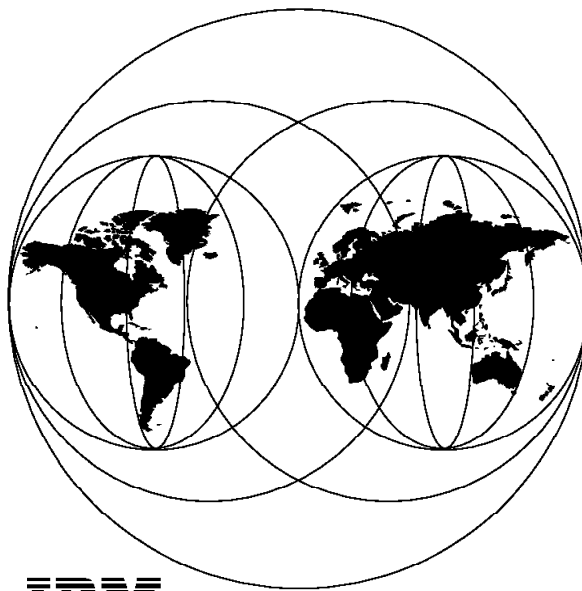


# **WorkSpace On-Demand Handbook**

December 1997



**IBM**

**International Technical Support Organization  
Austin Center**



SG24-2028-00

International Technical Support Organization

**WorkSpace On-Demand Handbook**

December 1997



**Take Note!**

Before using this information and the product it supports, be sure to read the general information in Appendix H, "Special Notices" on page 331.

**First Edition (December 1997)**

This edition applies to Version 1, Release Number 1 of WorkSpace On-Demand for use with OS/2 Warp Server. It was developed working with prerelease builds of the WorkSpace On-Demand product. This may result in small differences between what is documented in this edition and what you may see and experience with the product.

Comments may be addressed to:

IBM Corporation, International Technical Support Organization  
Dept. JN9B Building 045 Internal Zip 2834  
11400 Burnet Road  
Austin, Texas 78758-3493

When you send information to IBM, you grant IBM a non-exclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

© Copyright International Business Machines Corporation 1997. All rights reserved.

Note to U.S. Government Users — Documentation related to restricted rights — Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract with IBM Corp.

---

## Contents

<b>Figures</b> . . . . .	ix
<b>Tables</b> . . . . .	xiii
<b>Preface</b> . . . . .	xv
How This Redbook is Organized . . . . .	xv
The Team That Wrote This Redbook . . . . .	xvii
Comments Welcome . . . . .	xx
 <b>Chapter 1. Introduction to WorkSpace On-Demand</b> . . . . .	 1
1.1 WorkSpace On-Demand Enters the Network Computing Arena . . . . .	1
1.1.1 Workstations Available Today . . . . .	2
1.2 Let's Look at the Benefits of WorkSpace On-Demand . . . . .	5
1.2.1 Ease of Administration . . . . .	5
1.2.2 Security . . . . .	6
1.2.3 Cost of Ownership . . . . .	6
1.2.4 Application Support . . . . .	7
1.3 Making Up Your Mind for WorkSpace On-Demand . . . . .	8
 <b>Chapter 2. What is WorkSpace On-Demand?</b> . . . . .	 9
2.1 WorkSpace On-Demand: An Overview . . . . .	9
2.2 WorkSpace On-Demand and End-User Computing Environments . . . . .	10
2.2.1 End-User Environments Supported by WorkSpace On-Demand . . . . .	11
2.2.2 End-User Environments Not Appropriate for WorkSpace On-Demand . . . . .	12
2.3 WorkSpace On-Demand Components . . . . .	13
2.3.1 The WorkSpace On-Demand client Component . . . . .	13
2.3.2 The WorkSpace On-Demand Manager Component . . . . .	16
2.4 RIPL Functional and Usability Enhancements . . . . .	17
2.5 The Use of RIPL vs. BOOT-P Technology in WorkSpace On-Demand . . . . .	18
 <b>Chapter 3. RIPL and WorkSpace On-Demand</b> . . . . .	 19
3.1 RIPL Overview . . . . .	19
3.1.1 RIPL Hardware . . . . .	20
3.1.2 Understanding RIPL . . . . .	21
3.2 WorkSpace On-Demand RIPL Implementation . . . . .	25
3.2.1 The Remote IPL Control Files . . . . .	25
3.2.2 NDISDD.PRO Control File . . . . .	42
3.2.3 Other RPL Files and Directories . . . . .	44
3.2.4 Workstation Files . . . . .	47

<b>Chapter 4. WorkSpace On-Demand Planning</b>	51
4.1 Software Prerequisites	51
4.1.1 On the Target WorkSpace On-Demand server	51
4.1.2 On the Target WorkSpace On-Demand Administration Client	52
4.1.3 On the WorkSpace On-Demand Client	53
4.2 Hardware and Network	53
4.3 Network Topology	55
4.3.1 WorkSpace On-Demand and Network Capacity	55
4.4 Applications Supported by WorkSpace On-Demand	56
4.4.1 Applications Shipped with WorkSpace On-Demand	57
4.4.2 Applications Supported by WorkSpace On-Demand	57
4.4.3 Applications Compatible with WorkSpace On-Demand	57
4.4.4 Application-Specific Environment Variables	58
4.4.5 The Simplified, or Restricted, Shell: WorkSpace On-Demand Application Considerations	58
4.5 Migration Considerations	59
4.6 Resource Requirements	59
4.6.1 Processor	59
4.6.2 Memory and Hard Disk Required for Remote IPL Services	59
4.6.3 Hard Disk Required for WorkSpace On-Demand	60
4.6.4 Memory Required for WorkSpace On-Demand Image	62
4.6.5 Network Resources	62
4.7 Performance Planning for WorkSpace On-Demand	63
4.7.1 WorkSpace On-Demand Manager Server Performance Considerations	64
4.7.2 WorkSpace On-Demand client Performance Considerations	67
4.8 Capacity Planning and High-Availability for WorkSpace On-Demand servers	68
4.8.1 WorkSpace On-Demand servers, Load-Balancing and Fail-Over Services	69
4.8.2 Creating a Back-up WorkSpace On-Demand server	69
<b>Chapter 5. Installing WorkSpace On-Demand</b>	71
5.1 Before You Install WorkSpace On-Demand	71
5.2 Attended Installation of WorkSpace On-Demand Manager Components	72
5.2.1 First Installation Phase	73
5.2.2 Second Installation Phase	75
5.2.3 Post Installation Procedures	84
5.3 Unattended Installation of WorkSpace On-Demand Manager Components	84
5.4 Attended Installation of the WorkSpace On-Demand Administrator Client	86
5.4.1 First Installation Phase	86

5.4.2 Second Installation Phase . . . . .	87
5.5 Unattended Installation of the WorkSpace On-Demand Administrator Client . . . . .	90
5.6 Uninstalling WorkSpace On-Demand . . . . .	90
<b>Chapter 6. Basic WorkSpace On-Demand Administration . . . . .</b>	<b>95</b>
6.1 Overview of Client Setup . . . . .	95
6.2 The Administration Tools . . . . .	95
6.3 Defining a New Client . . . . .	98
6.3.1 Identity . . . . .	100
6.3.2 System . . . . .	101
6.3.3 Hardware . . . . .	102
6.3.4 Printers . . . . .	103
6.3.5 TCP/IP Configuration . . . . .	105
6.3.6 Workstation Definition . . . . .	107
6.4 Adding Application Software . . . . .	108
6.4.1 Defining a Simple OS/2 Application . . . . .	109
6.5 User Setup . . . . .	119
<b>Chapter 7. Advanced WorkSpace On-Demand Administration . . . . .</b>	<b>123</b>
7.1 Logon Options . . . . .	123
7.1.1 Logon Bitmap . . . . .	125
7.1.2 Changing the Background Bitmap . . . . .	125
7.2 Printing . . . . .	126
7.2.1 Network Printing . . . . .	126
7.2.2 Local Printing . . . . .	126
7.3 File Index Tables . . . . .	127
7.3.1 Client Machine FIT . . . . .	127
7.3.2 User FIT . . . . .	128
7.3.3 The Default User FIT . . . . .	128
7.3.4 FITs in Memory . . . . .	131
7.4 Setting Up Applications Under WorkSpace On-Demand . . . . .	132
7.4.1 OS/2 Applications . . . . .	132
7.4.2 Enabling the OS/2 Applications Supplied with WorkSpace On-Demand . . . . .	133
7.4.3 Netscape Navigator Version 2.02 . . . . .	133
7.4.4 Netscape Navigator - The Quick Way . . . . .	135
7.4.5 Java Runtime Version 1.0x . . . . .	136
7.4.6 Personal Communications Lite . . . . .	136
7.4.7 DOS and Windows Applications . . . . .	136
7.4.8 Special WorkSpace On-Demand Launch Parameters . . . . .	137
7.5 Running Java Applications from the WorkSpace On-Demand Desktop . . . . .	138
7.6 Adding a Network Adapter . . . . .	139

7.6.1 Enabling Supported Adapters . . . . .	139
7.6.2 Enabling Unsupported Adapters . . . . .	141
<b>Chapter 8. WorkSpace On-Demand Machine Classes . . . . .</b>	<b>149</b>
8.1 Machine Classes: An Overview . . . . .	149
8.1.1 What is a Machine Class? . . . . .	149
8.1.2 Machine Class File Structure . . . . .	151
8.1.3 Components of a WorkSpace On-Demand Machine Class . . . . .	151
8.1.4 Machine Classes Shipped with WorkSpace On-Demand . . . . .	152
8.1.5 Modifying Machine Classes . . . . .	153
8.1.6 Machine Class Feasibility Tests . . . . .	153
8.2 Implementing Machine Classes: Getting Started . . . . .	154
8.2.1 Utilities and Tools Needed to Create a Machine Class . . . . .	155
8.2.2 Choosing a Prototype Machine Class . . . . .	155
8.2.3 Creating a New Machine Class's Directory Structure . . . . .	156
8.3 Creating Machine Classes: An Example . . . . .	157
8.3.1 Creating a Machine Class for a Specific Video Adapter . . . . .	157
8.3.2 Creating a Machine Class with SCSI Support . . . . .	172
8.3.3 Creating a Machine Class for a CD-ROM . . . . .	177
8.3.4 Mouse Considerations . . . . .	182
8.3.5 Keyboard Considerations . . . . .	185
8.3.6 Serial Port Drivers . . . . .	188
<b>Chapter 9. Administering WorkSpace On-Demand in the Enterprise . . . . .</b>	<b>193</b>
9.1 WorkSpace On-Demand in an Enterprise Environment: An Overview . . . . .	193
9.2 Hardware Configurations and Resulting Machine Classes . . . . .	195
9.3 System Software Necessary in the WorkSpace On-Demand Environment . . . . .	196
9.3.1 IBM OS/2 Access Feature in the WorkSpace On-Demand Environment . . . . .	199
9.3.2 IBM Personal Communication/3270 for OS/2 in the WorkSpace On-Demand Environment . . . . .	203
9.3.3 IBM TME 10 NetFinity in the WorkSpace On-Demand Environment . . . . .	206
9.3.4 General Guidelines for Implementing Middleware . . . . .	208
9.4 Infrastructure Considerations . . . . .	211
9.5 The Management of WorkSpace On-Demand Management Servers . . . . .	216
9.5.1 Unattended Installation of WorkSpace On-Demand . . . . .	217
9.5.2 Managing the WorkSpace On-Demand Server Using the Command Line Interface . . . . .	229
<b>Appendix A. Tuning WorkSpace On-Demand . . . . .</b>	<b>253</b>
A.1 Performance Considerations . . . . .	253



A.2 Java Runtime Performance Considerations . . . . .	254
<b>Appendix B. WorkSpace On-Demand Hints and Tips . . . . .</b>	<b>255</b>
B.1 OS/2 Warp Server SYS0071 Errors When Attempting to Start an Application . . . . .	255
B.2 WorkSpace On-Demand client Network Adapter Configuration Failures . . . . .	255
B.2.1 Phase 0: Message Numbers Missing from Screen . . . . .	256
B.2.2 Phase 1: RQ Counter Increments, with No Other Result . . . . .	256
B.2.3 Phase 2: Failure During DOS Driver Initialization . . . . .	257
B.2.4 Phase 3: 3174 Message - Can't Find OS2LDR.MSG . . . . .	257
B.2.5 Phase 4: Various Early Boot Failures . . . . .	258
B.2.6 Phase 5: Failure of the OS/2 Redirector (NETWKSTA.200) . . . . .	258
B.3 Access Control Issues . . . . .	259
B.4 File Index Table (FIT) Considerations . . . . .	259
B.4.1 WorkSpace On-Demand FIT Files . . . . .	260
B.4.2 Using RDRDEBUG to View the Client FIT . . . . .	260
B.5 Changing System File Considerations . . . . .	261
B.6 LAN Adapter Address Consideration . . . . .	261
B.7 Undocumented Internal Public Application Parameters . . . . .	262
B.8 Caution Required When Naming Public Applications . . . . .	263
B.9 Public Application Command Parameter 119 Character Limitation . . . . .	264
B.10 Adding Application Parameters to a Public Application . . . . .	264
B.11 Disabling of an IBM WorkSpace On-Demand Requester . . . . .	264
B.12 Upgrade to TCP/IP for OS/2 Version 4.1 . . . . .	264
B.12.1 Install TCP/IP for OS/2 4.1 . . . . .	265
B.12.2 Copy the TCP/IP files to WorkSpace On-Demand Directory . . . . .	266
B.13 Consideration for TCP/IP Configuration Using DDNS . . . . .	267
<b>Appendix C. Interesting Web Pages to Read . . . . .</b>	<b>269</b>
C.1 WorkSpace On-Demand Home Page . . . . .	269
C.2 WorkSpace On-Demand SAPR Guide . . . . .	269
C.3 Understanding Network-Computing Fundamentals . . . . .	269
C.4 Web Pages That Referred to Performance . . . . .	269
C.5 WorkSpace On-Demand Web Pages for Related Products . . . . .	270
<b>Appendix D. Network Adapter Considerations and Work-Arounds . . . . .</b>	<b>271</b>
D.1 IBM Turbo Token-Ring Card Issues . . . . .	271
D.1.1 Step One: Copy the Updated Device Drivers to WorkSpace . . . . .	272
D.1.2 Step Two: Update RPL.MAP - ADD OTKTTR . . . . .	272
D.1.3 Step Three: Edit Boot Block Definition File (.CNF) . . . . .	273
D.1.4 Step Four: Edit NDISDD.PRO . . . . .	273
D.1.5 Step Five: Edit IBMDXM.NIF . . . . .	274
D.2 DOS Network Driver Errors . . . . .	274

