windows for Workgroups starts up, double-check to ensure that you have performed all of the documented installation procedures. (Although the provided installation procedures are relatively straightforward, installation of some of the protocols may require many steps, and thus is error-prone.)

NDIS Transport Protocol Integration

Microsoft LAN Manager 2.1 provides support for several protocols not provided as part of Windows for Workgroups. These additional protocols include TCP/IP, ASYBEUI (used for remote asynchronous connectivity; also known as Remote Access Services or RAS), and DLC.

In general, any NDIS-compliant protocol can work with Windows for Workgroups. The issue is whether the protocol has been integrated with a SETUP installation procedure, or whether the settings files (for example, CONFIG.SYS, AUTOEXEC.BAT, and PROTOCOL.INI) must be hand-edited to configure the protocol. The protocol installer for use with LAN Manager was not revised before Windows for Workgroups was made available, but the procedures below describe how the settings files can be modified to use other protocols.

The protocols provided by Microsoft for use with LAN Manager will be modified to install on a Windows for Workgroups workstation, but they were not available at the time the *Windows for Workgroups Resource Kit* manual was released to print (September 1992).

The following information was compiled during the beta test for Windows for Workgroups and will aid you in getting these additional transport protocols to work with Windows for Workgroups.

Configuring the Microsoft TCP/IP Protocol

Installation of the TCP/IP protocol stack requires the manipulation of the CONFIG.SYS, AUTOEXEC.BAT, PROTOCOL.INI, and SYSTEM.INI files on Windows for Workgroups computers.

The following steps have been prepared to make the task easier. A completed example of each of these files has been provided at the end of this section for reference.

Note The following steps discuss the installation of the basic TCP/IP protocol stack. These procedures do not discuss the installation of the TCP/IP Utility pack. Thus, this installation won't permit the use of the sockets library.

1. Install MS-DOS LAN Manager on a computer.

Before installing TCP/IP, you must first get a copy of the files. This is easily done by installing MS-DOS LAN Manager on a single computer,

and then copying the files on a floppy disk or a file share point for distribution to Windows for Workgroups computers.

Note Please follow the procedures and installation prompts for installing the TCP/IP protocol stack. When prompted for the IP address, subnet mask, and the default gateway address, you may want to use information for one of the workstations you will be installing. You will be adding this information manually to the PROTOCOL.INI file during the actual installation.

2. Copy the following files to a floppy disk:

```
C:\LANMAN.DOS\DRIVERS\PROTOCOL\TCPIP\EMSBFR.EXE
C:\LANMAN.DOS\DRIVERS\PROTOCOL\TCPIP\NEMM.DOS
C:\LANMAN.DOS\DRIVERS\PROTOCOL\TCPIP\TCPTS.R.EXE
C:\LANMAN.DOS\DRIVERS\PROTOCOL\TCPIP\TCPDRV.DOS
C:\LANMAN.DOS\DRIVERS\PROTOCOL\TCPIP\TINYRFC.EXE
C:\LANMAN.DOS\DRIVERS\PROTOCOL\TCPIP\UMB.COM
C:\LANMAN.DOS\DRIVERS\PROTOCOL\TCPIP\NMTSR.EXE
C:\LANMAN.DOS\TCPUTILS.INI
C:\LANMAN.DOS\NETPROG\PING.EXE
```

3. Install Windows for Workgroups on a computer.
4. Copy the TCP/IP files from the floppy disk in step 2 on the WINDOWS directory of the Windows for Workgroups computer. (If Windows for Workgroups was installed in a directory besides WINDOWS, copy these files to that directory.)
5. In the CONFIG.SYS file, insert the following lines immediately *after* the line that installs PROTMAN.DOS (before WORKGRP.SYS is referenced):

```
device=c:\windows\nemm.dos
device=c:\windows\tcpdrv.dos /i:c:\windows
```

Note The directories for the device drivers and the information pointer for the TCPUTILS.INI file (/i:c:\windows) may be different if Windows for Workgroups was installed in a directory other than that referenced in these procedures. Refer to the directory discussion in step 4.

6. Make the follow modifications to the AUTOEXEC.BAT file.
 - Immediately *before* the **net start** line, add the following line:
c:\windows\umb.com
 - Immediately *after* the **net start** line, add the following lines:
c:\windows\emsbfr
c:\windows\tcpsr

```
c:\windows\tinyrfc
c:\windows\nmtsr
```

7. Make the following modifications to the \WINDOWS\PROTOCOL.INI file:
 - After the last existing transport line in the **[network.setup]** section, add the following line:
transport=msStcp,msStcp
 - After the last existing LANA number line in the **[network.setup]** section, add the following line:
lanan1=msSelnkii,1,msStcp

Note The LANA number may be different if other protocols are being loaded before the TCP/IP stack. Also, the device driver identifier (shown as MS\$ELNKII in these steps) is dependent on the type of network interface card you are using.

- Add the following lines to the end of the PROTOCOL.INI file:

```
[msStcp]
drivername=tcip$
nbssessions=10
bindings=msSelnkii
lanabase=1
ipaddress0=11 1 7 99
subnetmask0=255 255 0 0
defaultgateway0=11 1 0 1
```

Notes

- The **lanabase=** value must match the **lanax=** entry in the **[network.setup]** section. The **bindings=** entry is the section name ([xxxxxx]) of the device driver to which the TCP/IP stack will be bound. This value is dependent on the type of network interface card used. The **ipaddress0**, **subnetmask0**, and **defaultgateway0** values are computer- and site-dependent. These values are provided as an example. Of special interest is that spaces are used as separators in the numbers instead of the more traditional periods.
 - Notice the **nbssessions** value was increased from its LAN Manager default value of 6, to increase the number of sessions supported by the TCP/IP device driver.
-
8. Make the following modifications to the \WINDOWS\SYSTEM.INI file:
 - After the **[386Enh]** section add the following line:
V86ModeLanas=1

Note The parameter for **V86ModeLanas=** is a list of LANA numbers that are supported by real mode protocol stacks. If TCP/IP has been given a different LANA number it should be used.

- Increase the **netheapsize=** parameter in the **[386Enh]** section from its default value of 20 to 60.

Note The additional memory is needed as buffer space for passing network frames between real and protected mode.

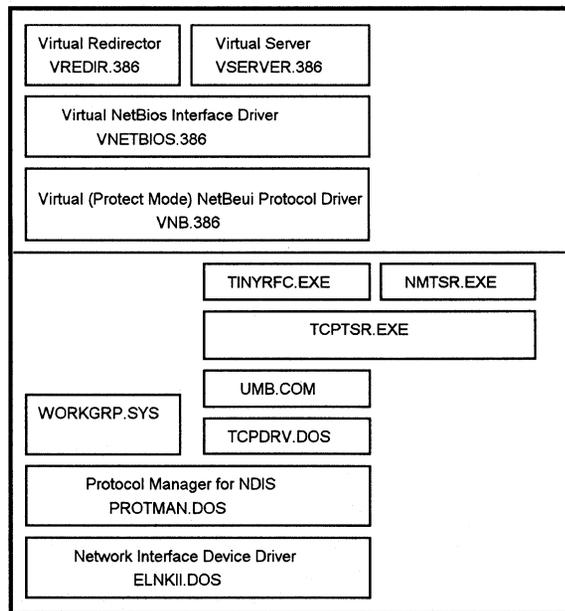
9. Restart the target computer.

Once all steps are completed and verified, restart the computer to load the needed device drivers, and TSRs.

Figure E.1 details the active components of the TCP/IP installation.

Figure E.1

Active components in a NetBEUI and TCP/IP installation



Notice that NMTSR.EXE is an optional component and is used only with the PING utility.

The following are the final installation images of CONFIG.SYS, AUTOEXEC.BAT, and PROTOCOL.INI. Areas where you need to make changes manually (per the preceding procedure) are highlighted in **bold**.

The following is an example of a CONFIG.SYS file image:

```
device=c:\dos\setver.exe
device=c:\windows\himem.sys
device=c:\windows\emm386.exe noems x=b800-c800
dos=high,umb
files=30
buffers=10
lastdrive=z
device=c:\windows\protman.dos /i:c:\windows
device=c:\windows\nemm.dos
device=c:\windows\tcpdrv.dos /i:c:\windows
device=c:\windows\workgrp.sys
device=c:\windows\elnkii.dos
stacks=9,256
```

The following is an example of a AUTOEXEC.BAT file image:

```
c:\windows\smartdrv.exe
c:\windows\umb.com
c:\windows\net start
c:\windows\lemsbfr
c:\windows\tcptsr
c:\windows\tinyrfc
c:\windows\nmtsr
@echo off
prompt $p$g
path c:\windows;c:\dos
set temp=c:\dos
```

The following is an example of a PROTOCOL.INI file image:

Note The **ms\$elnkii** keyname used in the following example specifies the use of the 3Com EtherLink II network adapter card driver. Use the name of the network adapter card driver that is present in your existing PROTOCOL.INI file

```
[network.setup]
version=0x3100
netcard=ms$elnkii,1,ms$elnkii
transport=ms$netbeui,ms$netbeui
transport=ms$tcp,ms$tcp
lana0=ms$elnkii,1,ms$netbeui
lana1=ms$elnkii,1,ms$tcp

[protman]
drivename=protman$
priority=ms$netbeui

[ms$elnkii]
drivename=elnkii$
interrupt=5
ioaddress=0x300
```

```
[ms$elnkii]
drivename=elnkii$
interrupt=5
ioaddress=0x300
maxtransmits=40
transceiver=external
```

```
[ms$netbeui]
drivename=netbeui$
sessions=6
ncbs=12
bindings=ms$elnkii
lanabase=0
```

```
[ms$tcp]
drivename=tcpi$
nbsessions=10
bindings=ms$elnkii
lanabase=1
ipaddress0=11 17 99
subnetmask0=255 255 00
defaultgateway0=11 1 0 1
```

Note Here is some additional information about TCP/IP installations:

- WORKGRP.SYS will give a diagnostic error at startup (No NETBEUI section found in configuration information) if you do not have NetBEUI as an additional protocol. This is expected and harmless.
 - TCP/IP may require time to identify itself on the LAN/WAN before starting Windows for Workgroups. Therefore, if you start Windows for Workgroups automatically from AUTOEXEC.BAT, it is recommended that you place a PAUSE statement before you invoke Windows for Workgroups and allow approximately 10 seconds for the TCP/IP protocol to initialize before proceeding.
-

Configuring Microsoft RAS 1.0

Installation of the Remote Access Service (RAS) version 1.0 protocol stack involves not only installing the protocol stack but adding the support programs as well. Installation of the RAS protocol stack requires you to make manual changes to the CONFIG.SYS, AUTOEXEC.BAT, PROTOCOL.INI, and SYSTEM.INI files.

The following steps were prepared to identify the steps that are necessary to install the RAS protocol. Completed examples of the files are provided for reference at the end of this section.

1. Install RAS on an MS-DOS LAN Manager computer.

Before installing RAS, you must first get a copy of the files. This is easily done by installing MS-DOS LAN Manager on a single computer, and then copying the files to a floppy disk or a file share point for distribution to Windows for Workgroups workstations.

Note Please follow the procedures and installation prompts for installing RAS. When prompted for the COM port, modem type, and phone number, you may wish to use information for the most common workstation environment. Further manipulation of MODEM.INF may be needed if this information does not match the workstation's hardware configuration.

2. Copy the following files to a floppy disk:

```
C:\LANMAN.DOS\DRIVERS\ASYNC\ASYMAC.DOS
C:\LANMAN.DOS\DRIVERS\PROTOCOL\ASYBEUI\ASYBEUI.EXE
C:\LANMAN.DOS\NETPROG\VCMMIOD.EXE
C:\LANMAN.DOS\NETPROG\WANTSR.EXE
C:\LANMAN.DOS\NETPROG\RASADMIN.EXE
C:\LANMAN.DOS\NETPROG\RASADMIN.HLP
C:\LANMAN.DOS\NETPROG\RASDIAL.EXE
C:\LANMAN.DOS\NETPROG\RASHELP.EXE
C:\LANMAN.DOS\NETPROG\RASPHONE.EXE
C:\LANMAN.DOS\NETPROG\RASPHONE.HLP
C:\LANMAN.DOS\NETPROG\RASPHONE.ICO
C:\LANMAN.DOS\COMDEV.INI
C:\LANMAN.DOS\MODEMS.INF
C:\LANMAN.DOS\PHONE.INF this file may not be present if this is a new installation
C:\LANMAN.DOS\RASDIAL.MSG
C:\LANMAN.DOS\RASHELP.MSG
C:\LANMAN.DOS\RASPHONE.MSG
```

3. Install Windows for Workgroups on a target computer.
4. Copy the RAS files from the floppy disk created in step 2 to the \WINDOWS directory (assuming this is the directory where Windows for Workgroups is installed) of the Windows for Workgroups computer.
5. Copy the following RAS files from the floppy disk created in step 2 to the root directory of the Windows for Workgroups computer (for example, C:\):

```
COMDEV.INI
MODEMS.INF
PHONE.INF
RASDIAL.MSG
RASHELP.MSG
RASPHONE.MSG
```

6. In the CONFIG.SYS file, insert the following line immediately *after* the line that installs PROTMAN.DOS:

device=c:\windows\asymac.dos

7. Make the following modifications to the AUTOEXEC.BAT file:

- Immediately *before* the **net start** line, add the following line:

c:\windows\asybeui.exe

- Immediately *after* the **net start** line, add the following lines:

c:\windows\vcommiod.exe

c:\windows\wantsr.exe

8. Make the following modifications to the \WINDOWS\PROTOCOL.INI file:

- After the last existing **netcard=** line, add the following line:

netcard=ms\$asymac,1,MSSASYMAC

- After the last existing **transport=** line, add the following line:

transport=ms\$asybeui,MSSASYBEUI

- After the last existing **lanax=** line, add the following line:

lanax=ms\$asymac,1,ms\$asybeui

Note The LANA number listed above may be different on your configuration if you are using more than one protocol.