D.2.1 IBM Auto Wake Token-Ring ISA Adapter . . . . .	274
D.2.2 IBM PCI Wake On LAN Token-Ring Adapter . . . . .	274
<b>Appendix E. RIPL Workstation Error/Status Messages . . . . .</b>	<b>277</b>
<b>Appendix F. Tools for Modifying and Managing CONFIG.SYS and INI Files in Machine Classes . . . . .</b>	<b>281</b>
F.1 Creating a Delta INI File . . . . .	281
F.2 Merging the Target and Delta INI Files . . . . .	283
F.3 Copying an Application from One INI File to Another . . . . .	284
F.4 Comparing Two CONFIG.SYS Files . . . . .	285
<b>Appendix G. Samples for Managing WorkSpace On-Demand in the Enterprise . . . . .</b>	<b>297</b>
G.1 WSODOAF.RSP: Sample OS/2 Access Feature Response File . . . . .	297
G.2 PCOMOS2.WS: Sample Personal Communication/3270 for OS/2 Configuration File . . . . .	301
G.3 WSODINST.CMD: Installing WorkSpace On-Demand Unattended . . . . .	304
G.4 QUERYCFG.CMD: Gathering the Client's Hardware Configuration . . . . .	312
G.5 ADDNEWCL.CMD: Automatted WorkSpace On-Demand Client Definition . . . . .	314
G.6 ADDAPPL.CMD: Creating WorkSpace On-Demand Applications Unattended . . . . .	321
G.7 ADDAPPL.RSP: Responsefile for Application Creation . . . . .	329
<b>Appendix H. Special Notices . . . . .</b>	<b>331</b>
<b>Appendix I. Related Publications . . . . .</b>	<b>335</b>
I.1 International Technical Support Organization Publications . . . . .	335
I.2 Redbooks on CD-ROMs . . . . .	335
I.3 Other Publications . . . . .	335
<b>How to Get ITSO Redbooks . . . . .</b>	<b>337</b>
How IBM Employees Can Get ITSO Redbooks . . . . .	337
How Customers Can Get ITSO Redbooks . . . . .	338
IBM Redbook Order Form . . . . .	339
<b>List of Abbreviations . . . . .</b>	<b>341</b>
<b>Index . . . . .</b>	<b>343</b>
<b>ITSO Redbook Evaluation . . . . .</b>	<b>345</b>

## Figures

1. Network-Computing Client Offerings . . . . .	3
2. WorkSpace On-Demand Components . . . . .	13
3. RIPL Process and Control Files . . . . .	23
4. RPL.MAP File . . . . .	27
5. RPL.MAP Workstation Record with BBCLT01 Definition Using an IBM Token-Ring Adapter . . . . .	28
6. RPL.MAP Server Record for an IBM Token-Ring Adapter . . . . .	28
7. A Sample RPL.MAP File . . . . .	29
8. WorkSpace On-Demand Boot Block Definition File BB1USNTR.CNF . . . . .	36
9. <User>.FIT Entry - NETSCAPE.INI Redirection . . . . .	42
10. <User>.FIT Entry - User FIT File Redirection . . . . .	42
11. NDISDD.PRO with IBM Token-Ring Entry . . . . .	44
12. Directories and Files . . . . .	45
13. Select Location for Netscape and Java Files . . . . .	75
14. Select WorkSpace On-Demand Components . . . . .	76
15. WorkSpace On-Demand Client Components . . . . .	78
16. IBM Software Registration - WorkSpace On-Demand . . . . .	82
17. Select Method of Submitting Registration . . . . .	83
18. Unattended Install REXX File . . . . .	85
19. Select CLI and API Upgrade Option for OS/2 Warp Version 4 Client . . . . .	88
20. The Administrator GUI on an OS/2 Warp Version 4 Client . . . . .	89
21. The Remove WorkSpace On-Demand Feature . . . . .	91
22. Selectively Uninstalling WorkSpace On-Demand . . . . .	92
23. The Removal Complete Window . . . . .	93
24. OS/2 Warp Server Administration GUI . . . . .	97
25. WorkSpace On-Demand Workstation Definition . . . . .	99
26. Identity Page of Notebook . . . . .	100
27. System Page of Client Definition Notebook . . . . .	101
28. Hardware Page of Client Definition Notebook . . . . .	102
29. Printer Page of Notebook . . . . .	104
30. TCP/IP . . . . .	105
31. Workstation Record Entry for a WorkSpace On-Demand Client . . . . .	107
32. Directory Alias - Create Notebook . . . . .	110
33. Access Control Profile Does Not Exist Window . . . . .	110
34. Access Control Profile - Settings View Notebook . . . . .	111
35. Propagate Access Profile to Subdirectories Window . . . . .	111
36. Public Application Definitions Folder . . . . .	112
37. WorkSpace On-Demand Application Definition - Create Settings Notebook . . . . .	113
38. WorkSpace On-Demand Application Definition - Program Location Page . . . . .	114

39.	WorkSpace On-Demand Application Definition - Work Directory Page . . . . .	115
40.	WorkSpace On-Demand Application Definition - Network Resources . . . . .	116
41.	WorkSpace On-Demand Application Definition - Parameters . . .	117
42.	User Notebook - Applications . . . . .	120
43.	User Notebook - . . . . .	121
44.	Sample Fit Entry . . . . .	127
45.	Default User FIT File - BB10USDU.FIT . . . . .	129
46.	Sample User FIT File Including the Default User FIT . . . . .	129
47.	Parameters for Netscape Navigator . . . . .	134
48.	User FIT Extensions for Netscape Navigator . . . . .	135
49.	Invocation Page for the Java Configuration Applet . . . . .	139
50.	Enabling LANStreamer MC32 Support in the RPL.MAP File . . . .	141
51.	Madge PROTOCOL.INI File for DOS . . . . .	145
52.	Madge PROTOCOL.INI for OS/2 . . . . .	146
53.	WorkSpace On-Demand Boot Block Definition File BB1USMDG.CNF . . . . .	147
54.	WorkSpace On-Demand NDIS Driver Profile File, NDISDD.PRO . .	148
55.	WorkSpace On-Demand RPL.MAP File, Server Record Entry . . .	148
56.	Files Changed by the Cirrus Logic Install Process . . . . .	161
57.	CPQ5120E Machine Class Subdirectory . . . . .	168
58.	CPQ5120E.FIT File . . . . .	169
59.	CONFIG.SYS Changes . . . . .	170
60.	WIN.INI Changes . . . . .	171
61.	SYSTEM.INI Changes . . . . .	172
62.	350WSCSI.FIT File . . . . .	175
63.	CONFIG.SYS Changes . . . . .	176
64.	Files Changed by the CD-ROM Install Process . . . . .	179
65.	ISACD.FIT File . . . . .	181
66.	CONFIG.SYS Changes . . . . .	182
67.	Mouse Drivers . . . . .	184
68.	Structure of an Enterprise Environment . . . . .	194
69.	Application Structure . . . . .	198
70.	System Structure for an Application in a WorkSpace On-Demand Environment . . . . .	199
71.	FFST/2 Files Necessary for OS/2 Access Feature . . . . .	200
72.	CONFIG.SYS Changes for OS/2 Access Feature . . . . .	202
73.	Sample for Additions in Client's FIT File for OS/2 Access Feature Support . . . . .	203
74.	Sample REXX Script to Create the PCOMM/3270 ICON on a Client's Desktop . . . . .	205
75.	Sample RC File to Modify the Model OS2.INI . . . . .	205

76.	FIT File Extensions for Personal Communication/3270 for OS/2/3270 Support . . . . .	206
77.	CONFIG.SYS Changes for Personal Communication/3270 for OS/2/3270 Support . . . . .	206
78.	Sample REXX Script to Create a TME 10 NetFinity Icon on a Client's Desktop . . . . .	207
79.	Sample RC File to Modify the Model OS2.INI . . . . .	208
80.	Sample RC File to Modify an OS2.INI File . . . . .	210
81.	Infrastructure of Enterprise Networks . . . . .	211
82.	LAN Segments Connected by Bridges . . . . .	213
83.	LAN Segments Connected by a Backbone . . . . .	214
84.	WorkSpace On-Demand WSODCID.RSP Response File . . . . .	217
85.	NCINVAR.RSP - The Variable Part of NCINSTAL.RSP . . . . .	224
86.	Syntax for the NET RIPLMACH Command . . . . .	230
87.	Example for a Response File to Create New WorkSpace On-Demand Clients . . . . .	231
88.	Syntax for the NET RIPLMACH Command . . . . .	234
89.	Sample REXX Command to Delete Clients Using a Response File . . . . .	235
90.	Sample Response File to Delete Clients . . . . .	235
91.	NET RIPLMACH - Sample Output for a Specific Client . . . . .	236
92.	Sample Program to Gather Client Information . . . . .	237
93.	Automated Client Detection and Definition . . . . .	240
94.	RPL.MAP file with Model Client Definitions . . . . .	242
95.	Model Client CONFIG.SYS Changes . . . . .	243
96.	FIT File Changes for a Model Client . . . . .	243
97.	Dynamic Detection of Network Adapter Name . . . . .	245
98.	RPL.MAP File with Model Client Definitions . . . . .	246
99.	RPL.MAP File with Model Client Definitions . . . . .	246
100.	RPL.MAP File with Model Client Definitions . . . . .	246
101.	TME 10 NetFinity System Information Tool Output . . . . .	248
102.	TME 10 NetFinity System Information Tool Video Adapter Detection . . . . .	249
103.	Output of NET RIPLMCLASS Command . . . . .	250
104.	Output of NET RIPLMCLASS IBM350 /OS:BB10.US . . . . .	250
105.	Network Adapter Address in the PROTOCOL.INI File . . . . .	261
106.	README.APP Set Up Information from WorkSpace On-Demand CD-ROM . . . . .	263
107.	BB350.FIT . . . . .	267
108.	Sample Logon Script File - PROFILE.CMD . . . . .	268
109.	Non-NDIS DOS Device Drivers . . . . .	272
110.	RPL.MAP . . . . .	272
111.	BB1USTTR.CNF . . . . .	273
112.	NDISDD.PRO . . . . .	273
113.	IBMDXM.NIF . . . . .	274

114.	BB1USTRP.CNF	275
115.	PROTOCOL.INI	276
116.	INICREATEDELTA4.CMD Create Delta INI File REXX Utility	281
117.	CopyApplication.CMD Copy Application REXX Utility	283
118.	INIMERGE.CMD Merge INI File REXX Utility	283
119.	INICOPYAPPLICATION.CMD Copy INI File REXX Utility	284
120.	COMPCFRS.CMD Compare Two CONFIG.SYS Files REXX Utility	285
121.	WSODOAF.RSP OS/2 Access Feature Client Response File	298
122.	PCOMOS2.WS Personal Communication/3270 for OS/2 Configuration File	301
123.	WSODINST.CMD - REXX Batch to Install WorkSpace On-Demand	305
124.	QUERYCFG.CMD - REXX Batch for Gathering Client's Hardware	313
125.	ADDNEWCL.CMD Program to Automatically Create WorkSpace On-Demand Clients	314
126.	ADDAPPL.CMD - REXX Batch to Create WorkSpace On-Demand Application Definitions	322
127.	WorkSpace On-Demand ADDAPPL.RSP - Response File to Create WorkSpace On-Demand	329

## Tables

1. WorkSpace On-Demand RIPL Functional and Usability Enhancements . . . . .	17
2. Description of the Workstation Record Fields . . . . .	30
3. Description of the Server Record Fields . . . . .	31
4. Description of the .CNF (Boot Block Definition File) Entries . . . . .	36
5. Description of the NDISDD.PRO File . . . . .	43
6. Files Associated with the RIPL Client . . . . .	48
7. Additional Memory Requirements for Remote IPL Service . . . . .	60
8. Disk Requirements for WorkSpace On-Demand System Components . . . . .	61
9. Relevant Remote IPL Parameters . . . . .	63
10. Matrix of Machine Classes for a Given Hardware Combination . . . . .	150
11. SCSI Adapters . . . . .	176
12. Mouse Drivers . . . . .	183
13. Supported Pointing Devices . . . . .	184
14. Keyboard Files . . . . .	185
15. Keyboard Layouts . . . . .	187
16. Subcountry Parameter . . . . .	187
17. Files Used to Access the Serial Port . . . . .	188
18. Standard I/O Port Addresses and IRQ Assignments . . . . .	190
19. Standard Definitions of IRQs . . . . .	190
20. System Software Necessary to Provide Special Functions . . . . .	197
21. Description of the Token-Ring RPL Messages . . . . .	277
22. Description of the Ethernet-Specific Messages . . . . .	279





---

## Preface

This redbook will help you install, configure and administer a WorkSpace On-Demand environment. It is the result of a residency conducted at the International Technical Support Organization, Austin Center, during the final development of the first release of the WorkSpace On-Demand product. The residents working with this first release of WorkSpace On-Demand have taken their experiences and knowledge gained during the residency and captured it in this redbook. By using this redbook, you can capitalize on their experiences as you work with WorkSpace On-Demand in your environment.

---

## How This Redbook is Organized

The authors of this redbook have organized the knowledge they want to share with you in the following manner:

- Chapter 1, "Introduction to WorkSpace On-Demand" provides a general overview of network computing and the positioning of WorkSpace On-Demand as a network computing implementation.
- Chapter 2, "What is WorkSpace On-Demand?" describes WorkSpace On-Demand and its components. It also gives an overview of the WorkSpace On-Demand product and discusses those end-user computing environments that are appropriate to migrate to WorkSpace On-Demand.
- Chapter 3, "RIPL and WorkSpace On-Demand" contains an introduction to Remote Initial Program Load (RIPL) and its implementation in WorkSpace On-Demand. To gain a complete technical understanding of WorkSpace On-Demand, you will need to be familiar with RIPL and the WorkSpace On-Demand use of the RIPL feature of OS/2 Warp Server. So that all readers can understand this important concept, this chapter provides information of the WorkSpace On-Demand RIPL implementations without requiring any previous knowledge or experience with RIPL.
- Chapter 4, "WorkSpace On-Demand Planning" gives you guidance on the planning steps for the installation of WorkSpace On-Demand. In addition to the hardware and software prerequisites, some rule-of-thumb planning information will help you prepare for installing WorkSpace On-Demand.
- Chapter 5, "Installing WorkSpace On-Demand" covers the installation of WorkSpace On-Demand. It gives you step-by-step guidance for installing WorkSpace On-Demand server and the WorkSpace On-Demand

Administrative Client. If you choose to install them remotely, there is information provided for doing unattended installation of WorkSpace On-Demand server and WorkSpace On-Demand Administrative Client.

- Chapter 6, “Basic WorkSpace On-Demand Administration” covers WorkSpace On-Demand administration using the administration Graphical User Interface (GUI). You can follow the guidance provided to set up your first WorkSpace On-Demand client. Guidance on basic tasks, such as defining the client machine, network adapter, TCP/IP, printer, and adding installed applications, is provided.
- Chapter 7, “Advanced WorkSpace On-Demand Administration” provides you with information and directions on administering your WorkSpace On-Demand environment beyond the topics covered in Chapter 6, “Basic WorkSpace On-Demand Administration.” You can analyze and set up both public and private applications, enable supported and unsupported network adapters, set up local and private printing, and perform other administrative tasks using the information provided in this chapter.
- Chapter 8, “WorkSpace On-Demand Machine Classes” discusses the fundamental concepts of WorkSpace On-Demand Machine Classes. A methodology for analyzing and creating Machine Classes is presented. You can follow this methodology to add Machine Classes to your WorkSpace On-Demand server. By adding Machine Classes, you will be able to extend WorkSpace On-Demand so that you can utilize existing and new hardware beyond the support provided in the product. An example of adding a new Machine Class to WorkSpace On-Demand is provided to help you understand how to implement the methodology presented.
- Chapter 9, “Administering WorkSpace On-Demand in the Enterprise” covers topics of interest to administrators of WorkSpace On-Demand in an enterprise environment. You are presented with suggestions on how to best manage WorkSpace On-Demand in an enterprise, including setting up and using response files and using the command line interface.
- Appendix A, “Tuning WorkSpace On-Demand” provides information on performance and WorkSpace On-Demand.
- Appendix B, “WorkSpace On-Demand Hints and Tips” is where the residents have noted the useful information they discovered working with WorkSpace On-Demand while preparing this redbook.
- Appendix C, “Interesting Web Pages to Read” lists several URLs which you may want to visit to find out more about WorkSpace On-Demand and related topics.

- Appendix D, “Network Adapter Considerations and Work-Arounds” gives you directions on how the residents working on this redbook were able to get the IBM Token-Ring Card to Remote IPL the WorkSpace On-Demand client using non-NDIS drivers.
- Appendix E, “RIPL Workstation Error/Status Messages” lists a number of Remote IPL error and status messages. You will find this list valuable if you are having problems initially setting up your WorkSpace On-Demand clients to Remote IPL.
- Appendix F, “Tools for Modifying and Managing CONFIG.SYS and INI Files in Machine Classes” contains the source listing for the REXX tools used in building new Machine Classes as described in Chapter 8, “WorkSpace On-Demand Machine Classes.” Use of these tools will ease your job when creating new Machine Classes in your WorkSpace On-Demand environment.
- Appendix G, “Samples for Managing WorkSpace On-Demand in the Enterprise” contains the source listing for the sample REXX tools and the sample response files that were used in the examples in Chapter 9, “Administering WorkSpace On-Demand in the Enterprise”
- Appendix H, “Special Notices” provides the intent and purpose of this redbook and lists the various trademarked products that are found throughout this redbook.
- Appendix I, “Related Publications” lists other publications for WorkSpace On-Demand and related products discussed in this redbook. At the end of this Appendix, information on where and how to order this and other redbooks is provided.

---

## The Team That Wrote This Redbook

This redbook was produced by a team of specialists from around the world working at the International Technical Support Organization Austin Center.

**Mike Foster** is a senior programmer at the International Technical Support Organization, Austin Center and holds a bachelor of science degree in electrical engineering from the University of Kansas. He writes extensively and teaches classes in all areas of OS/2 and LAN systems. Before joining the ITSO in 1995, he held both management and technical positions in IBM marketing and development divisions for over 25 years.

**Martin Bense** is a CID and communications specialist in the GAD Gesellschaft fuer automatische Datenverarbeitung eG in Muenster, Germany. He has five years experience in the OS/2 environment and during the last three years his team has designed and implemented a CID

environment based on NetView DM MVS and NetView DM/2. This CID solution supports more than 400 connected banks in over 2000 locations. He was a contributing author to the *OS/2 Installation Techniques: The CID Guide* redbook, second edition.

**Axel Buecker** is a LAN Systems Specialist in IBM Germany. He holds a degree in computer science from the University of Bremen. He has ten years experience in a variety of areas related to OS/2, including design and implementation of complex networks in large organizations, utilizing different aspects of OS/2, LAN Server, Warp Server environments, system management and software distribution.

**Scott Johnston** is a network systems engineer with the Global Services division of IBM Canada, Ltd. For four years, he has completed various projects related to the design, installation, and performance tuning of OS/2 LAN systems, with particular emphasis on remote access solutions. Recently, He has begun working on projects involving RS/6000, AIX, and the Netstation network computer. He holds a bachelor of science degree in mechanical engineering from the University of Saskatchewan.

**Larry D. Jones** is a Senior Systems Engineer with Innova Solutions, Inc. currently acting as a consultant to IBM's Personal Systems Solution Center located in Westlake, Texas. His current responsibilities with the Enterprise Server Integration Team include large-scale LAN/WAN design, troubleshooting, and tuning, advanced technical support for OS/2 & OS/2 Warp Server, and general consulting for IBM's customer base. He holds a master's degree in social psychology from University of Munich. He has worked with the OS/2 family of products for eight years, and has more than 20 years of technical design and field experience.

**Indran Naick** is a Senior IT Specialist in IBM South Africa. He holds a degree in computer science from the University of the Witwatersrand. He has eight years experience in a variety of areas related to OS/2, LAN Systems, and Systems Management. He is also a Certified NetWare Engineer and OS/2 Warp Server Engineer and has written extensively on OS/2 Warp Server and related products.

**Charles Rouh** is a Senior Marketing Support Representative in the Personal Solutions Systems Center located in Westlake, Texas. He is responsible for providing IBM Business Partners and Software Marketing Specialists with advanced technical support for IBM Warp Server and LAN Server products. He holds a bachelor degree in English literature from the University of Virginia. He has worked with OS/2 since 1988 and has over 28 years of field and headquarters experience in both marketing and technical organizations.

**Keiichi Togashi** is an Advisory IT Specialist in IBM Japan Systems Engineering Co., Ltd. He holds a bachelor of law degree from the University of Chiba. Before joining IBM, he worked as a programmer for six years. After joining IBM he has worked in the technical support office with OS/2, PC LAN and related products for the past eight years.

**Clive Williams** is a network consultant for IBM Global Services in Warwick, England. He provides technical advice and support on OS/2, LAN Server, PC and network products. He has been working with the OS/2 family of products for eight years. He holds a bachelor of science degree in biological sciences from the University of Birmingham.

Thanks to the following people for their invaluable contributions to this project:

Marcus Brewer  
International Technical Support Organization, Austin Center

Oscar Cepeda  
International Technical Support Organization, Austin Center

Alex Gregor  
International Technical Support Organization, Austin Center

Neil Stokes  
International Technical Support Organization, Austin Center

Uwe Zimmermann  
International Technical Support Organization, Austin Center

Marcus Brewer, Editor  
International Technical Support Organization, Austin Center

Ron Aguirre  
EMEA Education and Training, Austin

Glenn Brew  
PSP, Austin

Andy Erhenzeller  
PSP, Austin

Steve French  
PSP, Austin

Aidon Jennery  
PSP, Austin

Toshi Shimizu  
PSP, Austin

Larry Sullenger  
PSP, Austin

Robert Rose  
PSP, Austin

Dan Wiggins  
PSP, Austin

---

## Comments Welcome

### **Your comments are important to us!**

We want our redbooks to be as helpful as possible. Please send us your comments about this or other redbooks in one of the following ways:

- Fax the evaluation form found in "ITSO Redbook Evaluation" on page 345 to the fax number shown on the form.
- Use the electronic evaluation form found on the Redbooks Web sites:

For Internet users                      <http://www.redbooks.ibm.com>

For IBM Intranet users                <http://w3.itso.ibm.com>

- Send us a note at the following address:

[redbook@vnet.ibm.com](mailto:redbook@vnet.ibm.com)

---

## Chapter 1. Introduction to WorkSpace On-Demand

This chapter discusses the choices you have for implementing network computing along with the positioning of WorkSpace On-Demand as a network computing implementation in your operating environment. Also provided is an overview of the advantages you can expect in implementing WorkSpace On-Demand in your enterprise.

---

### 1.1 WorkSpace On-Demand Enters the Network Computing Arena

Network computing can be described as the use of different open technologies providing connectivity, ease-of-use, application functionality, information access, scalability and systems management across widely dispersed types of networks.

One of the main intentions of network computing is the any-to-any type of connectivity between applications without having to worry about the hardware or software platforms you are using. By making use of open standard technologies, network computing provides many benefits of the client/server paradigm, while avoiding its numerous shortcomings. How WorkSpace On-Demand fits into this area is covered in more detail in Section 1.2, "Let's Look at the Benefits of WorkSpace On-Demand" on page 5.

The network computing approach provides a three-tier solution. This three-tier environment is made up of:

- A client workstation

This handles the user interface and a minimal set of application functions.

- A Web server

The major application functions, some of the data processing and updating are performed at this level.

- A corporate central processor

Corporate data accessing and legacy processing are performed here.

The client workstations play an important role in a network computing environment because they provide the end user with an overall view of the complete solution. Let's take a brief look of the types of workstations available today.

### **1.1.1 Workstations Available Today**

In the early 1980s, as the first PCs came on to the market, people in the Information Systems (IS) industry thought that PCs might replace mainframe computers and cut operating costs drastically. Over the years, as Personal Computers gained more functionality and better user interfaces, end-users improved their productivity. While enterprise data and legacy applications were still placed on the more reliable mainframe platforms, there was more and more need for distributed access to these resources.

The IS industry succeeded in connecting the two worlds of PCs and mainframes by implementing a client/server model with distributed databases. Then with the evolution of multi-platform applications over a variety of networking infrastructures, it appeared that PCs were getting closer to replacing mainframe computers. However, as people in the IS industry realized the immense overall costs of this approach, the client/server model was reaching a deadlock. Even though a lot of data and programming logic was being distributed to PC servers, the Million Instructions Per Second (MIPS) power sold in mainframe computing was at its historical peak.

The goal of network computing is to provide functions for accessing legacy data and applications while reducing the overall costs of operating an enterprise-wide environment. This goal can be achieved only by the implementation of standards on all the platforms involved, such as TCP/IP for the networking protocol and 100% pure Java applications. This will lead to truly portable applications and solutions where in the network-computing environment, all devices are able to communicate with one another easily.

Looking towards a transition to network-computing, we can say for certain that the new network-computing devices are not going to replace the PC everywhere. Because different users have varying application needs, different technologies have to be employed to serve those needs. The advantage of network-computing technologies is that it lets you choose from a wider scope of workstations to fit your requirements and reduce the overall costs. Figure 1 on page 3 shows a range of applications with network components matched to the needs of the applications.