- Add the following lines to the end of the PROTOCOL.INI file:

**[ms\$asymac]
drivename=asymac\$**

**[ms\$asybeui]
drivename=asybeui\$
lanabase=1
bindings="ms\$asymac"
sessions=12
ncbs=25**

9. In the \WINDOWS\SYSTEM.INI file, add the following lines to the **[386Enh]** section:

**V86ModeLANAs=1
MaintainServerList=No**

Notes

- The **V86ModeLANAs=** entry identifies a list of LANA numbers that are supported by real mode protocol stacks. If RAS has been given a different LANA number it should be used.
- The **MaintainServerList=No** entry prevents the remote workstation from being designated as the computer that maintains the list of workgroups or computers on the network—this is for performance reasons.

10. Reboot the target computers to load the device drivers and TSRs, and then start RASPHONE to ensure that everything was properly installed.
11. When you are sure everything is installed correctly, add RASPHONE to the Startup group to automatically start this utility each time you start Windows for Workgroups.

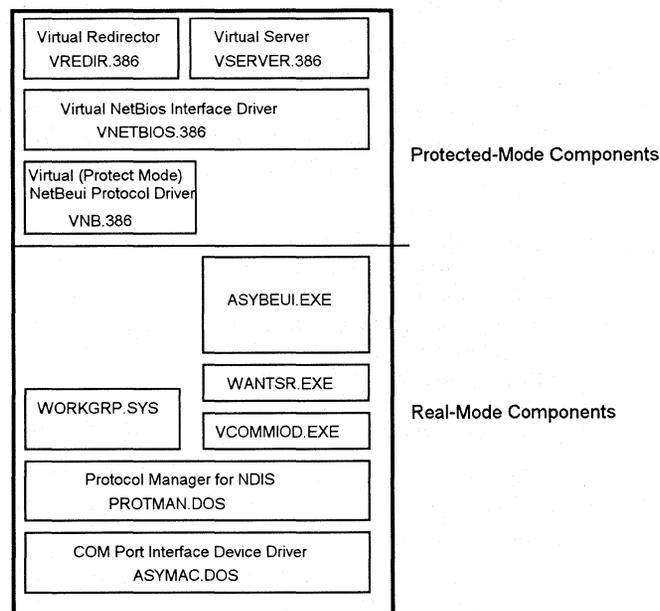
RASPHONE is used to dial in. It can't be started until after the redirector is started. In 386 enhanced mode, the redirector is VREDIR.386, and it isn't loaded until Windows is started.

When setting up RASPHONE, make sure the Working Directory is defined as "WINDOWS". RASPHONE uses this information to locate its message files.

Figure E.2 details the active components of the RAS installation.

Figure E.2

Active components in a NetBEUI and RAS installation



The following are the final installation images of CONFIG.SYS, AUTOEXEC.BAT, and PROTOCOL.INI. Areas in the file where you need to make changes manually (per the preceding procedure) are highlighted in **bold**.

The following is an example of a \CONFIG.SYS file image:

```
device=c:\dos\setver.exe
device=c:\windows\himem.sys
device=c:\windows\emm386.exe noems x=b800-c800
dos=high,umb
files=30
buffers=10
lastdrive=z

device=c:\windows\protman.dos /i:c:\windows
device=c:\windows\asymac.dos
device=c:\windows\workgrp.sys
device=c:\windows\elnkii.dos
stacks=9,256
```

The following is an example of a \AUTOEXEC.BAT file image:

```
c:\windows\smartdrv.exe
c:\windows\asybeui
c:\windows\net start
c:\windows\vcommiod
c:\windows\wantsr
@echo off
prompt $p$g
path c:\windows;c:\dos
set temp=c:\dos
```

The following is an example of a \WINDOWS\PROTOCOL.INI file image:

Note The **ms\$elnkii** keyname used in the following example specifies the use of the 3Com EtherLink II network adapter card driver. Use the name of the network adapter card driver that is present in your existing PROTOCOL.INI file

```
[network.setup]
version=0x3100
netcard=ms$elnkii,1,ms$elnkii
netcard=ms$asymac,1,ms$asymac
transport=ms$netbeui,ms$netbeui
transport=ms$asybeui,ms$asybeui
lana0=ms$elnkii,1,ms$netbeui
lana1=ms$asymac,1,ms$asybeui

[protman]
drivename=protman$
priority=ms$netbeui
```

```
[ms$elnkii]
drivename=elnkii$
interrupt=5
ioaddress=0x300
maxtransmits=40
transceiver=external
```

```
[ms$netbeui]
drivename=netbeui$
sessions=10
ncbs=32
bindings=ms$elnkii
lanabase=0
```

```
[ms$asymac]
drivename=asymac$
```

```
[ms$asybeui]
drivename=asybeui$
lanabase=1
bindings="ms$asymac"
sessions=12
ncbs=25
```

More notes

One final note about RAS installations:

- When Windows starts up, do not log on—you will log on once you have made a connection to the RAS server.

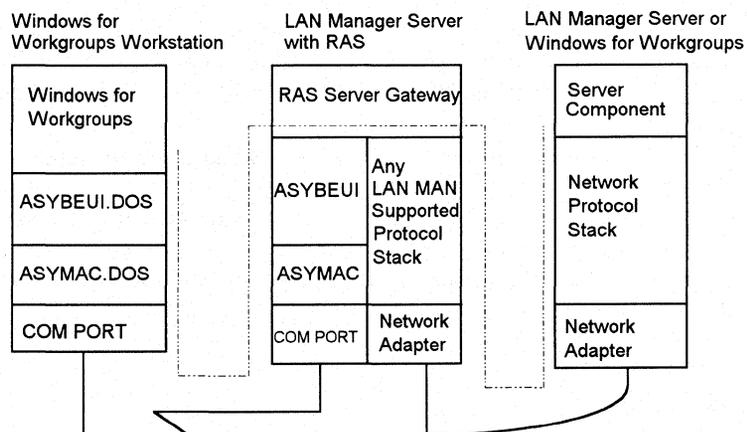
RAS Tuning Considerations

There are no special tuning considerations on the server when using Windows for Workgroups workstations rather than LAN Manager workstations. The server should be tuned according to the number of workstations and type of activities being performed.

Figure E.3 illustrates in general the workstation and server components in a RAS connection.

Figure E.3

General diagram illustrating RAS data flow



One of the components of the workstation that is communicated over the telephone line is its *name space*, the list of names by which the workstation is known. A workstation is known by many names on the network, including its computer name, its user name, its workgroup name, and so on.

A Windows for Workgroups workstation has a total of seven different names. All of these names are added to the network protocol stack when the user establishes a connection through RAS.

The server also adds its names to the network protocol stack. From five to nine names are used by the server. Each RAS server can support up to 16 concurrent remote workstations. Thus, 16 times seven workstation names, plus nine server names, creates a need for a maximum of 127 names. By default, the network protocol stack allows only 17 names to be added to its name space, so you can see that this limit can be quickly exceeded.

The number of names in a NetBEUI protocol stack is determined by the **names=** parameter in the **[netbeui]** section of the PROTOCOL.INI file in the \LANMAN directory on the server. In the TCP/IP protocol stack the parameter is **numnames=** in the **[tcpip]** section. Each name uses additional memory, so be sure to accurately estimate the number of names that will be needed before modifying PROTOCOL.INI.

Note In TCP/IP the maximum for the **numnames=** parameter is 127. In NetBEUI it is 256.

Configuring Microsoft DLC

Installation of the Microsoft DLC (MS-DLC) protocol stack is important for organizations that want to communicate to their mainframes (either S/370 or AS/400s) from Windows for Workgroups workstations. The MS-DLC protocol stack is also used for connections to a variety of network-attached printers. Installation of the MS-DLC protocol stack requires you to make some manual changes to the AUTOEXEC.BAT, PROTOCOL.INI, and SYSTEM.INI files. The following steps have been prepared to make the task easier. Completed examples of the files are provided for reference at the end of this section.

1. Install MS-DOS LAN Manager on a computer.

Before installing DLC, you must first get a copy of the files. This is done by installing MS-DOS LAN Manager on a single computer, and then copying the files to a floppy disk or a file share point for distribution to Windows for Workgroups computers.

Note Please follow the procedures and installation prompts for installing the DLC stack. MS-DOS LAN Manager requires loading one of the Session layer protocol stacks (NetBEUI or TCP/IP in addition to MS-DLC). Select the one you wish.

2. Copy the following file to a floppy disk:

C:\LANMAN.DOS\DRIVERS\PROTOCOLMSDLC\MSDLC.EXE

Note There are additional files in this subdirectory that outline different PROTOCOL.INI parameter settings for different connection options to different mainframe platforms. A quick review of these files may be helpful in configuring and tuning the MS-DLC protocol stack.

3. Install Windows for Workgroups on a target computer.
4. Copy the DLC file from step 2 to the \WINDOWS directory (or the directory where Windows for Workgroups is installed) on the Windows for Workgroups target computer.

Note No modifications to the CONFIG.SYS file are needed.

5. In the AUTOEXEC.BAT file, add the following line immediately *before* the **net start** line:

c:\windows\msdlc.exe

6. Make the following modifications to the \WINDOWS\PROTOCOL.INI file:

Note DLC doesn't use the LANA number to route command blocks. No **lanax=** number line or **transport=** line should be included for the DLC interface.

- Move the current protocol stack using LANA 0 to the last unused LANA number. (Refer to the final PROTOCOL.INI file image later in this section.)

- Add the following lines to the end of the PROTOCOL.INI file:

```
[dlcxif]
drivename=msdlc$
bindings=ms$elnkii
commands=24
saps=8
stations=20
timers=12
bufqelements = 512
t1_tick_one = 10
t1_tick_two = 50
ti_tick_one = 255
ti_tick_two = 255
```

Note The DLC protocol stack is treated as any transport (that is, NetBIOS stack) by the VNETBIOS.386 virtual device. This is possible because the real-mode DLC uses INT 5C, the same as standard NetBIOS protocol stacks. Commands are routed to the DLC protocol stack based on the first byte of the command block issued to the NetBIOS interface.

7. Make the following modifications to \WINDOWS\SYSTEM.INI:

- Add the following line to the [386Enh] section:

```
V86ModeLanas=0
```

- Add the following line to the [network] section:

```
exclude=0
```

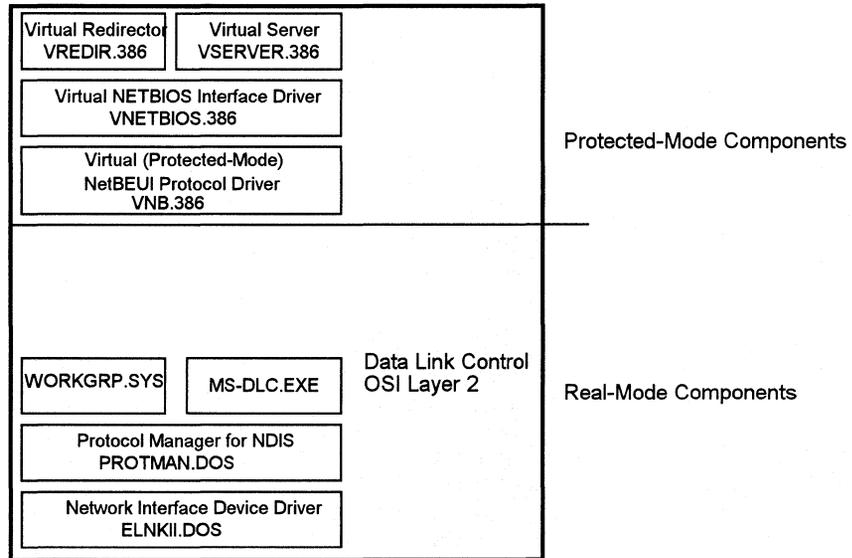
Note The V86ModeLanas= entry identifies LANA 0 as a real-mode protocol stack as used by DLC. The exclude= entry is used to inform the redirector and server not to use this LANA number. DLC doesn't support NCBs but does use INT 5C. The lines in step 7 are used to prevent Windows from Workgroups from trying to use the DLC as a NetBIOS stack and generating errors.

8. Once the parameters have been entered and verified, reboot the computer to load the TSR.

Figure E.4 details the active components of the DLC installation.

Figure E.4

Active components in a
NetBEUI and DLC
installation



The following are the final installation images of AUTOEXEC.BAT and PROTOCOL.INI. Areas where you need to make changes manually (per the previous procedures) are highlighted in **bold**.