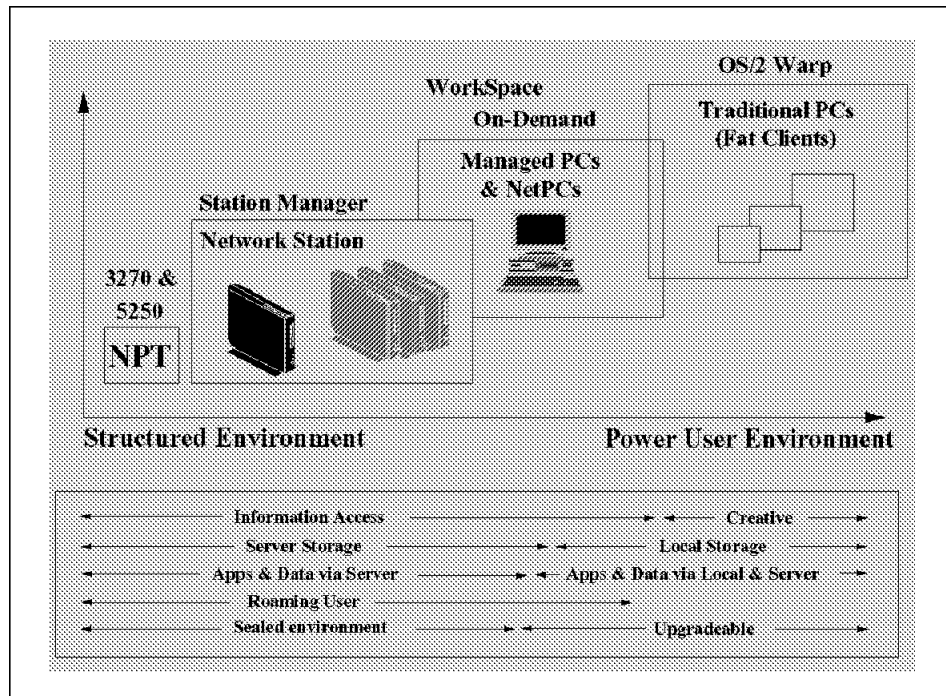


Figure 1. Network-Computing Client Offerings

The choice of a wider variety of workstations improves the enterprise's ability to make appropriate investments in the evolving network-computing marketplace. Following is a description of the various network client systems and some of their characteristics.

#### Non-Programmable Terminals

NPTs are often found in large enterprises connected to host-based applications systems. With the NPT, the user interface is managed and controlled by the central processing system. Historically, these terminals were the first to bring end-user access to information in the enterprise's central databases.

#### Network Computers

Based on RISC processors, NCs offer greater versatility than NPTs because they have the built-in capability to provide 5250 and 3270 emulation sessions, access to Java and Windows-based applications and a browser. While natively providing the Java virtual machine to run Java applications, the network computer needs a

dedicated Windows NT application server to provide support for 32-bit Windows applications. This means that the applications are running on a server, and only the presentation is performed on the network computer.

### **Managed PCs**

This type of Intel-based (or compatible) hardware platform offers you the ability to run new network-computing operating systems. You may already have a large number of Intel-based (or compatible) PCs installed in your enterprise that can be turned into managed PCs, or you can purchase new hardware specifically designed to be managed PCs. With WorkSpace On-Demand, for example, you'll be able to run the operating system and a set of applications on the client machine including:

- Java applications
- OS/2 applications
- DOS and Windows 3.x applications
- 32-bit Windows applications
- 5250 and 3270 emulator sessions
- Browser access

You can find more information about applications in Section 1.2.4, "Application Support" on page 7.

### **NetPCs**

Managed PCs and NetPCs are very similar. The major difference is that NetPCs have sealed cases and are not upgradeable, while Managed PCs have locked covers and can be upgraded.

### **Traditional PCs**

There is still demand for desktop and laptop PCs because they are designed to offer highly sophisticated end-user environments.

A laptop computer may be used by people who travel a lot or who work at various

locations. They need a full-function set of applications and access to data wherever they are, even when there is no network connection available. The installation of workgroup computing software and complete application suites requires a powerful machine with a lot of local capabilities.

A powerful desktop PC is still a basic requirement for people who, for instance, are developing software, administering networks or designing complex graphics. These people need full control of their desktop environment, as well as local resources such as hard disks and memory, in order to apply configuration changes and updates.

---

## **1.2 Let's Look at the Benefits of WorkSpace On-Demand**

There are several benefits in using a network-computing operating system like WorkSpace On-Demand, especially for large enterprises. Because WorkSpace On-Demand builds on LAN-based workstations, it inherits the benefits associated with maintaining this type of environment. These include:

- Ease of administration
- Security
- Cost of ownership
- Application support

### **1.2.1 Ease of Administration**

All client boot images are provided by the WorkSpace On-Demand server. This reduces the hassle associated with distributing operating software updates to large numbers of client workstations. Upgrading to a new software release in a WorkSpace On-Demand environment is as easy as getting a new boot image to the server; that image is then loaded automatically by all users when they start their workstations.

Designing and distributing user-specific desktops always has been a lot of work for administrators. Special implementations even needed third-party products for securing access and desktop arrangements. Additionally, users would play around with their desktops, adding to the administration and

maintenance workload. WorkSpace On-Demand lets the administrator design standard user desktops centrally and distribute them, while limiting the users' ability to change or alter them.

### **1.2.2 Security**

One of the main reasons for centralized management and access control is security. There are quite a few viewpoints on security, and it can rapidly expand to a large-scale project of its own. The security WorkSpace On-Demand provides consists of the following:

- Access control

From the WorkSpace On-Demand client, the user is able only to log on to the network or to shut down the machine. This prevents anybody who does not have an appropriate ID and password from gaining access to network data.

- Simplified Workplace Shell

Once the users have identified themselves, they get access only to a Simplified Workplace Shell. This shell provides access to applications and data directories that the user has been granted authority to work with by the system administrator. Users cannot accidentally gain access to items they are not authorized to use.

- Central data storing

All data is stored on central servers, and users at the client workstations cannot gain access to their local disk. This allows for controlled backup and recovery of critical data elements.

- Virus protection

Although using full-function PC applications, there is very good virus protection in the WorkSpace On-Demand environment. There is no diskette drive available to the end-user. Viruses cannot be introduced into the network unknowingly by the end-users at their workstations.

If there is the need for additional third-party security products, these can be implemented as they are in former versions of OS/2 Warp and OS/2 Warp Server.

### **1.2.3 Cost of Ownership**

WorkSpace On-Demand can offer an economical alternative to terminals and under-utilized personal computers, especially where there is the need to run native OS/2 applications. With WorkSpace On-Demand, you have the ability to access network applications, intranets and the Internet and centrally administer the environment, which lowers the total cost of ownership.

Additionally, being able to utilize existing PC hardware in the enterprise can reduce the up-front expenses.

Centralized administration can significantly reduce your expense and the time-consuming tasks of upgrading operating systems and reinstalling or reconfiguring user desktops.

Helpdesk functions become more effective with standardized, unchangeable desktops, and simple desktop layouts reduce the burden of end-user training.

There is a reduction in the hidden costs associated with end-users' unauthorized access to applications and tools for which they have not been trained.

Central data storage provides the administrator with the most efficient way of handling backup and restore functions.

#### **1.2.4 Application Support**

Compared with other available network-computing solutions, WorkSpace On-Demand provides the end-user with a broad range of application support today, including support for:

- *Java applications*

Java support is built directly into the operating system, thus providing a performance advantage.

- *OS/2 applications*

OS/2 applications are natively supported by WorkSpace On-Demand.

- *DOS and Windows 3.x applications*

WorkSpace On-Demand provides the same DOS and Windows 3.x support available in OS/2 Warp Version 4.

- *3270 and 5250 access*

Host access to S/390 or AS/400 mainframes is provided through native OS/2 applications such as the IBM Personal Communications family of products.

- *Intranet and Internet access*

There are many native OS/2 utilities available, including a browser to access all Intranet and Internet resources in the enterprise environment.

- *X-Windows applications*

You can implement a third-party X-Server application to use X-Windows applications residing on any UNIX-based server.

- *32-bit Windows applications*

To run 32-bit Windows applications, you can install a third-party multi-user Windows NT application server to serve your WorkSpace On-Demand clients.

With all this application support provided to users in the WorkSpace On-Demand environment, you are well positioned in the evolving network-computing arena.

Since WorkSpace On-Demand consists of a client operating system component and a set of server utilities working together as one, there are technical benefits you can expect to achieve. These will be discussed in more detail in Chapter 2.3, "WorkSpace On-Demand Components" on page 13.

---

### 1.3 Making Up Your Mind for WorkSpace On-Demand

When should a customer decide to implement WorkSpace On-Demand instead of a full OS/2 Warp Version 4 or an IBM Network Station implementation? There is no universal recommendation. The appropriate choice depends on the customer's requirements, priorities and existing applications.

For customers who understand the benefits of network-computing, who want to reduce their operational expenses, who need to protect current investments, or who want to begin the transition from client/server to network computing, WorkSpace On-Demand is the right solution.

WorkSpace On-Demand is the industry's first network-computing operating system to address the range of emerging Intel-based (or compatible) hardware environments as well as existing PC hardware and software environments. It helps lower the cost of administration by streamlining management processes. It is based on OS/2 Warp Server's proven technology and provides functions on which some of the largest enterprises in the world can run their business-critical applications.

---

## Chapter 2. What is WorkSpace On-Demand?

This chapter builds on the previous discussion of network computing in Chapter 1, "Introduction to WorkSpace On-Demand" on page 1 and describes how WorkSpace On-Demand fits into this environment. We discuss those end-user computing environments that are appropriate to migrate to WorkSpace On-Demand and network computing. We also examine WorkSpace On-Demand's components and discuss the functional and usability enhancements that WorkSpace On-Demand adds to Remote Initial Program Load (RIPL) technology.

---

### 2.1 WorkSpace On-Demand: An Overview

WorkSpace On-Demand is IBM's Intel-based (or compatible) network operating system that is optimized for network computing. It is a simple, economical software alternative that extends the use of your current PC investment and allows you to add new NetPC hardware while lowering your total cost of ownership. IBM's WorkSpace On-Demand allows you to access network applications, the intranet and the Internet using your current PC hardware and server software infrastructure while protecting your investment in existing applications.

With WorkSpace On-Demand, you have centralized installation and management of applications at the server. Your up-front expenses for installing clients is reduced, and your ongoing costs to maintain client applications, manage upgrades, and provide end-user support are significantly lessened. End-users and administrators benefit from improved software management and maintenance without losing the benefits that a PC can provide, such as:

- Executing your applications on a local Intel x86 or compatible processor
- Local disk access for swapping
- Support for multiple protocols and languages
- DOS, OS/2, Java, and Windows 3x application compatibility

WorkSpace On-Demand bridges the evolution to network computing for organizations that want to realize the benefits of centralized, server-managed clients without losing the flexibility and power offered by traditional PCs. Users can access both intranet and Internet applications and information as well as conventional client/server and in-house software.

---

## 2.2 WorkSpace On-Demand and End-User Computing Environments

Today's end-user computing environment spans a broad spectrum ranging from stand-alone personal computers to interconnected, networked workstations. This section's discussion is limited to the networked environments that are appropriate for migrating to WorkSpace On-Demand and the network computing end-user environment.

One effective way of categorizing end-user computing environments is by the computing paradigm on which they're based. As a general rule, these environments range from the relatively simple, characterized by a lower cost of ownership, to the extremely complex, with their correspondingly higher cost of ownership. The major contributing factor to the cost of ownership calculation is that of systems management and maintenance.

For example, at one end of the spectrum is the Host-Centric paradigm based on non-programable terminals connected to a central computing facility. In this environment, application availability is limited to those applications provided by a centralized Information Systems staff; management is centralized, and computing costs are centrally managed. This type of end-user computing environment is often referred to as *fixed-function* or *green screen*. WorkSpace On-Demand provides an effective way to address the needs of end-users in this environment and enables the additional capabilities and benefits associated with the network computing environment.

The other end of the end-user computing spectrum is based on the client/server computing paradigm and is often referred to as a *full-function PC*, or a *Fat Client*. End-users in this environment obtain file, print and/or application services through a network connection from a server running some type of network operating system. A Fat Client is also characterized by localized desktop processing power, a multitasking (usually with a graphical user interface) desktop operating system, local code and data storage on a hard disk, and a wide variety of application software that is available as shrink-wrapped applications in addition to those applications available from the network. In this environment, cost of ownership is typically high due to the the increased complexity of system administration, maintenance, and support of complex hardware and software in a networked environment.

WorkSpace On-Demand technology provides support for a wide subset of user requirements that lie clearly between the fixed-function and full-function PCs. WorkSpace On-Demand is the IBM network-computing solution that addresses the range of emerging Intel-based (or compatible)



hardware environments including NetPCs, Network PCs, Managed PCs, and Thin Clients.

## **2.2.1 End-User Environments Supported by WorkSpace On-Demand**

It is important to understand that even though WorkSpace On-Demand supports a wide spectrum of end-user environments, it is designed to coexist with, rather than replace, other end-user computing environments that may exist within any given user community. Some end-user computing environments are good candidates for migration to the WorkSpace On-Demand computing environment, while others are not. What follows is an examination of these environments that explains how you can benefit from the advantages that WorkSpace On-Demand has to offer.

### **2.2.1.1 Requirements Suitable for Migration to WorkSpace On-Demand**

Clearly, for those end-users who want to gain the benefits available from making the transition to network computing, WorkSpace On-Demand is the right solution. However, WorkSpace On-Demand may also be an appropriate solution for end-users who have any of the following requirements:

- Access to mainframe and/or mid-range applications: WorkSpace On-Demand provides end-users with support for 3270 and 5250 emulation.
- Access to the corporate intranet, extranet or the Internet: WorkSpace On-Demand provides end-users with a Web browser.
- Access to legacy applications: WorkSpace On-Demand provides end-users with access to OS/2, Windows 3.X and DOS applications.
- Access to Lotus Notes: WorkSpace On-Demand allows for the setting up of Lotus Notes as a public application for use by the end user.
- Transition from client/server to network computing: WorkSpace On-Demand provides the essential elements of network computing that include:
  - Full TCP/IP support including DHCP and DDNS
  - Support for native Java-based applications
  - An optional browser-based user interface
  - An optional Lotus Notes-based user interface that supports a Lotus Notes client, including support for Domino and Kona applications
  - Support for user-written or third-party user interface shells
- 32-bit Windows applications
- X-Windows applications

**Note**

Support for 32-bit Windows and X-Windows applications requires additional hardware and software in order to be supported by WorkSpace On-Demand.