The following is an example of a \AUTOEXEC.BAT file image:

```
C:\WINDOWS\SMARTDRV.EXE
c:\windows\msdlc.exe
C:\WINDOWS\net start
@ECHO OFF
PROMPT $p$g
PATH C:\WINDOWS;C:\DOS
SET TEMP=C:\DOS
```

The following is an example of a \WINDOWS\PROTOCOL.INI file image:

```
[network.setup]
version=0x3100
netcard=ms$elnkii,1,MS$ELNKII
transport=ms$netbeui,MS$NETBEUI
lana1=ms$elnkii,1,ms$netbeui

[protman]
DriverName=PROTMAN$
PRIORITY=MS$NETBEUI
```

```
[MS$ELNKII]
DriverName=ELNKII$
INTERRUPT=5
IOADDRESS=0x300
MAXTRANSMITS=40
TRANSCEIVER=EXTERNAL
```

```
[MS$NETBEUI]
DriverName=netbeui$
SESSIONS=10
NCBS=32
BINDINGS=MS$ELNKII
LANABASE=1
```

```
[DLCXIF]
DRIVERNAME=msdlc$
BINDINGS=MS$ELNKII
COMMANDS=24
SAPS=8
STATIONS=20
TIMERS=12
BUFQELEMENTS = 512
T1_TICK_ONE = 10
T1_TICK_TWO = 50
T1_TICK_THREE = 255
T1_TICK_FOUR = 255
```

Configuring the AT&T StarLAN ISO Stack

The following steps for installing the AT&T StarLAN ISO stack under Windows for Workgroups were generated from a StarLAN installation using a PC NAU 8-bit network adapter card. Other StarLAN network adapter cards may contain slightly different settings from those used in these instructions.

The information provided in this section assumes that the specified protocol is already working properly on an MS-DOS client workstation.

Required Components

In order to support the StarLAN ISO stack on a Windows for Workgroups workstation, you will need the following components:

A working installation of AT&T StarLAN, including the following:

- The \LANMAN.DOS\DRIVERS\STARLAN\ATTCSMA and \LANMAN.DOS\DRIVERS\PROTOCOL\OSI directories containing ATTCSMA.DOS and ATTISO.DOS.

- A valid PROTOCOL.INI file from your \LANMAN.DOS directory
- Copies of AUTOEXEC.BAT and CONFIG.SYS from your StarLAN installation.

Installation Procedure

Important Make a backup copy of your CONFIG.SYS, AUTOEXEC.BAT, and Windows for Workgroups PROTOCOL.INI file *before* performing the following installation steps.

Pre-Installation steps:
CONFIG.SYS and
AUTOEXEC.BAT files

1. Use a text editor to open the CONFIG.SYS file. Use "REM" at the beginning of a line to disable any lines relating to network drivers that load during the boot process. For example, the following line is commented out:

`REM DEVICE=C:\DEVICE\XXX.DOS`
2. Use a text editor to open AUTOEXEC.BAT. Use "REM" at the beginning of a line to comment out any lines relating to StarLAN.
3. Remove your StarLAN directory from your path. This is usually implemented in the StarLAN section with a statement like:

`SET PATH=C:\LANMAN.DOS\NETPROG;%PATH%`
4. Save these files, then reboot your computer.

*Installing Windows for
Workgroups*

5. Insert Disk 1 in your computer and run Windows for Workgroups Setup.

Follow the usual installation steps, all the way through to the selection of a network adapter card.
6. When prompted for your network adapter, choose the HP StarLAN network adapter.
7. Select LAN Manager in the Compatible Networks dialog box, and then complete the installation..

*Altering your
CONFIG.SYS file*

8. Use a text editor to open the CONFIG.SYS file that you use when starting AT&T StarLAN.

Copy the following two device driver lines for inclusion in your Windows for Workgroups CONFIG.SYS file:

```
DEVICE=C:\LANMAN.DOS\DRIVERS\STARLAN\ATTCSMA\ATTCSMA.DOS
DEVICE=C:\LANMAN.DOS\DRIVERS\PROTOCOL\OSI\ATTISO.DOS
```

This step assumes you have installed StarLAN in C:\LANMAN.DOS. If this differs from your configuration, simply replace C:\LANMAN.DOS with the correct drive and path to your LAN Manager installation.

9. Use a text editor to open the CONFIG.SYS file used when starting Windows for Workgroups.

Insert the two lines from step 8, placing them just before the device line containing WORKGRP.SYS in the order listed above.

10. Save the Windows for Workgroups CONFIG.SYS file.

*Altering your
PROTOCOL.INI file*

11. Use a text editor to open the PROTOCOL.INI file that you use with your StarLAN installation.

Find the section that contains the line that reads something like this:

```
DRIVERNAME = ATTISO$
```

Copy this whole section for inclusion in your Windows for Workgroups PROTOCOL.INI file. Be sure to include the section tag with the square brackets [] around it.

12. Use a text editor to open the PROTOCOL.INI file in your Windows for Workgroups directory.

Paste the sections copied from step 11.

13. In your StarLAN PROTOCOL.INI file, find the section that contains the line that reads something like this:

```
DRIVERNAME = ATTCSMA$
```

Copy this whole section for inclusion in your Windows for Workgroups PROTOCOL.INI file. Be sure to include the section tag with the square brackets [] around it.

14. Paste the sections copied from step 13 in your Windows for Workgroups PROTOCOL.INI file.

15. In the Windows for Workgroups PROTOCOL.INI file, find the section that has the following line (assuming you selected the HP StarLAN net card in step 6):

```
DRIVERNAME = HPLAN$
```

Remove this entire section along with the line within square brackets.

Note If you don't need NetBEUI as an additional protocol, proceed to step 18.

Enabling NetBEUI

16. Find the **LANABASE=** line in any of the transport protocol sections with the highest number following it and make note of it for step 17.

17. In the PROTOCOL.INI file that you have in your Windows for Workgroups directory, find the section [**MSSNETBEUI**].

Change the **LANABASE=** in any of the transport protocol sections with the higher than the number you found in step 16. This will most likely presently be set to a 1 (that is, you will probably change this to a 2).

(If you just followed steps 15 and 16, you are done.)

18. If you do *not* want NetBEUI as a protocol, use the **/n** option with the WORKGRP.SYS driver—the **/n** option tells the WORKGRP.SYS driver to not load the NetBEUI protocol.

Integration with Banyan VINES

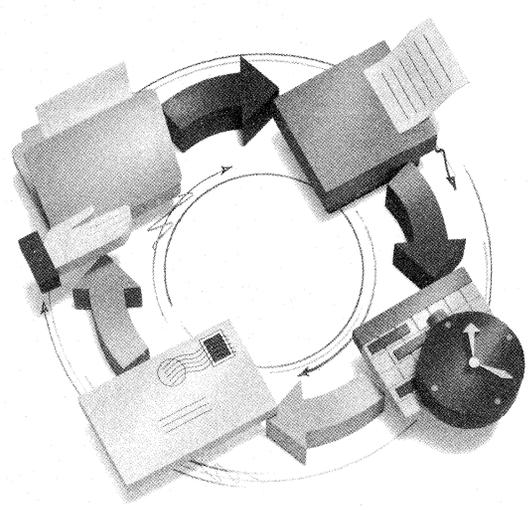
Microsoft is working with Banyan Systems to provide secondary network integration with Windows for Workgroups. Information about this support was not available at the time this manual went to print. Contact Banyan Systems, Microsoft PSS, or check the appropriate forums in CompuServe for more information and availability of integration components to support Banyan VINES with Windows for Workgroups.

Integration with Other Network Operating Systems

The support for integrating Windows for Workgroups with network operating systems other than LAN Manager (and compatible) and Novell NetWare as discussed in the preceding chapters is primarily dependent on whether the NOS vendor supports NDIS 2.x compliant network adapter drivers.

For example, Windows for Workgroups will not interoperate with Artisoft LANtastic, or Novell NetWare Lite. Both of these network operating environments do not support the use of NDIS-compliant network adapter drivers, and therefore are incompatible with Windows for Workgroups.

Index



Index

Accessories
ClipBook Viewer 10-6-9, 11-13, 11-16-18
deleting 4-22
filenames 4-17-19
Net Watcher 10-9-11
WinMeter 10-12

Adapter card. *See* Network adapter card

Add/Remove Files dialog box 3-14

Address Book] section in MSMAIL.INI 6-82

Advanced Power Management (APM) 10-13

Application. *See also* MS-DOS-based applications; Windows applications

Application Programming Interface (API) 2-2-4

common dialog boxes 10-2-4

errors 14-45

exclusive mode 6-31

filename extension 6-26

font files 4-9

function libraries 2-4-5

installing with Setup 3-16, 5-3, 5-42-46, 5-70-72

layer
IEEE networking model 1-9
OSI networking model 1-7

pathname 6-19

private initialization files 6-7

processing time statistics 10-12-13

Program Manager group 5-72

scheduling (appointment) applications 13-9

standard mode 2-5-6, 2-12-13

starting
automatically 3-2, 3-31, 6-22
disabling automatic 7-17
from Mail 12-15
with document file 6-10-11

swap file 9-5

Application, *continued*

temporary (.TMP) files 9-3

386 enhanced mode 2-6-11, 2-15-19

UNC directory searches 10-4-5

working directory 5-73

Application Programming Interface (API)
described 2-2-4
messaging interface (MAPI) 6-15, 12-13, 12-26-27
Microsoft Mail File Format API (FFAPI) 13-9-10
Network DDE 11-24
Network DDE library 11-12
parameter validation 14-43

APPS.INF
described 5-3
editing 5-5
information file format 5-4-5
sections
[dontfind] 5-42
[pif] 5-43-47, 5-73-74
list of 5-42

ArcNet network
described 1-23
Novell NetWare network 8-17-18
specifications 1-24

Audio troubleshooting 14-45-48

AUTOEXEC.BAT
editing with other system files 6-4-5
minimized configuration 14-5-7
modified at system installation 3-16, 3-20
mouse driver entries 9-4
net start command 2-13
pause statements 14-39
rebooting after changing 14-5
ROUTE.COM (Token Ring card) 8-18
SET TEMP statement 14-34
SHARE.EXE 2-19

B

- Banyan VINES E-21
- Base memory address 1-29
- Baseband transmission 1-17-18
- Blind and low vision users D-4
- Boot sequence
 - multi-boot programs 7-4
 - standard mode 2-14-15
 - 386 enhanced mode 2-20
- [boot.description] section in SYSTEM.INI 6-59
- [boot] section in SYSTEM.INI
 - disabling file-handle caching 7-20
 - entries 6-57-59
 - modified at system installation 3-11-12
 - secondary network added to 5-60-61
 - 386 grabber file version 14-17-18
 - Windows boot shell 3-13
- Broadband transmission 1-17-18
- Browsing services 2-17, 8-7-8
- Bus topology 1-10-11

C

- Cable. *See* also specific type
 - ArcNet network 1-23
 - connectors 1-25-26
 - Ethernet networks 1-20
 - specifications 1-24
 - token-ring network 1-22-23
 - transmission techniques 1-17-18
 - troubleshooting network problems 14-38, 14-40
 - types of 1-13-17
- Calendering software. *See* Schedule+
- Cannot read/write to drive error 14-29
- Cannot-find-file error message 14-21
- CD-ROM player 7-13-14, 14-46-47
- Central file server network 1-37-38
- chkdsk command (MS-DOS) 9-4
- Classes
 - Microsoft University A-12-18
 - Windows 3.1 A-11
 - Windows for Workgroups A-13
- Clean boot configuration 14-3
- Clipboard
 - ClipBook Viewer 10-6

Clipboard, *continued*

- DDE data exchange 11-4
- Link clipboard 11-8
- metafile support 10-18
- Network DDE conversations 11-13
- ClipBook Server
 - described 10-7
 - Network DDE conversations 11-16-18
 - Network DDE example 11-18-22
- ClipBook Viewer
 - maximum pages 10-6
 - Network DDE
 - conversation interface 11-13
 - DDE share 11-16
 - example 11-18-22
 - system component 11-12
 - pastings Clipboard data 10-6
 - sharing pages 10-6-9
- [ClipShares] section in SYSTEM.INI 6-60
- Coaxial cable
 - connectors 1-25-26
 - Thicknet network 1-20, 1-21-22
 - Thinnet network 1-20-21
 - types of 1-13-15
- [colors] section in WIN.INI 6-7
- Common Dialogs DLL
 - described 10-2
 - NetWare button 10-3-4
 - Network button 10-3
 - UNC directory searches 10-4-5
- Compatible Networks dialog box 8-4-5, 8-13
- CONFIG.SYS
 - buffer statement 9-3
 - commands to load drivers in UMBs 9-8-11
 - editing with other system files 6-4-5
 - environment space setting 9-4
 - expanded memory driver entry 9-2-3
 - files statements 9-3
 - lastdrive statement 5-59
 - memory settings 14-20
 - minimized configuration 14-5-7
 - modified at system installation 3-16, 3-20
 - mouse driver entries 9-4
 - net start command 2-13
 - network driver entries 2-12-13
 - Novell NetWare drive assignments 8-18, 8-20
 - rebooting after changing 14-5

- Connect Network Drive dialog box 10-3
 - Control Panel
 - 386 Enhanced icon 3-31
 - foreground priority 14-30
 - customizing Windows for Workgroups 3-5
 - deleting system files with 4-22
 - information file. See CONTROL.INF 5-3
 - network adapter card drivers 5-53
 - Network icon
 - Change Password button 8-12
 - computer name 1-35
 - installation defaults 5-51
 - maintaining settings 3-31
 - network adapter card 1-26, 6-73
 - Network Settings dialog box 7-14-15
 - workstation optimization 9-16-17
 - permanent swap file size 4-22
 - current printer driver 14-34
 - protocol options 5-55-56
 - secondary networks 5-57
 - CONTROL.INF
 - described 5-48
 - editing 5-5
 - information file format 5-4-5
 - removing drivers from 5-69-70
 - sections 5-48-49
 - CONTROL.INI
 - editing 6-3-4
 - sections 6-79
 - source file 6-5
 - copy command (MS-DOS) 3-3, 3-20
 - Copying files to a network share 7-17-19
 - Currency defaults 6-11-14
 - [Custom Commands] section
 - MSMAIL.INI 6-82-83, 12-15-16
 - SHARED.INI 6-94, 12-16-19
 - Custom message types
 - customizing Mail with 12-13
 - described 12-19-21
 - help request sample 12-23-25
 - installing in Mail 12-21-23
 - maximum number installed 12-20
 - Schedule+ 12-21, 13-7-9
 - Schedule+ initialization file 13-3
 - [Custom Messages] section
 - MSMAIL.INI
 - initialization file entries 6-83-84
 - installing custom messages 12-21
 - [Custom Messages] section, *continued*
 - Schedule+ predefined message types 13-7
 - SHARED.INI
 - initialization file description 6-94
 - installing custom messages 12-22
 - installing help request forms 12-25
 - Custom Separator File 10-18-20
- ## D
- Data frame 1-5-6, 1-7
 - Date defaults 6-11-14
 - [DDEShares] section in SYSTEM.INI 6-60, 11-14
 - DDE. *See* Dynamic data exchange; Network DDE
 - Deaf and Hard of Hearing Users D-2
 - DEC Pathworks server 8-3
 - Deleting files
 - backup and temp files 9-4
 - files you can delete 4-22
 - freeing disk space 9-4-5
 - initialization files 7-4
 - Desktop configuration 14-11
 - [desktop] section in WIN.INI 6-8-9
 - Device drivers
 - CONFIG.SYS entries 2-12-13
 - driver files
 - communications devices 4-5
 - display drivers 4-5
 - keyboard 4-4
 - laptop/notebook PCs 4-5
 - mouse 4-5
 - MS-DOS drivers 4-13-14
 - multimedia 4-9
 - network 4-8
 - Novell NetWare 8-15
 - printers 4-6-8
 - system 4-4
 - types of 4-3
 - grabber 6-57
 - in upper memory 9-4
 - installing
 - automatically 3-2
 - custom choices in Setup 3-5
 - incompatible drivers 3-2, 3-16
 - OEM network drivers 5-65-68

- Device drivers, *continued*
 - removing from Setup 5-69-70
 - Setup switch to select drivers 3-4
 - System Information screen 3-10-11
 - system settings 3-23-30
 - SYSTEM.INI modifications 3-11-12
 - troubleshooting 14-9-10
 - updating driver files 3-2, 3-15
- loading in upper memory 9-8-11
- multimedia 4-9, 5-34-35, 6-16, 6-62
- NDIS interface 1-27-28
- network drivers 2-11-13
- printer drivers in Control Panel 5-48-49
- Protocol Manager 6-73
- SETUP.INF entries
 - display drivers 5-12-15
 - driver updates 5-35-36
 - incompatible drivers 5-33-34
 - keyboard 5-15-17
 - mouse 5-18-19
 - multimedia drivers 5-34-35
 - network drivers 5-20-23
 - system files 5-38-39
- system driver switches 9-2
- SYSTEM.INI entries
 - driver alias names 6-60
 - keyboard 6-61-62
 - MCI drivers 6-62
 - startup driver configuration 6-57-59
- virtual devices 2-8-11, 2-16-19
- Windows interface 2-2-4
- devicehigh** command (MS-DOS) 9-9, 9-15, 14-21
- Dialog box
 - Add/Remove Files 3-14
 - Common Dialogs DLL 10-2-5
 - Compatible Networks 8-4-5, 8-13
 - Connect Network Drive 10-3
 - File Open 7-17, 10-2, 10-4-5
 - File Save As 7-17-19, 10-2
 - NetWare button 10-3-4
 - Network button 10-3
 - Network Settings 7-14-15, 9-17-18
 - Paste 10-6
 - Separator Pages 10-15-17
 - Share Clipbook Page 10-7
 - Windows Setup 3-13-14
- Disk-compression software 9-16
- Disk space. *See* Hard disk
- Diskless workstation 9-3
- Display driver
 - files 4-5
 - grabber files 4-14-15
 - incompatible adapter chip 14-11
 - troubleshooting display problems 14-15-20
 - troubleshooting fonts 14-32-33
- Display fonts
 - loaded at startup 6-14
 - MS-DOS-based application 6-32
- DLL (dynamic link library)
 - Common Dialogs DLL 10-2-5
 - described 2-4-5
 - language libraries 4-12
 - Mail custom commands 12-14-15
 - Windows 3.1 extensions 2-2-4, 2-3
- DMA (direct memory access) 6-36, 6-39
- Domain, LAN Manager 8-7-8
- DOS. *See* MS-DOS
- Dr. Watson 14-45
- [drivers] section in SYSTEM.INI 6-60
- Dual-network environment 1-33-34, 8-3
- Dynamic binding 2-5
- Dynamic data exchange (DDE)
 - application 11-7
 - client application 11-6
 - Clipboard transfers 11-4
 - conversation 11-3, 11-6-8
 - data hierarchy 11-6-7
 - data item 11-7
 - data links 11-8
 - DDE Management Library 11-3-4
 - described 11-3
 - example uses 11-6-7
 - global atom/shared-memory handle 11-4
 - messages 11-3-4, 11-8-10
 - networks. *See* Network DDE
 - OLE comparison 11-10-11
 - server application 11-6-7
 - server name service 11-3
 - topic 11-7
 - transferring Mail messages 12-15-16
 - Windows messages 11-4
- Dynamic link library. *See* DLL

E

- Edit MSMAIL.INI command 12-15
- [embedding] section in WIN.INI 6-10
- EMM386.EXE 9-8, 9-11
- Empty Wastebasket Mail command 12-18-19
- Environment space 9-3
- Ethernet
 - cabling connectors 1-25-26
 - described 1-19-20
 - specifications 1-24
 - Thicknet 1-21
 - Thinnet 1-20
 - troubleshooting 14-40
 - twisted-pair 1-22
- Exclusive mode 7-20-21
- Expanded memory
 - 386 enhanced mode settings 6-30-56
 - allocating UMBs 9-8-16
 - expansion board configuration 9-2-3
 - optimizing EMM386.EXE 9-11-13
 - troubleshooting
 - EMM386 adapter conflict 14-23
 - invalid path 14-23
 - out-of-memory message 14-24
- Extended memory
 - 386 enhanced mode 2-7
 - adding 14-26
 - expansion board configuration 9-2-3
 - HIMEM.SYS driver 2-5-6
 - memory manager for setup 3-13
 - RAM drive size 14-25, 14-27
 - RAMDrive 9-3
 - SMARTDrive 3-16
 - standard mode 2-5-6, 4-15
- Extensions for Windows for Workgroups
 - Mail extensions 12-29-35
 - Schedule+ extensions 13-17
- [extensions] section in WIN.INI 6-10-11

F

- FFAPI (Microsoft Mail File Format API) 13-9-10
- File Manager
 - Connect dialog 7-8
 - described 10-14
 - disabling file sharing interface 10-15

File Manager, *continued*

- file sharing interface 7-4-5
 - Mail user interface 12-27
 - MRU network drive connections 10-15-17
 - network connections 2-17, 8-22
 - toolbar 10-14
 - UNC redirection 7-17-19
 - WINFILE.INI entries 6-80-81
- File Open dialog box
 - accessing shared resources 10-2-4
 - UNC directory searches 10-4-5
 - UNC redirection 7-17-19
 - File Save As dialog box 7-17-19, 10-2, 10-4-5
 - File-handle caching 7-20
 - Fixed-pitch screen fonts 6-32, 6-37
- Font
 - desktop icons 6-8
 - filename extensions 4-9
 - grabber 4-15
 - installation defaults 5-23, 5-32-33
 - loaded at startup 6-14
 - MS-DOS-based application
 - font files 4-11-12
 - mouse support 6-64
 - screen font 6-32, 6-37
 - OEM character set 4-9-10, 6-59
 - PostScript font files 4-7-8
 - raster font files 4-10-11
 - soft font installer 4-7
 - substitutions 6-15
 - system
 - font files 4-10
 - specifying 6-59
 - troubleshooting 14-32-33, C-20-23
 - TrueType
 - font files 4-11
 - WIN.INI entries 6-21
 - vector font files 4-11
- [fonts] section in WIN.INI 6-14
 - [FontSubstitutes] section in WIN.INI 6-15
 - Frame postamble 1-7
 - Frame preamble 1-7

G

- Games
 - deleting 4-22
 - filenames 4-17-19

Games, *continued*
 Hearts 6-11, 10-14
 Gateways, Mail 12-30-31, B-17, B-24-48
 GDI (Graphics Device Interface)
 described 2-3
 metafile 10-18-20
 standard mode 2-5-6
 386 enhanced mode 2-7
 Windows 3.1 core file 4-3
 General Protection (GP) faults 14-42-45
 Global atom 11-4
 Global shared-memory handle 11-4
 Grabber
 386 grabber fonts 5-15
 application data exchange 4-14-15
 incompatible driver 14-16
 installed 3-12
 mouse support 6-64
 support files 4-15
 SYSTEM.INI entries 6-57
 video display problems 14-15
 Graphics Device Interface. *See* GDI

H

Hard disk
 disk space requirements
 LAN Manager server 8-7-8
 Mail postoffice 12-38
 server 3-17
 Windows for Workgroups 3-17
 Windows for Workgroups
 components 3-14
 workstation 3-17
 fragmented 9-4
 freeing disk space
 archiving server calendar data 13-6
 deleting application files 9-4
 deleting swap files 4-22, 9-4-5
 deleting system files 4-22
 disk compression software 9-16
 RAMDrive 9-2, 9-8
 swap file 9-8
 interleave 9-2
 problems in 386 enhanced mode 14-29
 32-bit disk access 6-30
 Hearts game 6-11, 10-14
 [Hearts] section in WIN.INI 6-11

Help Request Mail command 12-23-25
 Help window WIN.INI entries 6-27-28
 High memory area (HMA), MS-DOS in 9-8-9
 HIMEM.SYS driver 2-5-6, 2-7, 9-3, 9-14

I

I/O system 1-28, 2-3
 IBM LAN Server workstation 8-3
 IEEE 802 networking model 1-8
 Incorrect system version error 14-16
 Installation information (.INF) files. *See also*
 individual filename
 APPS.INF 5-42-47
 CONTROL.INF 5-48-49
 custom installations 5-69-74
 editing 5-5, 6-3-4
 file descriptions 5-3-4
 format 5-4-5, 6-2-3
 list of 6-2
 NETWORK.INF 5-50-63
 OEMSETUP.INF 5-64-68
 SETUP.INF
 sections 5-8-41
 specifying pathname 3-4
 source files 6-5
 Installing Windows for Workgroups
 custom installation
 Add/Remove Files dialog box 3-14
 adding applications 5-70-72
 Compatible Networks dialog box 3-15
 custom setup 3-6-7
 customization options 3-5-6
 installation defaults 5-3-4, 5-69
 Network Adapters dialog box 3-15
 Printers dialog box 3-15
 removing drivers 5-69-70
 required shared copy installation 5-70
 Set Up Applications dialog box 3-16
 settings file 3-23-30
 Setup switches 3-5
 specifying WINDOWS
 directory 3-9-10
 System Information screen 3-10-11
 Virtual Memory dialog box 3-14
 Windows components 3-12-13
 express setup
 starting 3-6-7

Installing Windows for Workgroups, *continued*

- WINDOWS directory 3-9
 - memory required 3-9, 3-13
 - MS-DOS mode setup 3-7, 3-9-13
 - network installation
 - automated setup 3-21-23
 - MS-DOS workstation 3-20
 - options 3-17-18
 - server 3-19-20
 - Setup switches 3-18
 - shared copy 7-3
 - overview 3-2-3, 3-7
 - pop-up programs 3-19, 7-2
 - procedure 3-3
 - running Setup 3-3
 - system requirements 3-8
 - troubleshooting
 - MS-DOS mode setup 3-13, 14-9-10
 - pop-up programs 7-2
 - TSRs in memory 14-8-9
 - Windows mode setup 14-10
 - updating to new version
 - file version detection 3-9
 - files upgraded 3-10
 - from Windows 3.0 3-16-17
 - preparing for 7-3-4
 - prompt to upgrade 3-10
 - running Setup 3-3
 - search for existing version 3-9
 - system files 3-15-16
 - user and company names 3-13
 - WINDOWS directory 3-9-10
 - user and company name 3-13
 - Windows mode setup 3-7, 3-13-17
 - Windows Setup dialog box 3-13-14
- Interchange scheduling file format 13-10-17
- Interleave, hard disk 9-2
- Internal-stack-overflow error 14-29
- Institute of Electrical and Electronic Engineers.
See IEEE
- Interrupt
 - COM 6-33
 - conflicts 14-37
 - keyboard 6-31
 - request lines 1-27
 - timer 7-21
 - virtual devices 2-8
- [intl] section in WIN.INI 6-11-14