## **2.2.2 End-User Environments Not Appropriate for WorkSpace On-Demand**

WorkSpace On-Demand does not provide for any local storage, other than a swapper file, on the hard disk. All application code as well as system code is delivered from the WorkSpace On-Demand server to the WorkSpace On-Demand client through the network. What follows are some general guidelines for those types of end-user computing environments that may not be appropriate for migration to WorkSpace On-Demand.

### **2.2.2.1 Environments That May Not Be Suitable for Migration to WorkSpace On-Demand**

As a general rule-of-thumb, the end-user requirements that are not as well-suited for migration to WorkSpace On-Demand tend to fall towards the *full-function* or *Fat Client* end of the end-user computing spectrum. Reasons not to migrate to WorkSpace On-Demand would include, but are not limited to, the need to support any of the following requirements:

- Environments that may not be suitable:
  - Applications that require local storage
  - Environments where the server cannot be located on the same LAN segment as the client
  - Large/complex applications that dynamically load a significant number of DLLs during execution
  - Environments where the end user spends a significant percentage of the time using traditional fat client (Windows 95 and Windows NT) applications
- Environments that will require additional planning:
  - Applications that require transport protocols other than NetBIOS/802.2 or TCP/IP
  - Environments with specialized hardware/peripheral devices that are not supported, for instance ultra-high-resolution video adapters, badge readers, scanners, or devices requiring proprietary or non-standard interfaces

## 2.3 WorkSpace On-Demand Components

Now that we have discussed those end-user computing environments that can potentially benefit from using the WorkSpace On-Demand and migrating to the network computing environment, we should spend some time discussing the architecture and components that comprise WorkSpace On-Demand.

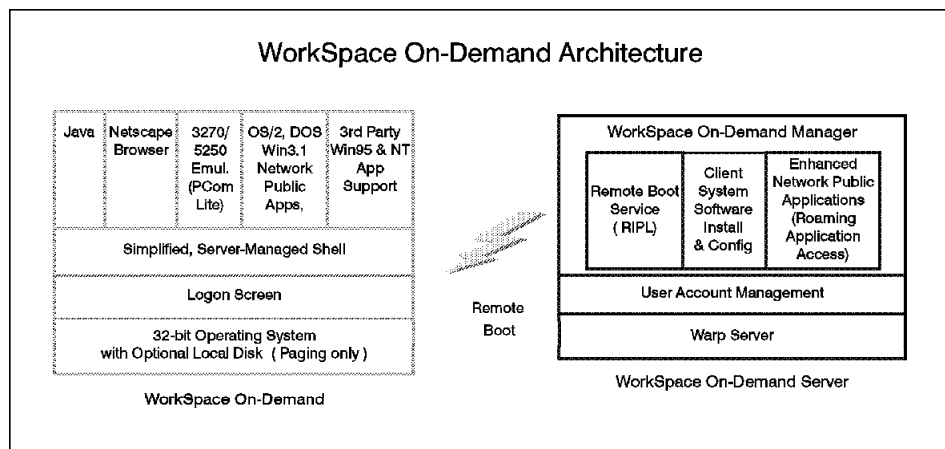


Figure 2. WorkSpace On-Demand Components

Figure 2 shows an overview of the WorkSpace On-Demand components. WorkSpace On-Demand technology consists of two components:

1. The client component
2. The manager component

Both of these components are physically installed on the WorkSpace On-Demand server.

### 2.3.1 The WorkSpace On-Demand client Component

The WorkSpace On-Demand client is the client operating system and software that runs on a Network PC or Managed PC. This component is downloaded across the LAN from the WorkSpace On-Demand server to the WorkSpace On-Demand client workstation. The WorkSpace On-Demand client workstation can be any of the following machine types:

- Network PC, for example the IBM PC 300GL
- Managed PC

#### Note

"Managed PC" is defined as an Intel-based or compatible PC that loads all of its software from a server.

See Chapter 4, "WorkSpace On-Demand Planning" on page 51, for a complete discussion of supported hardware.

### 2.3.1.1 The WorkSpace On-Demand Shell

The WorkSpace On-Demand's user interface is different than the Workplace Shell that is available on other OS/2 clients. WorkSpace On-Demand provides a simplified shell with a logon screen. This can help reduce end-user training costs and improve manageability. The WorkSpace On-Demand shell is a *restricted* shell in the sense that it is populated by only three system icons, other than the icons representing the user-unique, network-public applications that can be added by the system administrator.

WorkSpace On-Demand System Icons:

- Refresh the Desktop
  - Updates the user interface without performing a logoff/reboot
- Logoff the User ID
  - Logs off the current user ID but maintains the network connection between the WorkSpace On-Demand client and the WorkSpace On-Demand server
- Shutdown the Workstation
  - Closes the network connection between the WorkSpace On-Demand client and the WorkSpace On-Demand server

### 2.3.1.2 Replacing the WorkSpace On-Demand Shell with Another Interface

The WorkSpace On-Demand shell can be replaced by another user interface, for example a Web browser, by modifying the RUNWORKPLACE statement in the CONFIG.SYS files. For example:

```
rem SET RUNWORKPLACE=Z:\OS2\PMLOGON.EXE.  
SET RUNWORKPLACE=Z:\NETSCAPE\NETSCAPE.EXE
```

#### **ATTENTION!!**

If you replace PMLOGON.EXE with *ANY* other user interface, be aware that you will lose the logon security provided by PMLOGON.EXE. In addition, replacing PMLOGON.EXE has other implications such as removing programming hooks to the USER FIT file and the application launcher.

#### **2.3.1.3 Restricted Local Disk Access**

If a WorkSpace On-Demand client workstation contains a local hard disk, access to the disk is restricted so that end-users cannot write data to the disk. If a hard disk is present, it is used only for writing to the SWAPPER.DAT file as a *local* swapping device. Providing a *local* swapping device can improve system performance. Restricting the local disk access is accomplished by adding the PROTDISK.SYS device driver statement to the CONFIG.SYS file. For example:

```
DEVICE=Z:\OS2\BOOT\PROTDISK.SYS
```

#### **2.3.1.4 Application Roaming**

Application roaming is the ability to locate and successfully execute personal user files and user-unique, network-public applications from any workstation from which the user is logged on. No longer must users run their applications from their own, custom-configured workstations in order to insure a successful result.

Application roaming is accomplished by implementing a two-stage initialization process.

1. The first stage provides a hardware affinity.

Stage 1 uses the WorkSpace On-Demand client machine's network hardware address to associate a machine's hardware image with a specific workstation. This software is loaded across the network from the WorkSpace On-Demand server to the WorkSpace On-Demand client workstation.

2. The second stage provides a user affinity.

Stage 2 employs the user ID to specify a user's unique applications and the environment required to support them. This environment includes the PATH, LIBPATH and DPATH statements as well as the environmental variables required to execute those network-public applications to which the end-user has access. This information is dynamically loaded from the WorkSpace On-Demand server to any WorkSpace On-Demand client from which the user logs on.

### 2.3.2 The WorkSpace On-Demand Manager Component

The WorkSpace On-Demand Manager component is the set of server utilities used to install, configure and maintain the network client hardware and software. A OS/2 Warp Server server with the WorkSpace On-Demand Manager installed is referred to as a WorkSpace On-Demand server. All operating system, WorkSpace On-Demand client and WorkSpace On-Demand Manager software resides on the server.

Installing this software on the server, instead of the client(s), greatly simplifies system management and reduces the total cost of ownership. The WorkSpace On-Demand Manager component is installed on a suitable server hardware platform in addition to OS/2 Warp Server with both RIPL Support and File and Print Sharing enabled. The WorkSpace On-Demand server consists of the following:

- OS/2 Warp Server (prerequisite)
  - File and Print Sharing (prerequisite)
  - RIPL Support (prerequisite)
- WorkSpace On-Demand Manager software including:
  - Client Configuration/Administration Utilities
  - RIPL Service Enhancements
  - Client Application Management

#### 2.3.2.1 Heterogeneous Client Support

As was discussed in 2.2.1, “End-User Environments Supported by WorkSpace On-Demand” on page 11, WorkSpace On-Demand is designed to coexist with other end-user computing solutions that may exist in your environment. Therefore, it is possible to configure a single OS/2 Warp Server to provide support for a combination of traditional Fat Clients and DOS or OS/2 RIPL clients in addition to WorkSpace On-Demand clients. However, you should consider the performance and capacity requirements of your particular situation before determining whether or not to configure your WorkSpace On-Demand server(s) for heterogeneous client support.

#### 2.3.2.2 Multi-Function Server Support

It is also possible to combine the WorkSpace On-Demand Manager function with other server functions such as the domain controller, backup domain controller or an additional server to create a *multi-function* server if appropriate, for example in a small branch office. Here again, it is important that you consider capacity and performance implications before implementing multi-function server support.

## 2.4 RIPL Functional and Usability Enhancements

RIPL technology is a fundamental part of the initial release of WorkSpace On-Demand. However, there are several enhancements to RIPL technology that are incorporated into WorkSpace On-Demand that cause it to differ from the standard RIPL support provided by OS/2.

Table 1 summarizes the functional and usability enhancements added to RIPL support in the WorkSpace On-Demand product.

<i>Table 1. WorkSpace On-Demand RIPL Functional and Usability Enhancements</i>	
<b>Today's OS/2 RIPL</b>	<b>WorkSpace On-Demand</b>
One, generic, machine class <ul style="list-style-type: none"><li>• One, generic, HW image</li></ul>	Support for most 486 66 MHz + Intel or compatible machines <ul style="list-style-type: none"><li>• Generic HW Image (supported by today's OS/2 Warp)</li><li>• Selected PCCo Systems (Networked PCs and NetPCs)</li><li>• Selected OEM systems</li></ul>
Selectable LAN Support <ul style="list-style-type: none"><li>• RIPL Support</li></ul>	Selectable LAN Support <ul style="list-style-type: none"><li>• RIPL Support</li><li>• BOOT-P/DHCP support (future release)</li></ul>
VGA Video (640x480x16)	SVGA Video with selected machines <ul style="list-style-type: none"><li>• Selectable resolution</li><li>• Updated VGA support (640x480x256)</li></ul>
No Local Printer Support	Local Printer Support in HW Image
No audio	Business-Audio

### Note

While it has always been possible to manually configure capabilities such as video resolutions higher than VGA and the ability to print from locally-attached printers on previous RIPL clients, this is a complex task, and it must be done on a machine-by-machine basis. WorkSpace On-Demand automates this process and enables the administrator to specify these capabilities in the machine-class hardware image that is downloaded to the appropriate WorkSpace On-Demand clients at boot time.

See Chapter 3, "RIPL and WorkSpace On-Demand" on page 19, for a complete discussion of the RIPL process as implemented in WorkSpace On-Demand.

---

## 2.5 The Use of RIPL vs. BOOT-P Technology in WorkSpace On-Demand

WorkSpace On-Demand enables you to exploit, or make a transition to, the network computing environment. The network computing model is based on Internet technologies, which implies the use of routable, wide area network protocols, specifically TCP/IP. One could ask the question, "Why does WorkSpace On-Demand utilize RIPL on NetBIOS instead of BOOT-P on TCP/IP?"

The answer to this hypothetical question is that support for BOOT-P will be available in a future release of WorkSpace On-Demand. IBM's published Statement of General Direction for WorkSpace On-Demand states that, over time, it intends to include support for booting clients over a TCP/IP network as well as providing additional enhancements to the product.

A detailed discussion of RIPL can be found in Chapter 3, "RIPL and WorkSpace On-Demand" on page 19.



---

## Chapter 3. RIPL and WorkSpace On-Demand

This chapter contains an introduction to Remote Initial Program Load (RIPL) and its implementation in WorkSpace On-Demand. It is assumed that the reader is familiar with OS/2 Warp Server. Knowledge of Remote IPL is not assumed.

OS/2 Warp Server supports DOS and OS/2 Remote IPL. This chapter focuses primarily on the introduction of WorkSpace On-Demand. For more information on pure OS/2 and DOS environments, refer to the redbook titled *Inside OS/2 LAN Server 4.0*, SG24-4428.

Remote Initial Program Load is the process of loading an operating system onto a workstation from a location that is remote to the workstation.

The RPL protocol was co-developed by 3Com, Microsoft and IBM. It is used today with IBM OS/2 Warp Server, DEC Pathworks, and Windows NT.

Two other commonly used Remote IPL protocols are:

1. Novell NCP (NetWare Core Protocol).
2. BOOT-P, an IEEE standard, used with UNIX and TCP/IP networks.

There are a number of differences between the various remote boot protocols. At this time, only RPL is supported with the WorkSpace On-Demand product.

---

### 3.1 RIPL Overview

RIPL is achieved using a combination of hardware and software. The requesting device, called the requester or workstation, starts up by asking the loading device to send it a bootstrap program. The loading device is another computer that has a hard disk and is called the RIPL server or file server. The RIPL server uses a loader program to send the bootstrap program to the workstation. Once the workstation receives the bootstrap program, it is then equipped to request an operating system, which in turn can request and use application programs.

The software implementations differ between vendors, but theoretically, they all perform similar functions and go through a similar process.

### 3.1.1 RIPL Hardware

The RIPL workstation requires a special ROM installed on its LAN adapter. This ROM is also known as a Boot ROM or RPL Module.

#### Note

The RPL Module contains the initial code to begin the process. In order for a network adapter to work with a WorkSpace On-Demand server, the adapter needs to support the RPL protocol on the RIPL chip. An adapter that only supports NCP or BOOT-P will not work.