- IPX protocol
 - dual-net environment 1-33-34, 8-13
 - MSIPX implementation 8-15-17
 - network troubleshooting 14-5-7
- IRQ. *See* Interrupt

K

- Kernel files 2-3, 4-3
- Keyboard
 - driver files 4-4
 - SETUP.INF entries 5-15-18
 - speed 6-22
- [keyboard] section in SYSTEM.INI 3-12, 6-61-62

L**LAN**

- ArcNet network 1-23
- cabling media 1-13-17
- compatible with Windows for Workgroups 8-3-4
- data transmission through cables 1-17-18
- dual-net environment 1-33-34
- Ethernet 1-19-22
- network adapter cards 1-25-29
- protocols 1-31-34
- security 1-39
- specifications 1-24
- token ring 1-22-23
- topologies 1-10-13
- LAN Manager. *See* Microsoft LAN Manager network
- Lana numbers 7-9-10
- Language libraries
 - SYSTEM.INF entries 5-41
 - WIN.INI entry 6-13
 - Windows for Workgroups files 4-12
- lastdrive** command (MS-DOS) 9-15
- Layers, in networking models
 - IEEE 802 networking model 1-8-10
 - OSI networking model 1-3-8
- loadhigh** command (MS-DOS) 9-9, 9-15, 14-21
- Local area network. *See* LAN

Logging on

- in 386 enhanced mode 2-20
- in standard mode 2-14-15
- Microsoft LAN Manager network 8-4-5
- Novell NetWare server 8-20

Lotus 1-2-3 release 3.1 14-31-32

LPT ports 7-20

M

Macintosh 12-32

Macro languages 11-23-24, 12-27

Mail application 7-11

client workstation

- Edit MSMAIL.INI
- command 12-15-16
- installing commands 12-15
- installing custom message types 12-21-22
- View Schedule button 13-3

compared to Microsoft Mail 12-28

connectivity enhancements

- Extensions for Windows for Workgroups 12-29-35
- FAX Gateway B-17
- Wide Area Network B-23, B-26
- X.400 Gateway B-28-32
- PROFS B-34-39

custom commands 12-14-19

customization options 12-13

described 12-3

disk space 12-38

Help Request command 12-23-25

mail server 12-3

mail session manager 12-12

message

- address book 12-5
- attachments 12-4-5, 12-38-39
- composing offline 12-7-8
- custom message types 12-19-22
- deleted 12-39
- embedded objects 12-4-5
- exporting 12-8
- folders 12-6-7
- forms 12-4, 12-23-25, 13-7-9
- local outbox 12-8
- notification 12-12-13
- packaged object 12-37-38

Mail application, *continued*

- Schedule+ 13-7-9
- searching for 12-6-7, 12-39
- sending with Send Mail 12-27
- text editor 12-4

names service 12-12-13

notification engine 12-12

postoffice

- administrator 12-10, 12-36-37
- custom commands 12-15-18
- custom message types 12-22-23
- described 12-3
- directory structure 12-8-9
- disk space 12-38
- Empty Wastebasket
- command 12-18-19
- Help Request form 12-23-25
- mail server 12-3
- pathname in MSMAIL.INI 12-36, 12-39
- session manager 12-12
- shared folders 12-6
- SHARED.INI file 12-16

reinitializing 12-36

resource accounts 13-6-7

Schedule+

- messaging 13-2
- starting from Mail 13-3
- user accounts 13-6-7

spooler 12-11

starting an application from 12-15

system modules 12-11

transport 12-11

[Mail] section in WIN.INI 6-15

MAPI. *See* Messaging Application Program Interface

MCI (Media Control Interface) drivers 5-34-35, 6-16, 6-62

[mci extensions] section in WIN.INI 6-16

[mci] section in SYSTEM.INI 6-62

Memory

- base memory address 1-29
- freeing conventional 9-13-14
- HIMEM.SYS extended memory driver 2-7, 4-2
- memory drivers 9-16
- memory managers
- Setup 3-13

- Memory, *continued*
 - third party 9-14-16, 14-7
 - V86 2-9
 - MS-DOS 5.0 configuration 2-16
 - optimizing
 - desktop background 9-5
 - expansion board configuration 9-2-3
 - memory managers 9-14-16
 - MS-DOS Share TSR 9-4
 - options 9-7-8
 - upper memory blocks (UMBs) 9-8-16
 - RAM disk
 - diskless workstation 9-3
 - RAMDrive directory 9-3
 - RAMDrive version 3-16
 - running in upper memory 9-9
 - third party drivers 9-16
 - RAM start address 1-29
 - requirements
 - diskless workstation 9-3
 - LAN Manager server 8-7-8
 - memory managers 9-15
 - SMARTDrive 3-16
 - standard mode 2-13, 3-8
 - 386 enhanced mode 2-13-14, 3-8
 - starting network outside Windows for Workgroups 3-18
 - TEMP environment variable 9-8
 - 386 enhanced mode settings 6-30-56
 - upper memory area
 - managing memory blocks 9-8-16
 - shared-memory addresses 14-38
 - troubleshooting 14-5, 14-20-24
 - usage
 - 386 enhanced mode 2-6-11, 2-17
 - SHARE.EXE 2-19
 - standard mode 2-5-6, 2-13
 - V86 mode memory 2-8-11
- Messaging Application Program Interface (MAPI)
 - customizing Mail with 12-13
 - described 12-26
 - detecting on a workstation 6-15
 - functions 12-27
- Metafile 10-18-20
- Microsoft CD-ROM Extensions 7-13-14
- Microsoft Certified Professional Program A-12
- Microsoft Comm Server 1.1 8-12
- Microsoft Consulting Services A-10
- Microsoft Developer Network A-22
- Microsoft Draw 10-18
- Microsoft Excel for Windows 11-18, 11-23, 12-27
- Microsoft LAN Manager network
 - domain 8-7-8
 - encryption service 8-7
 - log on support 8-4-5
 - messenger support 8-6
 - Microsoft Comm Server 1.1 8-12
 - named pipes 8-6-7
 - NetBEUI protocol 8-5-6
 - NetBIOS interface 1-35
 - security 1-39, 8-4-5, 8-8-12
 - server
 - LAN Manager 8-7-8
 - Windows for Workgroups 8-3-4
 - Setup option 3-15
 - UNC redirection support 7-18
 - workstation
 - broadcasting 8-7-8
 - MS-DOS 8-6
 - Windows for Workgroups 8-3, 8-11
- Microsoft Mail
 - Add Paks 12-35
 - connectivity map B-10-12
 - configuring B-1-50
 - File Format API (FFAPI) 13-9-10
 - gateways 12-30-31
 - Microsoft Mail for PC LANs 12-28
 - Remote Workstation software 7-11
- [Microsoft Mail] section in
 - MSMAIL.INI 6-85-91, 12-16, 12-25, 12-36
- Microsoft Network
 - NetBIOS interface 1-35
 - Windows for Workgroups workstation 8-3-4
- Microsoft Product Support Services
 - Worldwide A-4
- Microsoft Remote Access Services 7-12
- [Microsoft Schedule] sections in
 - SCHDPLUS.INI 6-95-100
- Microsoft Solutions Channels—Consulting Program A-11
- Microsoft Solutions Channels—Training Program A-11
- Microsoft Support Services A-2, D-2

- Microsoft University Technical Training A-14
- Microsoft Word for Windows 11-23, 12-27
- [MMF] section in MSMAIL.INI 6-92-93
- Monolithic protocol stack 1-31
- Most recently used (MRU) connections
 - file 10-15-17
 - printer 10-21
 - stored in WIN.INI 6-16
- Mouse
 - boot driver 6-59
 - click settings 6-22
 - driver files 4-5
 - loading driver 9-4
 - loading in upper memory 9-9
 - MS-DOS driver 5-19
 - SETUP.INF information 5-18-19
- Moving files to a network share 7-17-19
- [MRU_Files] section in WIN.INI 6-16, 10-15-17
- [MRU_Printers] section in WIN.INI 6-16, 10-21
- MS-DOS
 - applications. *See MS-DOS-based applications*
 - configuration troubleshooting 14-20-24
 - device drivers 2-9
 - driver files 4-13
 - file system management 4-2
 - OEM versions 14-44
 - version 3-9, 9-3
 - Windows for Workgroups support
 - files 4-13-17
 - workstation. *See Workgroup Connection for DOS*
- MS-DOS 4.0 2-19
- MS-DOS 5.0
 - recommended version 9-3
 - upper memory area management 9-8-16, 14-20-24, 14-27-29
 - 386 enhanced mode 2-16
- MS-DOS-based applications
 - 386 enhanced mode 2-6-11, 2-16-17
 - exclusive mode 7-20-21
 - font
 - 386 grabber 5-15
 - font files 4-9, 4-11-12
 - screen fonts 6-32, 6-37
- MS-DOS-based applications, *continued*
 - grabber
 - 386 grabber fonts 5-15
 - data exchange support 4-14-15
 - incompatible 14-16
 - mouse support 6-64
 - SYSTEM.INI entries 6-57
 - icon file 4-20
 - installing with Setup 5-43
 - MS-DOS driver files 4-13-17
 - multitasking support 2-9
 - network processing time
 - statistics 10-12-13
 - optimizing performance 9-6
 - out-of-memory message 14-24
 - PIFs 5-43-47, 7-17-19
 - repainting screen 6-317-17-1932
 - scheduling programs 13-17
 - screen display access 6-32
 - standard mode 2-57-17-196, 2-127-17-1913
 - timer interrupts 7-21
 - troubleshooting
 - 386 enhanced mode 14-31
 - incompatible grabber 14-16
 - README files 14-32
 - standard mode 14-30-32
 - UNC redirection 7-17-19
 - virtual machine 2-7-8
 - Windows application data
 - exchange 4-14-15
- MS-DOS Extender for Windows 4-16
- MS-DOS mode setup
 - described 3-2
 - SETUP.INF entries 5-8-12, 5-36
 - SYSTEM.INF entries 5-40
 - troubleshooting 14-7, 14-9-10
 - using 3-9-13
- MSIPX protocol
 - dual-network environment 1-33-34
 - Windows for Workgroups
 - implementation 8-15-17
 - MSIPX.COM 1-33-1-34, 8-15-8-17
 - MSIPX.SYS 8-14, 8-16
 - relationship to Novell's IPX 8-16-17
 - workstation configured for 8-3
- MS-Net. *See Microsoft Network*

MSMAIL.INI

- administrator name 12-36-37
- custom message types 12-21-25, 13-7-9
- Edit MSMail.INI command 12-15-16
- editing 6-3-4
- installing Help Request form 12-23-25
- Schedule+ usage 13-3
- sections
 - [Address Book] 6-82
 - [CustomCommands] 6-82-83
 - [CustomMessages] 6-83-84
 - [Microsoft Mail] 6-85-91
 - [MMF] 6-92-93
 - list of 6-82
- server custom commands 12-15-18
- server pathname 12-36, 12-39
- workstation custom commands 12-14-19

Multi-boot programs 7-4**Multimedia**

- CD-ROM 7-13-14
- driver files 4-9
- extensions 2-3
- MCI driver
 - driver file 4-9
 - SYSTEM.INI entries 6-62
 - WIN.INI entry 6-16
- SETUP.INF entries 5-34-35
- troubleshooting 14-45-48

Multitasking

- foreground application priority 14-30
- standard mode 2-5
- 386 enhanced mode 2-6-7

N**Named pipes**

- Microsoft LAN Manager network 8-6-7
- Novell NetWare network 8-22

NDIS (Network Device Interface Specification)

- adapter card driver 2-11-13
- described 1-29-30
- driver files 4-8, 4-13-14

Net Watcher accessory

- identifying workstation connections 10-9-10
- terminating connections 10-9-11

NetBEUI transport protocol

- described 1-32-33

NetBEUI transport protocol, *continued*

- disabling 7-8-9
- dual-net environment 8-13
- Microsoft LAN Manager 8-5-6
- network driver 2-11-13
- network troubleshooting 14-5-7
- standard mode 2-13-14
- virtual device driver 2-16-17

NetBIOS interface

- described 1-35
- implementing 2-11-13
- LAN numbers 7-9-10
- Novell NetBIOS 8-19
- standard mode 2-13-14
- virtual device driver 2-16-17

[netcard] section in PROTOCOL.INI 6-74-75**NetWare. *See* Novell NetWare****NetWare button (dialog box) 10-2, 10-3-4****Network**

- 386 enhanced mode functionality 2-6-7, 2-14-17
- compatible with Windows for Workgroups 8-3-4
- connections
 - ClipBook Viewer 10-8-9, 11-13
 - File Manager interface 10-14-17
 - identifying workstation 10-9-10
 - MRU files 6-16, 10-15-17
 - MRU network drive 10-15-17
 - MRU printers 6-16, 10-21
 - net use command 2-17
 - Network DDE 11-14-16
 - opening with dialog box 10-2-4
 - persistent 2-13, 7-6-7
 - printer 10-20-21
 - secondary network 6-16-17
 - terminating inactive 10-5
 - terminating with Net Watcher 10-10-11
 - troubleshooting 14-38-42
 - UNC directory search 10-4-5
 - virtual machine 2-17
 - Windows connection 7-10-11
- dual-net environment 1-33-34
- input/output system. *See* NetBIOS
- logging on
 - 386 enhanced mode 2-20
 - standard mode 2-14-15

Network, *continued*

models

- central file server 1-37-38

- IEEE 802 1-8-10

- OSI (Open System Interconnection) 1-3-8

- peer-to-peer 1-37-39

- NetBIOS interface 1-35

- protocols 1-31-34

- standard mode functionality 2-5-6, 2-12-13

- starting 7-10-11

- topologies 1-10-13

- transports 1-31-34

- troubleshooting 14-37-42

- usage time statistics 10-12-13

Network adapter card

- base I/O port address 1-28

- base memory address 1-29

- changing settings 6-73

- configuring 1-26

- described 1-25

- detected by Setup 3-2

- driver files 4-13-14

- installable 5-52-56

- interface connectors 1-25-26

- interrupt request lines 1-27-30

- NDIS card driver 1-29, 2-11-13, 4-8, 4-13-14

- OEM (original equipment manufacturer) 5-64-68

- PROTOCOL.INI 6-73

- removing from Setup 5-69-70

- troubleshooting 14-37-39

Network button (dialog box) 10-2-3

Network DDE

- 386 enhanced mode 2-16

- API 11-24

- application macros 11-16-18

- ClipBook Viewer example 11-18-22

- ClipBook Viewer interface 11-13

- data exchange with ClipBook 10-6-9

- DDE links 11-16-18

- DDE shares 6-60, 11-11-12, 11-14-16

- described 11-11

- direct network example 11-22-23

- Hearts game 10-14

- identifying conversations 10-9-10

Network DDE, *continued*

- loading 11-24

- standard mode 2-13

- system components 11-12-13

- terminating conversations 10-10-11

- Windows 3.1 11-25

- workstation connections 10-8-9

Network Device Interface Specification. *See* NDIS

Network drivers

- driver files 4-8

- NETWORK.INF entries 5-50-63

- SETUP.INF entries 5-20-23

- standard mode 2-12-13

- troubleshooting connections 14-38-42

- virtual devices 2-16-19

- Windows for Workgroups 2-11-13

- Winnet driver 2-11, 2-14, 6-59

Network redirector

- driver 2-11-13

- Microsoft LAN Manager network 8-6

- services 1-35-36

- starting in standard mode 2-13-14

- virtual device driver 2-16-17

- Windows for Workgroups compatibility 8-3

Network Settings dialog box 7-14-15, 9-17-18

NETWORK.INF

- described 5-4, 5-50

- editing 5-5

- entries used in PROTOCOL.INI 6-73

- information file format 5-4-5

- removing drivers from 5-69-70

- sections

- [multinet] 5-57-58

- [multinet_install] 5-58-61

- [netcard] 5-52-54

- [netcard_install] 5-54, 5-59-61

- [netcard_protocol] 5-54, 5-62-63

- [nwddata] 5-51

- [protman] 5-52

- [protman_install] 5-52

- [transport] 5-55-56

- [transport_install] 5-56, 5-59-61

- [transport_protocol] 5-57, 5-62-63

- [workgroup] 5-51-52

- list of 5-50

- values used in SETUP.SHH 3-23-24

- [network.setup] section in
 - PROTOCOL.INI 6-74, 7-9-10
 - [network] section in SYSTEM.INI
 - computer name 1-35
 - disabling resource sharing 7-5
 - entries 6-64-70
 - workgroup 8-7-8
 - workstation broadcasting 8-7
 - [network] section in WIN.INI 6-16-17
 - Non-Windows applications. *See* MS-DOS-based applications
 - [NonWindowsApp] section in SYSTEM.INI 6-62-64, 9-5
 - Novell NetWare
 - ArcNet support 8-17-18
 - directories 8-21, 8-22
 - drive assignments 8-18, 8-20
 - drivers 8-13-15, 8-19
 - file access limit 8-20
 - MSIPX protocol 1-33-34, 8-3, 8-15-17
 - named pipes 8-23
 - NetWare dialog box button 10-2, 10-3-4
 - Novell NetBIOS 8-19
 - resources, connecting to 8-23
 - security 1-39
 - server
 - attaching 8-20
 - connectivity 8-15
 - printing to 8-20
 - Windows for Workgroups 8-3
 - Setup option 3-15
 - 386 enhanced mode 8-21, 8-22
 - token ring card 8-18
 - troubleshooting 14-39, 14-40-42
 - Windows for Workgroups workstation 8-13
 - workstation 8-13
- O**
- Object Linking and Embedding (OLE)
 - described 11-10-11
 - Mail messages 12-5
 - Microsoft Draw program 10-21
 - OLE links 11-17, 11-25
 - troubleshooting C-39
 - WIN.INI [embedding] section 6-10
 - Object Packager utility 12-37-38
 - OEM character set 4-9-10, 6-59
 - OEMSETUP.INF
 - described 5-4, 5-64
 - directory 5-64
 - editing 5-5
 - entries used in PROTOCOL.INI 6-73
 - example file 5-68
 - information file format 5-4-5
 - sections
 - [disk] 5-64
 - [multinet] 5-67
 - [multinet_install] 5-67
 - [netcard] 5-65
 - [netcard_install] 5-65
 - [netcard_protocol] 5-65-66
 - [transport] 5-66
 - [transport_install] 5-66
 - [transport_protocol] 5-67
 - Online Services A-21
 - Open System Interconnection. *See* OSI
 - Operating modes. *See* 386 enhanced mode; Standard mode
 - Optical fiber
 - data transmission 1-17-18
 - described 1-16-17
 - Optimization
 - disk-compression software 9-16
 - hard disk 9-4, 9-6-9
 - hardware 9-2-3
 - memory 9-7-14
 - MS-DOS 9-3
 - MS-DOS-based applications 9-6
 - network utilization statistics 10-12-13
 - TSRs 9-6
 - Windows for Workgroups 9-4-5
 - workstation 9-16-17
 - OS/2 server 8-3-4
 - OSI networking model
 - Application layer 1-7, 1-34
 - Data Link layer 1-5-6, 1-29-30
 - described 1-3
 - interface 1-5
 - layer numbers 1-3
 - NDIS interface 1-29-30
 - NetBIOS interface 1-35
 - Network layer 1-6, 1-31-34
 - Physical layer 1-5, 1-10
 - Presentation layer 1-34
 - protocol 1-4-5

OSI networking model, *continued*
 sending data 1-4-5, 1-7-8
 Session layer 1-6-7, 1-34
 structure 1-4
 Transport layer 1-6, 1-7, 1-31-34

P

Page swapping 6-48-49
 Page-mapping conflict 14-44
 Pageswap device 2-10
 Paging 9-8
 Parallel ports 7-20
 [Password Lists] section
 WIN.INI 7-7-8
 SYSTEM.INI 6-70
 Paste dialog box 10-6
 Paste Link command 11-8, 11-16-18, 11-20
 Paste Special command 11-17
 Peer-to-peer network 1-37-39
 Permanent swap file 4-22, 9-4-5, 9-8
 disk drive location 6-49
 size 6-49
 PIF (program information file)
 directory location 5-43-47
 foreground application priority 14-30
 identical executable filenames 5-42, 5-73
 Lotus 1-2-3 3.1 14-31
 Memory Locked options 14-31
 run application in exclusive mode 7-20-21
 setup information 5-43
 UNC redirection 7-17-19
 Video Memory option 14-31
 Pop-up programs 3-19, 7-2
 Port
 base I/O port addresses 1-28, 14-37
 configuring COM ports 14-34
 network-redirection printer port 14-35
 parallel 7-20
 serial port adapter setting 6-33
 serial printer settings 14-35
 WIN.INI printer entries 6-17-19
 [ports] section in WIN.INI 6-17-18, 7-20
 Postscript printers 4-7-8, 14-36-37
 Print Manager
 Connect dialog 7-8
 dedicated print server 9-5
 disable printer sharing 10-20-21

Print Manager, *continued*
 processing time statistics 10-12-13
 separator pages 9-5, 10-17-20
 WIN.INI [spooler] section 6-20-21
 Printer
 disabling printer sharing user
 interface 7-4-6
 driver files 4-6-8
 drivers in Control Panel 5-48-49
 identifying workstation
 connections 10-9-10
 most recently used connections 10-21
 removing from Setup 5-69-70
 system settings entry 3-28
 terminating connection to 10-10-11
 troubleshooting print problems 14-33-37,
 C-24-27
 virtual device driver
 Windows 3.1 2-10
 Windows for Workgroups 2-16-17
 Windows 2.0 6-9
 [PrinterPorts] section in WIN.INI 6-18-19
 Priority values for virtual machines 7-13-17
 Product accessibility
 deaf and hard of hearing users D-2
 single-handed users D-3
 blind and low vision users D-4
 PROGMAN.INI
 described 6-76
 editing 6-3-4
 sections
 [groups] 6-77
 [restrictions] 6-77-78
 [settings] 6-76
 updating group pathname 14-11
 Program Manager
 application directory 5-73
 creating groups 3-31
 custom groups 5-72-73
 customizing Windows for Workgroups 3-5
 initialization file. *See* PROGMAN.INI
 installation defaults 5-28-31
 installed groups 5-72-73
 invalid or damaged group 14-11
 Setup shell 5-12
 Startup Group 3-2, 3-31
 Startup group 7-17
 UNC redirection 7-17-19

[programs] section in WIN.INI 6-19
 [protmat] section in PROTOCOL.INI 6-74
Protocol
 Banyan VINES E-21
 DDE (Dynamic Data Exchange) 11-3
 defined 1-5
 described 1-28-29
 DLC E-14-20
 information file. *See* PROTOCOL.INF
 Microsoft LAN Manager network 8-5-6
 MSIPX 1-33-34, 8-3, 8-15
 multiple drivers on a network 1-33
 NDIS interface 1-29-30
 NetBEUI 1-32-33
 on client and server 8-3
 RAS E-8-13
 Server Message Block protocol 1-35-36
 standardized 1-29-30
 TCP/IP E-3-7
Protocol Manager
 described 1-30
 device driver information 6-73
 driver name 6-74
 installation options 5-52
 PROTMAN.DOS driver 2-11-13
 Windows 3.1 operating mode 2-11-13
PROTOCOL.INI
 described 6-73
 editing 6-3-4, 6-73
 sections
 [netcard] 6-74
 [netcard_protocol] 5-54, 5-62-63,
 5-65-66
 [network.setup] 6-74
 [protman] 5-52, 6-74
 [protocol] 6-75
 [transport_protocol] 5-57, 5-62-63,
 5-67
 used by Protocol Manager 2-11
 [protocol] section in PROTOCOL.INI 6-75

R

RAM disk
 diskless workstation 9-6
 RAMDrive
 directory 9-3
 diskless workstation 9-3

RAM disk, *continued*
 TEMP environment variable 9-8
 updating to new version 3-16
 size 14-25
 third party memory drivers 9-16
RAM start address 1-29
RAMDRIVE.SYS
 directory 9-3
 parameters 9-2
 running in upper memory 9-9
RAS (Remote Access Services) 7-12
Raster font files 4-10-11, 5-32-33
Read-only files 3-19, 7-3
README files
 disk space for 3-14
 filenames 4-21
 list of 3-14
 NETWORK.WRI 14-37
 PRINTERS.WRI 14-33
 README.TXT 6-11
 README.WRI 14-20
 troubleshooting information 14-3
Receiver services 1-35-36
Redirector. *See* Network redirector
Registration database 11-21
Reminder TSR program 13-2
Remote Access Services (RAS) 7-12
Removing a Windows component 7-4
 [restrictions] section
 PROGMAN.INI 6-77-78
 WINFILE.INI
 initialization file, described 6-81
 disabling file sharing 7-5, 10-15

S

SCHDPLUS.INI
 described 13-3
 editing 6-3-4
 sections 6-95-100
Schedule+
 appointment notification 13-2, 13-18
 archiving calendars 13-6
 auto-pick searches 13-18
 custom message types 12-21, 13-7-9
 data exchange with other schedulers 13-9
 date and time format 13-18
 described 13-2