With OS/2 Warp Server, a bootable Remote IPL diskette can also be created for selected network adapters. The boot diskette initializes the adapter and starts the remote boot process. This diskette could be used if the network adapter card does not have a RIPL chip installed. MKRDPM is a Presentation Manager (PM) based utility supplied with OS/2 Warp Server and can be used to create diskettes for PC Network, token-ring, and Ethernet networks.

#### 3.1.1.1 How Does the Boot ROM Work?

When the workstation starts up the RIPL ROM code on the network adapter is given control. The RIPL ROM code initializes the adapter and starts sending special IEEE 802.2 FIND frames that are recognized by RIPL servers.

The server(s) respond with a FOUND frame which provides the RIPL ROM with the network address of the server. The RIPL ROM then sends a SEND.FILE.REQUEST frame to the server and goes into receive mode waiting for data frames.

The server builds a boot block that contains the boot code and sends the boot block data to the RIPL ROM using FILE.DATA.RESPONSE frames. The last FILE.DATA.RESPONSE frame contains the memory address that the RIPL ROM is to pass control to when it completes receiving the data.

After the RIPL ROM on the adapter card had received the boot block then the Boot ROM emulates a floppy drive. It takes over the floppy drive interrupt (INT 13h). As far as the workstation is concerned, it then has an "A" drive with a write-protected bootable disk in it.

Since the workstation thinks it has a floppy drive, it requires all of the low-level data normally contained on a floppy disk. This includes the system sectors, FAT table and directory tables. The Boot ROM obtains this information from a boot image file created on the server.

The diskette image consists of a CONFIG.SYS and the necessary device drivers that are required for the desired configuration. The boot image file is an exact image of the floppy that the workstation believes is in drive A.

When the workstation starts up and issues a read request, the Boot ROM intercepts the request and converts it into a network read request. Instead of reading data from the floppy, the data comes from the boot image file.

#### **3.1.1.2 Multiple Adapters**

Multiple adapters with the RPL feature may be installed in one computer and many adapters support the switching technique described in this section. However, the OS/2 Warp Server RIPL implementation does not support multiple RIPL adapters in the RIPL client. Only the primary adapter can be used for RIPL. Thus, only the primary adapter can be used for RIPL of WorkSpace On-Demand clients from OS/2 Warp Server.

The order in which the adapters take control is as follows:

1. If two adapters have the RPL feature with an engineering change (EC) level of A33861C or higher, the primary adapter will take control first. If the loading device does not respond to a requesting device within two minutes, the primary adapter passes control to the RPL feature on the alternate adapter. Control continues to be passed between the adapters at two minute intervals until a loading device responds or until an error is detected.
2. If one of the RPL features does not support being used in a multiple-adapter environment, the adapter with the highest ROM address will take control.
3. LANStreamer Adapters can support up to seven RPL adapters. If one adapter fails RPL, the next RPL adapter will be tried. The process continues until a loading device responds or until an error is detected. The adapter with the highest memory range in the memory map gains control first after power on. Control is passed only between IBM LANStreamer MC 32 Adapters or IBM LANStreamer MC 16 Adapters.

The RIPL server requires a network adapter supported by the server software. This network adapter does not have to have any special additional hardware such as a RIPL chip.

### **3.1.2 Understanding RIPL**

For the RIPL function to be operational on a network, the network must have a RIPL server and one or more workstations with the necessary Boot ROM module on its adapters. The RIPL server and the workstation do not need to

be on the same LAN segment. They can be on different LAN segments connected with bridges.

Both the RIPL server and workstations must be on the same logical network ID. That is, a workstation can operate across a bridge, but it cannot operate across a router.

In Figure 3 on page 23, we have two devices on a simple network. The purpose of the two devices are:

- OS/2 Warp Server (Loading Device)

The server must have a fixed disk. Its purpose is to supply files to the workstations. Also, both the RIPL server and the workstations must support the same network type. For example, token-ring or Ethernet.

- WorkSpace On-Demand (Requesting Device)

The workstation does not need a fixed disk or diskette drive. It receives all the data it needs from the server. It does need a network adapter with a Boot ROM installed.

The following section describes the sequence of events that occur during a remote program loading operation. To keep the scenario simple, we are only describing the major steps between the two devices.

Figure 3 on page 23 shows the communication flow as well as the key control files.

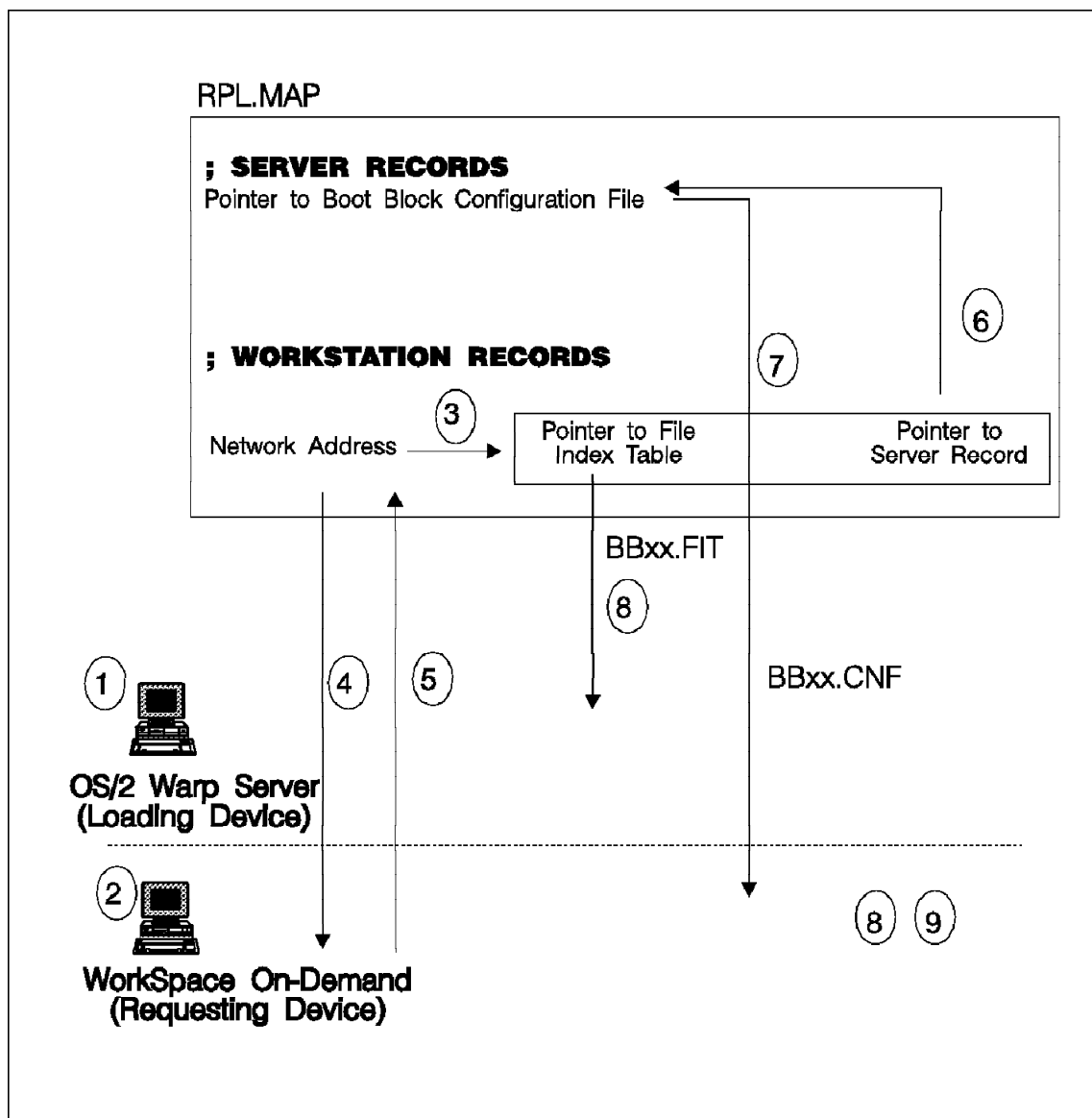


Figure 3. RIPL Process and Control Files

The steps in the Remote IPL process are:

1. The server (loading device) is powered on and attaches to the network ring. A loading program is started and waits for a FIND frame from the requesting device. The loading program is started by issuing the NET START REMOTEBOOT command on OS/2 Warp Server.

2. The WorkSpace On-Demand workstation is powered on, and the RPL feature (on the network adapter) obtains control. The RPL feature attaches the workstation to the network ring and sends a FIND frame. FIND frames contain the LAN adapter address and are repeated periodically until a server responds.
3. The server receives the FIND frame sent by the requesting device. The server then checks the RPL.MAP file for a workstation record entry that matches the network address contained in the FIND frame.  
  
The workstation record contains two pointers. The first pointer is to a server record entry, contained in the RPL.MAP file, and the second pointer is to the File Index Table (FIT) file.
4. If an entry exists, the server responds by sending a FOUND frame. The server provides its address in this frame so the workstation can identify it.
5. The workstation receives the FOUND frame with the address of the server and sends a SEND.FILE.REQUEST frame back. This frame is a request for the server to send the bootstrap program.
6. The bootstrap program information is contained in the Boot Block Configuration File (.CNF) on the server. This file contains the data to be sent in the FILE.DATA.RESPONSE frames used to start the initial boot of the workstation. The CNF file used is derived from the server record entry in the RPL.MAP file.
7. Upon receiving the SEND.FILE.REQUEST frame, the server sends the bootstrap program to the workstation by transmitting FILE.DATA.RESPONSE frames.  
  
The FILE.DATA.RESPONSE frames contain the boot block information. The boot block contains RIPL control data, the mini-file system driver, the FIT file, and the Loader.
8. The requesting device receives the bootstrap program and places it into memory. In most cases, more than one frame is required to send the complete bootstrap program. After the last frame of the program is received by the requesting device, control is transferred from the RPL feature to the bootstrap program.

After the boot block is loaded, control is passed to RPLBOOT.SYS. RPLBOOT.SYS hooks interrupt 21h to handle file I/O requests and moves the DOS drivers to high real memory (just below the 640 KB boundary) and initializes them. It then transfers control to the appropriate loader, OS2LDR, for WorkSpace On-Demand.

The workstation's FIT file and another file called MFSD20.SYS, the mini-File System Driver, are passed to the OS2LDR. The File Index

Table is used to map the standard file calls to the redirected file location on the server.

The OS/2 kernel and the base device drivers start loading. Up to this point, the DOS drivers and the mini-file system driver are still used for file redirection.

9. When the NETWKSTA.200 driver is loaded, file redirection is switched to using the OS/2 network drivers and LAN Server file redirection. WorkSpace On-Demand continues the boot from that point on as a standard OS/2 client. The default startup on a WorkSpace On-Demand client brings you up to a logon panel.

---

## 3.2 WorkSpace On-Demand RIPL Implementation

The Remote IPL implementation in WorkSpace On-Demand is an extension of the service already available in OS/2 Warp Server. The Remote IPL service in OS/2 Warp Server provides support for the Remote IPL of DOS and the following versions of OS/2:

- OS/2 2.0
- OS/2 2.0+Service Pak or Service Pak 2
- OS/2 2.1
- OS/2 2.1+Service Pak
- OS/2 2.1 for Windows
- OS/2 2.11
- OS/2 Warp (all versions)

### Note

OS/2 Warp Server CSD IP08265 contains the RIPL support for OS/2 Warp Version 4 for the non-DBCS countries.

Enhancements made to the Remote IPL service in WorkSpace On-Demand are listed in Table 1 on page 17.

### 3.2.1 The Remote IPL Control Files

As described in Section 3.1.2, “Understanding RIPL” on page 21, the Remote IPL process is controlled by four key files. In order to comprehend the process, it is important to understand the control files, their contents and their relationships. The four control files are:

1. RPL.MAP

2. The Boot Block Definition files (CNF)
3. The File Index Table (FIT) files
4. The NDISDD.PRO file

Each server contains only one RPL.MAP and NDISDD.PRO file. There are multiple CNF and FIT files. Each workstation has a unique FIT file.

#### **3.2.1.1 The RPL.MAP Control File**

OS/2 Warp Server contains a single RPL.MAP file for DOS, OS/2 and WorkSpace On-Demand, which is located in the \IBMLAN\RPL directory.

This file contains requester and server records that define the behavior of the Remote IPL service for each requester. Each requester must have a set of server and requester records residing on the Remote IPL server before Remote IPL can occur.

Each record in the RPL.MAP file takes up a single line and consists of several fields separated by space characters. Two types of records are defined in RPL.MAP: server records and workstation records. Figure 7 on page 29 shows part of the default RPL.MAP file.

A server record defines how the Remote IPL service functions when starting a specific operating system. Default server records are supplied in the RPL.MAP file when the remote boot service is installed. There is no GUI administration utility to update server records in this file. Workstations booting the same operating system and utilizing the same network adapter type use the same server record.

Requester records specify parameters used when starting a specific requester. Each requester record has a unique adapter ID field, which is used to match a specific machine on the network to that particular record. Requester records are created when Remote IPL requesters are defined on the server. Requester records can be defined either through the command line interface or the GUI. It is best to administer requester records through the LAN Server Administration GUI. Each requester record has a unique entry in the RPL.MAP file.