Schedule+, *continued*

- Extensions for Windows for
 - Workgroups 12-29-35
 - forms 13-8-9
 - initialization file 6-95-100
 - initialization files 13-3
 - Interchange file format 13-10-17
 - Microsoft Mail File Format API (FFAPI) 13-9
 - multi-server network 13-20
 - offline calendar
 - copying to another computer 13-6
 - file extension 13-3
 - saving changes to 13-2
 - online calendar
 - access privileges 13-3, 13-19
 - file location 13-2
 - filename 13-3
 - scheduling appointments 13-3-4, 13-19
 - print sizes 13-19
 - recurring appointments 13-18
 - resource accounts 13-6-7
 - Schedule+ Add Paks 12-35
 - standalone or networked 13-2
 - starting from Mail 13-3
 - synchronizing calendars 13-4-5
 - upgrading with Extensions 13-17
 - user accounts 13-6-7
 - user assistant 13-19
 - user license 13-19
- Screen fonts
- loaded at startup 6-14
 - MS-DOS-based application 6-32, 6-37
- Screen savers
- deleting 4-22
 - files 4-21
 - installing custom 5-70
 - workstation performance 9-5
- Security
- Microsoft LAN Manager network 8-4-5, 8-8-11
 - Network DDE shares 11-14-16
 - types of 1-39
 - Windows for Workgroups 1-39
- Separator pages (printing) 9-5, 10-17-20
- Server
- central file server network 1-37-38

Server, *continued*

- controlling functionality 7-4-6
 - DDE conversations 11-6-8
 - DDE data links 11-7-8
 - DDE server names 11-3
 - dedicated print server 9-5
 - disk space for Windows for Workgroups 3-17
 - Mail postoffice
 - administrator 12-10, 12-36-37
 - archiving calendar data 13-6
 - custom commands 12-15-18
 - custom message types 12-22-23
 - directory 12-3
 - directory structure 12-8-9
 - disk space requirement 12-38
 - Empty Wastebasket command 12-18-19
 - Help Request form 12-23-25
 - pathname 12-36, 12-39
 - Schedule+ calendar file 13-2, 13-3
 - Schedule+ files 13-2
 - session manager 12-12
 - SHARED.INI file 12-16
 - Microsoft LAN Manager network
 - browsing 8-7-8
 - LAN Manager server 8-7-8
 - messages 8-6
 - Microsoft Comm Server 1.1 8-12
 - security 8-4-5, 8-8-11
 - multiple network access 1-33
 - network operating system 8-3
 - network workstation services 1-35-6
 - Novell NetWare network 8-14, 8-15, 8-20
 - peer-to-peer network 1-37-39
 - processing time statistics 10-12-13
 - Schedule+ 13-2
 - server process priority values 7-13-17
 - virtual device driver 2-16-17
 - Windows for Workgroups 8-43
 - workstation as 2-15
- Session 1-35
- Setup. *See* Windows for Workgroups Setup
- SETUP.INF
- [blowaway] section 3-13
 - described 5-3, 5-6
 - editing 5-5
 - information file format 5-4-5

SETUP.INF, continued

- pathname 3-4
 - removing drivers from 5-69-70
 - sections
 - code page 5-15-18
 - copy-files 5-24-28, 5-71-72
 - display driver 5-12-15
 - fonts 5-32-33
 - general installation 5-8-12
 - incompatible drivers 5-33-34
 - keyboard 5-15-18
 - list of 5-6-7
 - miscellaneous 5-34-37
 - mouse driver 5-18-19
 - network installation 5-20-23
 - Program Manager 5-72-73
 - Program Manager groups 5-28-31
 - system configuration 5-38-41
 - system fonts 5-23
 - TSR information 14-9
 - values used in SETUP.SHH 3-23-24
- SETUP.SHH system settings file 3-23-30**
- Share ClipBook Page dialog box 10-7**
- Share TSR (MS-DOS) 9-4**
- Share-level security 1-39, 8-8-11**
- Shared resources**
- browsing workgroup 7-8
 - central file server network 1-37
 - connecting to
 - common dialog boxes 10-2-3
 - files with File Manager 10-15
 - MRU network drive connections 6-16, 10-15-17
 - Novell NetWare server 8-23
 - printers with Print Manager 10-20
 - UNC directory searches 10-4-5
 - disabling file sharing 7-5, 10-15
 - disabling printer sharing 7-5, 10-20-21
 - disconnecting from
 - files 10-15
 - printers 10-20
 - identifying connections 10-9-10
 - LAN Manager network 8-8-11
 - MRU printer connections 10-21
 - Network DDE shares 11-14-16
 - Novell NetWare network 8-18
 - passwords 7-7
 - peer-to-peer network 1-37-39

Shared resources, continued

- persistent connections 7-6-7
 - remote access 7-11-12
 - security 1-39
 - server processing time 10-12-13
 - standard mode 2-13
 - terminating connections
 - inactive 10-5
 - with Net Watcher 10-10-11
 - 386 enhanced mode 2-14
 - UNC redirection 7-17-19
 - user interface 7-4-5
- SHARED.INI**
- custom commands 12-15-19
 - custom message types 12-21-25
 - described 6-94
 - editing 6-3-4
 - sections 6-94
- Single-handed users D-3**
- SMARTDrive**
- disk-compression software 9-16
 - installing 3-16
 - memory allocation 9-3
 - troubleshooting 14-29
 - version 9-3
- SMB (Server Message Block) protocol 1-35-36, 8-3**
- Soft font installers 4-7**
- Sound files**
- deleting 4-22
 - filenames 4-20-21
 - sound driver 6-59
- [sounds] section in WIN.INI 6-19-20**
- [spooler] section in WIN.INI 6-20-21, 7-5**
- Standard mode**
- grabber file 4-14-15
 - MS-DOS-based application
 - problems 14-30-31
 - starting Windows for Workgroups in 2-14-15
 - system requirements 2-13, 3-8
 - task swapping files 4-16
 - unable to start in 14-25-27
 - Windows 3.1 2-5-6
 - Windows for Workgroups 2-12-13
 - WinOldAp 4-14-15
- [standard] section in SYSTEM.INI 6-71-72**
- Star bus topology 1-11-12**

- Starting
 - applications
 - automatically 3-2, 3-31, 6-22
 - from Mail 12-13
 - with document file 6-10-11
 - network software 2-13-14, 7-10-11
 - SHARE.EXE 2-18
 - Windows for Workgroups
 - network connection 7-10-11
 - standard mode 2-14-15
 - 386 enhanced mode 2-20
 - Windows for Workgroups Setup 3-3
 - Startup Group (Program Manager) 3-2, 3-31, 7-17
 - Static binding 2-5
 - Swap file
 - application swap file setting 9-5
 - creating a permanent 9-4-5
 - deleting 4-22
 - disk drive location 6-48-49
 - disk-compression software 9-16
 - network drive 9-6
 - size 6-49, 9-8
 - temporary swap file setting 9-5
 - virtual device 2-10
 - System Editor 6-4-5
 - System Information screen 3-24
 - SYSTEM subdirectory
 - described 3-2
 - files installed in 5-11-12, 5-24-28, 5-71-72
 - information files in 5-3-4
 - SYSTEM.INI
 - described 6-29-30
 - editing
 - entries 6-3-4
 - source file 6-5
 - with system files 6-4-5
 - modified at system installation 3-11-12, 5-37
 - sections
 - [boot.description] 6-59
 - [boot] 6-57-59
 - [ClipShares] 6-60
 - [DDEShares] 6-60, 11-14
 - [drivers] 6-60
 - [keyboard] 6-61-62
 - list of 6-29
 - SYSTEM.INI, *continued*
 - [mci] 6-62
 - [network] 6-64-70
 - [NonWindowsApp] 6-62-64
 - [PasswordLists] 6-70
 - [standard] 6-71-72
 - [386enh] 6-30
 - shared resource user interface 7-4-6
 - virtual device drivers 2-9-11
 - Windows 3.0 3-15
- T**
- TCP/IP protocol E-3-7
 - TEMP environment variable 9-3, 9-8
 - Temporary swap file
 - deleting 4-22
 - disk drive location 6-49
 - hard disk space 9-8
 - hard drive setting 9-5
 - network drive 9-8
 - Thicknet
 - cabling connectors 1-25-26
 - coaxial cable 1-14
 - network 1-20, 1-21-22
 - specifications 1-24
 - Thinnet
 - cabling connectors 1-25-26
 - coaxial cable 1-14
 - network 1-20-21
 - specifications 1-24
 - troubleshooting 14-40
 - [386enh] section in SYSTEM.INI
 - entries 6-30-56
 - installed network adapter device driver 5-59-61
 - installed protocol device driver 5-60-61
 - modified at system installation 5-39-40
 - modified at system installation 3-12
 - optimizing MS-DOS-based applications 9-6
 - page-mapping conflicts 14-44
 - timer interrupts 7-21
 - UMB conflict troubleshooting 14-27-29
 - VGA cards requiring additional memory 14-18
 - virtual device drivers 2-9-11
 - virtual device support files 4-16-17

[386enh] section in SYSTEM.INI, *continued*

Virtual Display Device file
version 14-17-18

386 enhanced mode

Control Panel settings 3-31
grabber file 4-14-15
NetBEUI implementation 8-5-6
starting Windows for Workgroups
in 2-18-19
system requirements 2-14-15
system requirements 3-8
tracking CPU statistics 10-12-13
troubleshooting 14-27-30
troubleshooting 14-31
viewing shared resource
connections 10-9-10
virtual device drivers 2-16-19
virtual device drivers 2-8-11
virtual device support files 4-16-17
virtual machines 2-7-8
Windows 3.1 2-6-11
Windows for Workgroups 2-14-18
WinOldAp 4-14-15
workstation broadcasting 8-7-8

Token-ring network

described 1-12-13, 1-22-23
Novell NetWare network 8-18
specifications 1-24

Topologies

bus 1-10-11
defined 1-10
star bus 1-11-12
token ring 1-12-13

Transport

defined 1-31-32
layer

IEEE networking model 1-9
OSI networking model 1-7, 1-31

Transport protocol. *See* Protocol

Troubleshooting

clean boot configuration 14-5-7
desktop configuration 14-11
fonts 14-32-33
General Protection (GP) faults 14-42-45
fonts C-20-23
hardware C-11-19
incompatible TSRs 14-7-9, 14-11
MS-DOS configuration 14-20-24

Troubleshooting, *continued*

multi-boot programs 7-4
multimedia 14-45-48, C-40-45
networks 14-37-42
MS-DOS-based applications 14-30
operating problems C-28-39
options 14-3-4
printing 14-33-37, C-24-39
Setup 3-13, 7-2, C-1-10
386 enhanced mode 14-27-30
video display problems 14-15-20
Windows for Workgroups Setup 14-7

TrueType fonts

font files 4-11, 5-32-33
video display problems 14-32-33
WIN.INI entries 6-21

[TrueType] section in WIN.INI 6-18

TSR programs

devicehigh/loadhigh commands 14-21
incompatible drivers removed 3-11
incompatible with Windows for
Workgroups 14-7-9, 14-11
loading in upper memory 9-4, 9-8, 9-9-10
optimizing performance 9-6
running Setup 3-19
SETUP.TXT information file 14-8
uninstalling 14-9
virtual device 2-9

Twisted-pair cable

cabling connectors 1-25-26
described 1-15-16
Ethernet network 1-20, 1-22
network specifications 1-24

Typologies 1-20

U

UNC (universal naming conventions)

described 7-17-19
directory searches 10-4-5
password-protected resources 10-5
pointers in Mail 12-37-38

Unexpected MS-DOS Error #11 message 14-30

UNIX server 8-3-4

Upper memory area

loading TSRs and device drivers 9-4
managing memory blocks 9-8-16
optimizing EMM386.EXE 9-11-13

Upper memory area, *continued*
 shared-memory addresses in 14-38
 third party memory managers 9-14-16
 troubleshooting 14-5, 14-20-24
 UMB conflict troubleshooting 14-27-29
 User-level security 1-39, 8-11-12
 USER.EXE 2-3, 4-3
 Utilization statistics 10-12-13

V

V86 memory manager 2-10
 VBROWSE.386 2-16-2-17
 Vector font files 4-11, 5-32-33
 VIPX.386 8-14
 Virtual 8086 (V86) mode 2-7
 Virtual devices
 described 2-8
 Novell NetWare drivers 8-14
 386 enhanced mode support files 4-16-17
 virtual machines
 described 2-7-8
 priority values 7-13-17
 VxD driver reference 2-8
 Windows 3.1 drivers 2-9
 Windows for Workgroups 2-16-19
 Virtual Machine Manager 2-8
 Virtual memory swap file device 9-16
 VNETSUP.386 2-16-2-17
 VNB.386 2-16-2-17, 8-5-8-6
 VNETBIOS.386 2-16-2-17
 VREDIR.386 2-16-2-17, 7-10
 VSERVER.386 2-16-2-17, 7-13
 VSHARE.386 2-16, 2-18
 VWC.386 2-16-2-17

W

Wallpaper
 default 6-9
 deleting 4-22
 files 4-20
 memory usage 9-5
 Warning beep setting 6-22
 WIN.COM 3-12-13, 4-2-3
 WIN.INI
 described 6-6
 disable printer sharing 10-20-21

WIN.INI, *continued*
 editing
 entries 6-3-4
 source file 6-5
 with system files 6-4-5
 modified at system installation 5-37
 sections
 [colors] 6-7
 [desktop] 6-8-9
 [devices] 6-9
 [embedding] 6-10-11
 [extensions] 6-10
 [fonts] 6-14
 [FontSubstitutes] 6-15
 [Hearts] 6-11
 [intl] 6-11-14
 list of 6-6-7
 [Mail] 6-15
 [mci extensions] 6-16
 [MRU_Files] 6-16, 10-15-17
 [MRU_Printers] 6-16, 10-21
 [network] 6-16-17
 [ports] 6-17-18
 [PrinterPorts] 6-18-19
 [programs] 6-19
 [sounds] 6-19-20
 [spooler] 6-20-21
 [TrueType] 6-21
 [Windows] 6-22-27
 [WindowsHelp] 6-27-28
 values used in SETUP.SHH 3-23-24
 Windows 3.0 3-15
 Windows 2.0 printers 6-9
 Windows 2.x system font 4-9
 Windows 3.0 3-16-17
 Windows 3.1
 Advanced Power Management (APM) 10-13
 APIs 2-2-3
 core files 4-3
 device drivers 2-2-4
 dynamic link libraries (DLLs) 2-4-5
 extensions 2-2-3
 Network DDE 11-25
 priority values 7-13
 scheduler 7-13-17
 standard mode 2-5-6
 system font 4-9

- Windows 3.1, *continued*
 - 386 enhanced mode 2-6-7
 - virtual device drivers 2-8-11
 - virtual machines 2-7-8
- Windows applications. *See also* Applications;
MS-DOS-based applications
 - adding mail-enabled features 12-13
 - APIs 2-2-4
 - data exchange
 - Clipboard 11-4
 - ClipBook Viewer 10-6-9, 11-13, 11-16-18
 - dynamic data exchange (DDE) 11-3-9
 - macros with DDE 11-23-24
 - MS-DOS-based application 4-14-15
 - Network DDE 11-11-25
 - Object Linking and Embedding (OLE) 11-10-11
 - OLE 11-25
 - Paste Link command 11-8
 - General Protection (GP) faults 14-42-45
 - installing with Setup 5-70
 - mail-enabled features
 - adding with macros 12-27
 - adding with MAPI functions 12-23-25
 - detecting Mail functionality 6-15
 - user interface 12-27
 - optimizing performance 9-2
 - processing time priority values 7-13-17
 - 386 enhanced mode 2-8-11, 2-16-17
 - user preferences 6-6-7
 - video display problems 14-16
 - Windows 2.0 printers 6-9
 - Windows fonts 4-9
- WINDOWS directory
 - described 3-2
 - files installed in 5-11, 5-24-28, 5-71-72
 - installing Windows for Workgroups 3-9-10
 - naming shared 3-19
 - specifying on server at installation 3-19
 - system settings file entry 3-25
 - user path statement for shared 7-3
- Windows for Workgroups
 - accessories. *See* Accessories
 - dual-net support 1-33-34
 - installing. *See* Installing Windows for Workgroups
- Windows for Workgroups, *continued*
 - NetBIOS interface 1-35
 - network drivers 2-11-13
 - network workstation software 1-35-36
 - operating mode
 - 386 enhanced mode 2-15-19
 - standard mode 2-12-13
 - system requirements 3-8
 - priority values 7-13
 - protocols 1-31-34, 1-35-36, 8-3
 - reconfiguring 6-3-5
 - remote access to 7-11-12
 - share-level security 1-39
 - shared copy
 - read-only files 3-19, 7-3
 - required at installation 5-70
 - running from network 3-23
 - running from the network 7-3
 - setup 3-21-22
 - WINDOWS directory path statement 7-3
 - starting
 - 386 enhanced mode 2-20
 - network connection 7-10-11
 - standard mode 2-14-15
 - supported networks
 - clients and servers 8-3-4
 - Microsoft LAN Manager 8-4-12
 - networks with SMB protocol 8-3
 - Novell NetWare 8-13-23
 - system maintenance with Setup 3-31
 - 386 enhanced mode
 - troubleshooting 14-27-30
- Windows for Workgroups files. *See also*
individual filenames
 - accessories and applications 4-17-19
 - core 4-3
 - deleting 4-22, 9-5
 - drivers 4-4-9
 - fonts 4-9-13
 - grabber support files 4-14-15
 - information files 5-3-4
 - initialization files 6-1-2
 - language libraries 4-12
 - miscellaneous hardware 4-21
 - MS-DOS driver files 4-13-14
 - MS-DOS-based application icon 4-20
 - README 4-21

- Windows for Workgroups files, *continued*
 - Setup 4-19-20
 - sound 4-20-21
 - standard mode 4-16
 - standard mode startup file 4-2
 - 386 enhanced mode startup file 4-2
 - types of 4-2
 - wallpaper 4-20
 - WIN.COM 4-2-3
- Windows for Workgroups Setup. *See also*
 - Installing Windows for Workgroups
 - administrative setup 3-18, 3-19-20
 - automated setup 3-18, 3-20, 3-21-23
 - custom setup 3-6-7
 - customizing Windows for Workgroups 3-5
 - express setup 3-6
 - files 4-19-20
 - font setup 4-9-13
 - information files. *See* Installation information files
 - installing drivers 4-3
 - maintaining Windows for Workgroups 5-1
 - MS-DOS mode setup 3-7, 3-9-13
 - network setup 3-17, 3-21, 3-23-30
 - pop-up programs 3-19, 7-2
 - reconfiguring system 6-3-5
 - removing devices from 5-69-70
 - removing initialization files 7-4
 - running from a server 3-20
 - running with active TSRs 3-19
 - server setup 3-19-20
 - starting 3-3
 - startup switches 3-5
 - startup switches for networks 3-18
 - System Information screen 3-10-11
 - system maintenance 3-31
 - troubleshooting
 - causes of failure 14-7
 - failure while installing 3-13
 - general C-1-10
 - memory-resident software 14-8-9
 - MS-DOS mode setup 14-9-10
 - pop-up programs 7-2
 - Windows mode setup 14-10
 - TSR incompatible drivers removed 3-11
 - Windows for Workgroups installation 3-2-3
 - Windows mode setup 3-7, 3-13-17
- Windows mode setup
 - described 3-2
 - SETUP.INF entries
 - files installed 5-24-28
 - profile strings 5-36
 - stop reading SETUP.INF 5-37
 - troubleshooting 14-7, 14-10
 - using 3-7, 3-13-17
- Windows NT server 8-3-4
- Windows Setup dialog box 3-13-14
- [Windows] section in WIN.INI 6-22-27
- [WindowsHelp] section in WIN.INI 6-27
- WINFILE.INI
 - editing 6-3-4
 - resource sharing restriction entry 10-15, 7-5
 - sections 6-73-74
- WinMeter accessory 10-12-13
- Winnet driver
 - loading 2-14
 - MRU network connections 10-15-17
 - Novell NetWare 8-14
 - SYSTEM.INI entry 6-59
 - Windows for Workgroups 2-11
- WinOldAp 4-14-15
- Word for Windows 12-27
- Workgroup
 - adding workstation at installation 3-13
 - data exchange with ClipBook 10-6-9
 - file sharing with File Manager 10-14-16
 - installation defaults 5-51-52
 - LAN Manager domain 8-7
 - Mail application
 - administrator 12-10
 - connecting workgroups 12-30
 - mail server 12-3
 - shared folders 12-6
 - remote access to 7-11-12
 - Schedule+ user accounts 13-6-7
 - shared resources 7-8
 - standard mode 2-13-14
 - 386 enhanced mode 2-15-16
 - Workgroups network driver 2-11-13
- Workgroup Connection
 - bootstrapping workstation to network 3-18
 - message alerts 7-2
 - workstation access to shared resources 8-6
 - workstation service 1-35-36

Workstation

- computer name 1-35, 3-13-14
- dedicated print server 9-5
- disconnecting remote 10-10-11
- disk space 3-17
- identifying network connections 10-9-10
- in 386 enhanced mode 2-15-16
- in standard mode 2-12-13
- Mail client
 - customizing Mail 12-13
 - Edit MSMAIL.INI
 - command 12-15-16
 - installing custom Mail
 - command 12-15
 - mail server 12-3
 - message notification 12-12-13
 - message store 12-3
 - postoffice pathname 12-39
 - remote dial-in 12-31
 - View Schedule button 13-3
- Microsoft LAN Manager network
 - broadcasting 8-7-8
 - messages 8-6
 - MS-DOS workstation 8-6
 - workgroups 8-7-8
- MRU network connections 10-15-17
- MRU printer connections 10-21
- multiple server access 1-33-34
- network services 1-35-36
- Novell NetWare 8-13
- Novell NetWare network
 - Novell NetWare workstation 8-14
 - server access 8-14, 8-15
 - upgrading to Windows for Workgroups 8-14
 - Windows for Workgroups workstation 8-14
 - workstation software 8-17
- optimizing performance 9-16-17
- printer connections 10-20-21
- registration database 11-21
- Schedule+
 - appointment notification 13-3
 - calendar file 13-3
 - copying calendar file 13-6
- sharing data with ClipBook 10-6-9, 11-13
- sharing data with ClipBook Viewer 11-16-18

Workstation, *continued*

- utilization statistics 10-12-13
- Windows for Workgroups client 8-3-4
- Windows for Workgroups server 8-3-4

Windows for Workgroups Resource Kit Feedback Sheet

We update the *Resource Kit* as we release new versions of Windows, and we'd like you to tell us the type of additional information you'd like to see in the Kit. If there are topics you want to be covered in an update, please fill out this sheet and send it to the address at the bottom of the page. Your input will be valuable when we prepare future information for the Kit.

Your Feedback

My priorities for more information are (number these in order of importance):

Technical Overview

Installation and Setup

Special Topics

Configuring Windows

Using Windows

Troubleshooting

Other: _____

I found the following information in this Kit the most valuable: _____

I would like access to more utilities, such as: _____

I would find the information in this Kit more helpful if: _____

Please send your completed feedback sheet to:

Microsoft
Attn: *Windows for Workgroups Resource Kit*
One Microsoft Way
Redmond, Washington 98052-6399

The Windows for Workgroups Resource Kit Addendum

for Version 3.11

The Microsoft Windows for Workgroups Resource Kit Addendum for Version 3.11 provides complete technical information for support professionals on Windows for Workgroups version 3.11 and is an addendum to the *Windows for Workgroups Resource Kit for Version 3.1*, covering new and changed information and functionality. The Windows for Workgroups Resource Kit consists of the *Windows for Workgroups Resource Kit for Version 3.1* and the *Windows for Workgroups Resource Kit Addendum for Version 3.11* technical reference manuals, and a disk that contains helpful utilities and accessories.

Comprehensive

The *Windows for Workgroups Resource Kit Addendum for Windows for Workgroups 3.11* technical reference manual is the most comprehensive technical information source available for Windows for Workgroups version 3.11. Written at Microsoft by the group that developed Windows for Workgroups, it is designed for support professionals and advanced users who want to better understand and get the most out of Windows for Workgroups.

Contents

The topics covered in the *Windows for Workgroups Resource Kit Addendum for Version 3.11* include the following:

- A technical discussion of the 32-bit architecture for several new components in Windows for Workgroups 3.11
- Specific information about the changes made to the Setup procedure and information on customizing the Windows for Workgroups environment
- An overview of the new security control enhancements implemented in Windows for Workgroups 3.11
- A description of the new and changed entries present in the Windows for Workgroups Initialization Files, including SYSTEM.INI, MSMAIL.INI, and EFAXPUMP.INI
- Detailed information on integrating Windows for Workgroups 3.11 with Microsoft Windows NT and Windows NT Advanced Server, and Novell NetWare
- Information about the protocols supported by Windows for Workgroups 3.11 including NetBEUI, TCP/IP, Data Link Control (DLC)

protocol, and the new 32-bit IPX/SPX compatible transport

- Tips for configuring, optimizing, and troubleshooting Windows for Workgroups 3.11 to get the most out of your system

The topics covered in the *Windows for Workgroups Resource Kit for Version 3.1* include the following:

- A technical discussion of networking and the architecture used by Windows for Workgroups
- An overview of the Setup procedures, and a description of the contents of the Setup Information Files used by Windows for Workgroups which can be used to customize the Windows for Workgroups configuration
- A description of the entries present in the Windows for Workgroups Initialization Files, including WIN.INI, SYSTEM.INI, PROTOCOL.INI, PROGMAN.INI, CONTROL.INI, WINFILE.INI, MSMAIL.INI, and SCHDPLUS.INI
- A discussion of special topics related to new and updated accessories including ClipBook Viewer, Net Watcher, WinMeter, File Manager, and Print Manager
- A technical discussion of Network DDE, and how it extends dynamic data exchange to the workgroup environment

The Windows Resource Kit Library

Available separately, the *Windows Resource Kit* for Windows version 3.1 covers topics not discussed in the *Windows for Workgroups Resource Kit* including using MS-DOS-based applications, memory management, fonts, printing, and using Object Linking and Embedding (OLE). The *Windows Resource Kit* also includes a set of helpful utilities, drivers, and accessories, for getting the most out of the Windows operating system.

The *Windows for Workgroups Resource Kit* contains important information that will help you to get the most out of Windows for Workgroups version 3.11. Whether you support Windows for Workgroups in your company, or just want to know more about it, the *Windows for Workgroups Resource Kit* is an invaluable information source that should be added to your Windows reference library.



* 5 4 3 0 3 *

Microsoft®