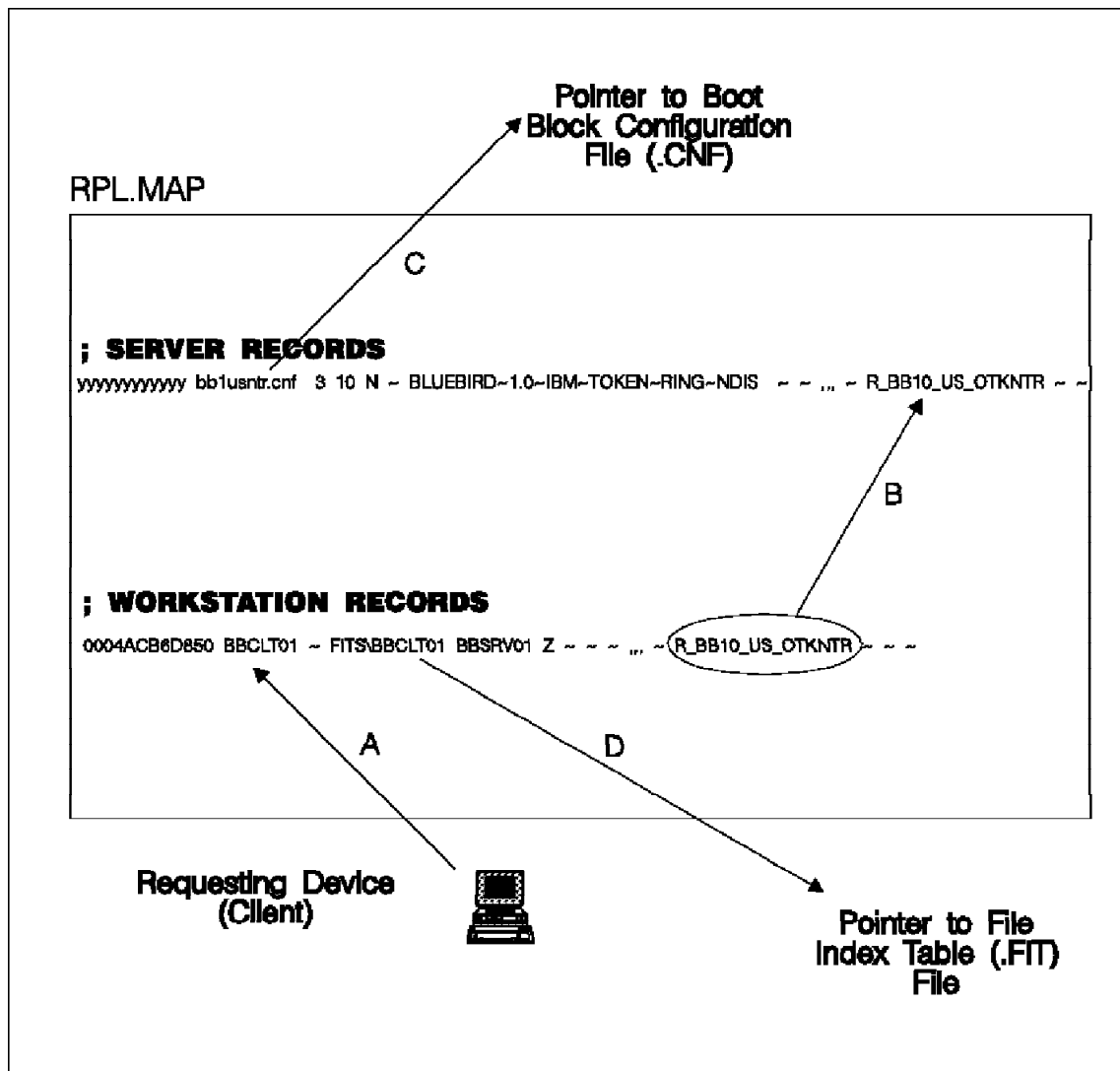


Figure 4. RPL.MAP File

Figure 4 shows an RPL.MAP file with just one workstation and one server record.

When the workstation is powered on, the LAN Adapter sends out a FIND frame containing the network address. In the example in Figure 4, this address would be `0004ACB6D850`. The server picks up the FIND frame and searches the RPL.MAP. In the example shown in Figure 4, the **(A)** indicates the location of the workstation record in the RPL.MAP file by the server for the network address, `0004ACB6D850`.

```
0004ACB6D850 BBCLT01 ~ FITS\BBCLT01 BBSRV01 Z ~ ~ ~ ,,, ~ R_BB10_US_OTKNTR ~ ~ ~
```

Figure 5. RPL.MAP Workstation Record with BBCLT01 Definition Using an IBM Token-Ring Adapter

The entry in the RPL.MAP file contains pointers to two related items. The first being the FIT filename (*BBCLT01* in the example indicated by **(D)**) and the second being the pointer into the server record section of the RPL.MAP file (*R\_BB10\_OTKNTR* in the example, indicated by **(B)**).

Using the workstation record, shown in Figure 5, the server locates the server record to use to start the initial boot phase. It locates the server record based on the value in the 12th field, in the example *R\_BB10\_OTKNTR*. Only one record in the server record section should match this field. The server record in our example is shown in Figure 6.

```
yyyyyyyyyyyyy bb1usntr.cnf 3 10 N ~ Workspace~ On-Demand~ 1.0~ IBM~ TOKEN~ RING~ COMPATIBLE~ NDIS ~ ~ ~ ,,, ~ R_BB10_US_OTKNTR ~ ~
```

Figure 6. RPL.MAP Server Record for an IBM Token-Ring Adapter

From the server record, the server locates the boot configuration file for the particular workstation. The boot configuration file is *BB10USTR.CNF* in our example.

#### NOTE

A workstation may have more than one entry in the RPL.MAP file. However, only one record may be active at any time.

The corresponding CNF file which is named in the server record is read. The linkage from the server record to the CNF file is shown by **(C)** in Figure 4 on page 27. The contents of the CNF file, the FIT file and the appropriate workstation record are downloaded to the workstation to begin the boot process.

Using the FIT file entry contained in the requester record in the RPL.MAP file, the requester remaps file requests until the workstation completes the boot.

Figure 7 on page 29 shows the RPL.MAP file once Workspace On-Demand has been installed. Not all the server record entries are shown in this diagram.

```

; server record fields:
; 1 = yyyyyyyyyyy
; 2 = boot block configuration file (.cnf)
; 3 = number of retries before default boot
; 4 = time window for retries (in seconds)
; 5 = acknowledge (A,N)
; 6 = loader parameters (~ for os2, image share name for dos)
; 7 = descriptive comment
; 8,9 = ~
; A = ...,
; B = ~
; C = workstation type; first letter is always R
; D,E = ~

; server records
;yyyyyyyyyyyy os220et.cnf 3 10 N ~ OS2~ 2.0~ IBM~ ETHERNET ~ ~ ~ R_20_OET ~ ~
;yyyyyyyyyyyy os220pc.cnf 3 10 N ~ OS2~ 2.0~ PCNET ~ ~ ~ R_20_OPC ~ ~
;yyyyyyyyyyyy os240ls.cnf 3 10 N ~ OS2~ WARP~ 4.0~ IBM~ LANSTREAMER~ MC~ 32~ TOKEN~ RING ~ ~ ~ R_240_OTKLS ~ ~
;
;
;
;yyyyyyyyyyyy pcinit.cnf 3 10 N PCNET INTERNAL~ USE~ ONLY ~ ~ ~ R_PCINIT~ ~
;yyyyyyyyyyyy bb1usntr.cnf 3 10 N ~ WorkSpace~ On-Demand~ 1.0~ IBM~ TOKEN~ RING~ COMPATIBLE~ NDIS ~ ~ ~ R_BB10_US_OTKNTR ~ ~
;yyyyyyyyyyyy bb1usej.cnf 3 10 N ~ WorkSpace~ On-Demand~ 1.0~ IBM~ ETHERJET~ ETHERNET ~ ~ ~ R_BB10_US_OETEJ ~ ~
;yyyyyyyyyyyy bb1uscla.cnf 3 10 N ~ WorkSpace~ On-Demand~ 1.0~ CRYSTAL~ SEMI~ ETHERNET ~ ~ ~ R_BB10_US_OETCLA ~ ~
;yyyyyyyyyyyy bb1use10.cnf 3 10 N ~ WorkSpace~ On-Demand~ 1.0~ INTEL~ 10/100~ ETHERNET ~ ~ ~ R_BB10_US_OETE10 ~ ~
;yyyyyyyyyyyy bb1us164.cnf 3 10 N ~ WorkSpace~ On-Demand~ 1.0~ IBM~ TOKEN~ RING~ NETWK~ 16/4~ ADT~ II ~ ~ ~ R_BB10_US_OTK164 ~ ~
;yyyyyyyyyyyy bb1usaae.cnf 3 10 N ~ WorkSpace~ On-Demand~ 1.0~ IBM~ LAN~ ADAPTER/A~ FOR~ ETHERNET ~ ~ ~ R_BB10_US_OETAAE ~ ~
;yyyyyyyyyyyy bb1uslae.cnf 3 10 N ~ WorkSpace~ On-Demand~ 1.0~ IBM~ LAN~ ADAPTER~ FOR~ ETHERNET ~ ~ ~ R_BB10_US_OETLAE ~ ~
;yyyyyyyyyyyy bb1uset.cnf 3 10 N ~ WorkSpace~ On-Demand~ 1.0~ IBM~ ETHERNET ~ ~ ~ R_BB10_US_OET ~ ~
;yyyyyyyyyyyy bb1usls.cnf 3 10 N ~ WorkSpace~ On-Demand~ 1.0~ IBM~ LANSTREAMER~ MC~ 32~ TOKEN~ RING~ ~ ~ ~ R_BB10_US_OTKLS ~ ~
;yyyyyyyyyyyy bb1us316.cnf 3 10 N ~ WorkSpace~ On-Demand~ 1.0~ 3COM~ ETHERLINK~ 16~ (3C507) ~ ~ ~ R_BB10_US_OET316 ~ ~
;yyyyyyyyyyyy bb1us3e3.cnf 3 10 N ~ WorkSpace~ On-Demand~ 1.0~ 3COM~ ETHERLINK~ III~ (3C509) ~ ~ ~ R_BB10_US_OET3E3 ~ ~
;yyyyyyyyyyyy bb1us3ei.cnf 3 10 N ~ WorkSpace~ On-Demand~ 1.0~ 3COM~ ETHERLINK~ II ~ ~ ~ R_BB10_US_OET3EI ~ ~
;yyyyyyyyyyyy bb1us3em.cnf 3 10 N ~ WorkSpace~ On-Demand~ 1.0~ 3COM~ ETHERLINK~ MC ~ ~ ~ R_BB10_US_OET3EM ~ ~
;yyyyyyyyyyyy bb1uswdp.cnf 3 10 N ~ WorkSpace~ On-Demand~ 1.0~ SMC~ ETHERCARD~ PLUS~ ELITE16 ~ ~ ~ R_BB10_US_OETWDP ~ ~

; workstation record fields:
; 1 = adapter id (12 hex digits)
; 2 = workstation name
; 3 = ~
; 4 = image file for dos (.img), fit file for os2 (.fit)
; 5 = name of rpl server
; 6 = boot drive for OS2, domain name for DOS
; 7,8,9 = parameters for device drivers 1,2,3
; A = additional memory for device drivers 1,2,3. Default: ...,
; B = ~ for os2, Z for dos
; C = workstation type; first letter is R -> enabled, D -> disabled
; D = ~
; E = volumeid string for dos, IML image file for os2
; F = P for OS/2 PCNet clients only

; workstation records
1000FFFFFF DFBB10US ~ FITSDFBB10US BBSRV01 Z ~ ~ ~ ~ ~ R_BB10_US_OTKNTR ~ ~

```

Figure 7. A Sample RPL.MAP File

**The Workstation Record Identifier:** The workstation entry in the RPL.MAP file contains 15 fields (1 through 9 and A through F). Fields are delimited by space characters. An empty field must contain a tilde (~) character, which serves as a place holder. Table 2 on page 30 describes each of the fields in the workstation entry.

<i>Table 2 (Page 1 of 2). Description of the Workstation Record Fields</i>		
<b>Field</b>	<b>Name / Value</b>	<b>Description</b>
1	100FFFFFFFFF	This field specifies the unique, burned-in 12-character LAN adapter address of the primary LAN adapter in the Remote IPL workstation to which this entry refers. This field can also contain question mark (?) characters for DOS Remote IPL records, which are used as wildcards, for example, 10005AABB???. Do not use these wildcard characters in OS/2 or WorkSpace On-Demand Remote IPL records.
2	DFBB10US	This field is the workstation machine name, which can be up to 15 characters long.
3	~	This field is not used. It must contain a tilde (~) character.
4	FITSDFFBB10US	For DOS Remote IPL, this field specifies the filename of the diskette image file to be used to IPL the DOS workstation. An extension of .IMG is assumed. This field is limited to 8 characters for a DOS workstation. For WorkSpace On-Demand OS/2 Remote IPL, this field specifies the name of the FIT file to be used to start the workstation. This name is relative to the RPLDIR path which is usually \IBMLAN\RPL. An extension of .FIT is assumed.
5	BBSRV01	For DOS Remote IPL, this field is the name of the Remote IPL server where the diskette image file and other files for the DOS workstation are stored. This name is limited to 15 characters (15 bytes).
6	Z or BBDOM	For DOS Remote IPL, this field contains the name of the domain. For WorkSpace On-Demand or OS/2 Remote IPL, this field contains the ID of the Remote IPL boot drive. Use a letter from C through Z. Do not specify a local hard-disk drive letter.
7,8,9	~	<p>For WorkSpace On-Demand OS/2 and DOS Remote IPL, these fields specify parameters for the IBM LAN Support Program device drivers 1, 2, and 3. They contain one tilde (~) character each, separated by spaces, indicating that no parameters are being passed.</p> <ul style="list-style-type: none"> <li>• Device driver 1 corresponds to the DXMA0MOD.SYS file.</li> <li>• Device driver 2 corresponds to the DXMC0MOD.SYS or DXMG0MOD.SYS file.</li> <li>• Device driver 3 corresponds to the DXMT0MOD.SYS file.</li> </ul> <p>You can replace ~ characters according to the device driver's parameter you want to pass along. For example, to pass a locally administered address to driver 2, fields 7, 8, and 9 would be: ~ 400063870001 ~</p>

<i>Table 2 (Page 2 of 2). Description of the Workstation Record Fields</i>		
Field	Name / Value	Description
10	,,,	For DOS Remote IPL, this field specifies additional memory for device drivers 1, 2, and 3. The default is three commas (...). Modify the driver 3 portion of this field (,,driver_3,) when the NetBIOS parameters must be increased over the default values. But be aware of additional memory required at the Remote IPL client. Determine the value for this field experimentally.  These fields are not used when the NDIS protocol stack is specified in the boot block configuration file referenced in field 12.
11	Z or ~	For DOS Remote IPL, this field specifies the LASTDRIVE value. The default is Z. For WorkSpace On-Demand or OS/2 Remote IPL, this field must contain a tilde (~) character.
12	R..... or D.....	This field designates the server record to use for this workstation. This field must match field 12 of one of the server records in the RPL.MAP file on the same Remote IPL server. D..... stands for a disabled record, R..... means enabled.
13	~	This field is reserved and must contain a tilde (~) character.
14	~	This field is not used. It must contain a tilde (~) character.
15	P or ~	If this field is set to P, the workstation has a PC Network adapter and field 12 contains an WorkSpace On-Demand OS/2 server record identifier. Otherwise, this field must contain a tilde (~) character. This field is not used by DOS Remote IPL.

**The Server Record Identifier:** The server record in the RPL.MAP file contains 14 fields. The syntax rules for this record are the same as those listed for the workstation record.

Table 3 describes each of the fields contained in the RPL.MAP file and the values that each of them can contain.

<i>Table 3 (Page 1 of 3). Description of the Server Record Fields</i>		
Field	Name / Value	Description
1	yyyyyyyyyyyy	This field consists of 12 y characters (12 bytes).
2	xxxxxxxx.CNF	This field specifies the boot block configuration file used to start a workstation. See 3.2.1.2, "The Boot Block Definition (.CNF) File" on page 35, for more details.

<i>Table 3 (Page 2 of 3). Description of the Server Record Fields</i>		
Field	Name / Value	Description
3	3 or 0	<p>This field specifies the number of retries in a given period that a requester with no specific entry in the RPL.MAP file must request IPL before a RPL.MAP entry containing a wildcard character (default boot) is used.</p> <p>If there is only one Remote IPL server, this value should be 0. Otherwise, the value must be non-zero so that every Remote IPL server has a chance to service the IPL request.</p>
4	10	This field specifies the time interval in seconds during which the number of unanswered IPL requests specified in field 3 must be received before a default IPL configuration is used.
5	N or A	<p>This field specifies whether the Remote IPL server acknowledges each packet sent. The valid values are either A (Acknowledge) or N (Do not acknowledge). Usually, N is specified since the setting A increases the traffic across the network.</p>
6	IBMLAN\$ or ~	For DOS Remote IPL, this field specifies the net name where the DOS images are located. The images are assumed to be in the IBMLANDCDBIMAGES directory path from this share. For WorkSpace On-Demand or OS/2 Remote IPL, this field must contain a tilde (~) character.
7	DOS~ IBM~ and others	<p>This field is used for a descriptive comment. This field is required when an OS/2 Remote IPL requester is running in a PC Network II or PC Network II/A network environment. OS2~ PCNET indicates running in a PC Network environment, whereas OS2~ PCNETA indicates PC Network/A. The GETRPL post-install utility uses this field also to enable and disable the server records correctly according to the adapter type of the remote boot server. The following naming conventions apply:</p> <ul style="list-style-type: none"> <li>• Token-ring server records must include the string TOKEN.</li> <li>• Ethernet server records must include the string ETHER.</li> <li>• PC Network server records must include the string PCNET.</li> </ul>
8	~	This field is not used. It must contain one tilde (~) character.
9	~	This field is not used. It must contain one tilde (~) character.
10	,,,	This field must contain three commas (,,,) .
11	Z or ~	This field must contain a tilde (~) for OS/2. For DOS Remote IPL, this field specifies the LASTDRIVE value. The default value is Z.

<i>Table 3 (Page 3 of 3). Description of the Server Record Fields</i>		
Field	Name / Value	Description
12	R_Dxxxx R_20_xxxx R_21_xxxx R_230_xxxx R_BB10_US_xxxx	This key field identifies a server record identifier. Each identical yyyyyyyyyyy line must have a different value for this field. Field 12 must begin with R, which enables the record. For DOS server records use R_Dxxxx where xxxx can be any unique string.
13	~	This field is not used. It must contain one tilde (~) character.
14	~	This field is not used. It must contain one tilde (~) character.

**Note**

Unused fields are retained for compatibility with previous versions of LAN Server.

Apart from pointing to the LAN adapter support to be used at the Remote IPL client, the 12th field in the OS/2 server record IDs also represent the OS/2 version to be started at the client. Certain rules apply when working with the LAN Server Administration GUI.

The rules for naming WorkSpace On-Demand server record identifiers are as follows:

s\_BBvv\_nn\_aaapppp... (maximum length is 48 characters)

s defines the state of the remote IPL client:

R - indicates the client is enabled  
D - indicates the client is disabled

WorkSpace On-Demand clients can be created in the enabled or disabled state.

vv defines the major version ID of WorkSpace On-Demand:

10 - indicates the initial WorkSpace On-Demand release

nn defines the language ID version of WorkSpace On-Demand, i.e., US, FR, DE, etc

aaa defines the network adapter type:

OTK - indicates Token-ring  
OET - indicates Ethernet

ppppp... defines a unique string that identifies the network adapter or protocol stack the remote IPL client is to use. This string must match field 4 of the adapter definition record in the NDISDD.PRO file.

Usually the name of the LAN adapter is mentioned here. With WorkSpace On-Demand these characters are also used for the country code, such as: R\_BB10\_US\_OTKNTR, which stands for loading WorkSpace On-Demand, U.S. English version with IBM Token-Ring LAN Adapter support.

The rules for naming server record identifiers for other versions of OS/2 are as follows:

s\_2vv\_aaapppp...(maximum length is 48 characters)

s defines the state of the remote IPL client:

R - indicates the client is enabled  
D - indicates the client is disabled

OS/2 clients are always created in enabled state.

vv defines the major version ID of OS/2:

0 - indicates OS/2 2.0  
1 - indicates OS/2 2.1  
30 - indicates OS/2 3.x  
40 - indicates OS/2 4.0

aaa defines the network adapter type:

OTK - indicates Token-Ring  
OET - indicates Ethernet  
3CE - This is an old naming convention originally used for 3COM ethernet adapters. OET is the recommended prefix for ethernet.

ppppp... defines a unique string that identifies the network adapter or protocol stack that the remote IPL client is to use. The only requirement for this string is that it results in a unique server record identifier.



**Note**

The server record identifiers are used only for standard OS/2 and DOS RIPL. If you do not follow the naming convention described here, the server record identifiers will not be selectable from the graphical user interface for DOS and OS/2 requesters. However, they may be used from the command line.

### **3.2.1.2 The Boot Block Definition (.CNF) File**

The Boot Block Definition files are located in the directory IBMLANRPL.

These definition files define an operating system and the way it is loaded into a Remote IPL workstation. A definition file is required for DOS, OS/2 and WorkSpace On-Demand requesters.

A number of defined requesters make use of the same .CNF file. For example, if two workstations boot the same operating system and have the same type of network adapter installed, they will use the same Boot Block Definition file.

Every server record in the RPL.MAP file must contain a reference to a valid Boot Block Definition file. A number of default .CNF files are provided for all supported network adapters and operating systems.

Typically, you do not need to change Boot Block Definition files unless it is necessary to have the Remote IPL requester operate with different network drivers. The files are in ASCII format and can be edited with any text editor.

Using the Boot Block Definition file, the Remote IPL service creates a boot block that is sent to the requester. This transmission occurs at the very beginning of the IPL process.

Figure 8 on page 36 shows the default WorkSpace On-Demand .CNF file for the IBM Token-Ring Network Adapter.

```

; OS/2 Boot Block Configuration
; (IBM Token-Ring Compatible Adapter)
RPL DOS\RPLBOOT.SYS
DAT DOS\MFSD20.SYS
ORG 1000H
LDR BB10.US\OS2LDR ~ OS2LDR UFSD.SYS MFSD20.SYS
DAT DOS\UFSD.SYS
DAT DOS\TOKENRNG\OS2\PROTOCOL.INI
DAT C:\IBMLAN\DOSLAN\LSP\DXM.MSG
DAT C:\IBMLAN\DOSLAN\LSP\DOS\LT2.MSG
EXE C:\IBMLAN\DOSLAN\LSP\NETBIND.COM ~ ~
; **NETBIOS and IEEE 802.2*****
; DRV C:\IBMLAN\DOSLAN\LSP\DXMTOMOD.SYS PBA=0~S=12~ST=12~C=14~O=N ~ ~
; DRV C:\IBMLAN\DOSLAN\LSP\DXMEOMOD.SYS ~ 10 ~
; **NETBIOS and IEEE 802.2*****
;
;
; **NETBIOS*****
DRV C:\IBMLAN\DOSLAN\LSP\DXMJOMOD.SYS ~ 6 ~
; **NETBIOS*****
DRV C:\IBMLAN\DOSLAN\LSP\DXMAOMOD.SYS 001 ~ ~
DRV C:\IBMLAN\DOSLAN\LSP\DOS\IBMTOK.DOS ~ ~ ~
DRV C:\IBMLAN\DOSLAN\LSP\PROTMAN.DOS /I: ~ ~

```

Figure 8. WorkSpace On-Demand Boot Block Definition File BB1USNTR.CNF

All filename fields either can be expressed relative to RPLDIR, which is specified in the [Remoteboot] section in the IBMLAN.INI file:

RPLDIR = C:\IBMLANRPL

or they can be fully qualified. If they are qualified, the entire path must be specified, including the drive letter and all the subdirectories.

All fields, including the .CNF file parameters, are separated by spaces. Some values in .CNF can contain parameter lists to be used by other software though. Fields of these embedded parameter lists must be separated by tilde (~) characters.

Table 4 describes .CNF file entry types along with their expected file names and parameters:

Table 4 (Page 1 of 2). Description of the .CNF (Boot Block Definition File) Entries	
Entry	Description
RPL	Only one RPL entry can be in a .CNF file. The filename field of this entry specifies the name of the first program to be run on the IPL workstation. No parameters come along with this file entry.

Table 4 (Page 2 of 2). Description of the .CNF (Boot Block Definition File) Entries	
Entry	Description
ORG	Only one ORG entry can be in a .CNF file. The second filename field of this entry specifies the hexadecimal segment number of a contiguous memory block on the IPL workstation. Files following this entry in the .CNF file are bound to this memory address. No parameters come along with this file entry.
DAT	Several DAT entries can be in a .CNF file. Their filename fields specify data files to be stored in the boot block. These files are not used by RPLBOOT.SYS, but they can be read by DOS handle I/O functions. No parameters come along with this file entry.
LDR	Only one LDR entry can be in a .CNF file. The filename field of this entry specifies the name of the loader to use on the IPL workstation. For OS/2 Warp Version 3, this entry must include the following filename and parameters: OS2.30\OS2LDR ~ OS2LDR OS2KRNL RPLMFSD.SYS For WorkSpace On-Demand this entry is: BB10.USOS2LDR ~ OS2LDR UFSD.SYS MFSD20.SYS
EXE	One or more EXE entries can be in a .CNF file. Their filename fields specify names of executable files that are to be used on the IPL workstation. The parameter fields of an EXE entry are passed on to the executable when it runs.
DRV	One or more DRV entries can be in a .CNF file. Their filename fields specify names of device drivers to be used on the IPL workstation. Each driver entry requires the following parameter fields: <ol style="list-style-type: none"> <li>1. The parameter list for the device driver. Fields of this embedded parameter list must be separated by tilde (~) characters in the .CNF file.</li> <li>2. The additional memory requirements of the device driver, if any, are expressed in decimal kilobytes.</li> <li>3. Use the character M if the driver can be moved after initialization. Otherwise, this parameter must be a tilde (~) character.</li> </ol> <p>If the driver can be moved and it requires less memory than the original driver image, RPLBOOT.SYS moves the driver to reclaim the unused memory and adjusts all interrupt vectors that point into the driver's memory area.</p> <p><b>Notes:</b></p> <p>The order of field entries must be considered. Place EXE entries before DRV entries. This change allows movable drivers to be placed in memory freed by completed .EXE files.</p> <p>DRV statements are order dependent. They are processed in reverse order, from last to first.</p>
BASE	Only one BASE entry can be in a .CNF file. The second field of this entry specifies the hexadecimal segment number (paragraph) that is the boot block base address. Default base address is X'00C0H'.

program

### 3.2.1.3 The File Index Table (FIT)

A file index table (FIT) is built for each OS/2 and WorkSpace On-Demand RIPL client machine at machine create time. DOS Remote IPL clients do not require a FIT file. The FIT maps the RIPL client filenames to server filenames. Since the FIT is a flat ASCII file, it can be updated using an ASCII editor.

#### Note

The FIT file is appropriate only for mapping files that will appear to exist on the boot drive (usually Z:) or a LAN attached drive. All other entries (such as local storage devices, hardfiles, CD-ROMs, and so forth) are ignored.

At RIPL boot time, the RIPL server builds a boot block that is used by the RIPL client. The FIT is included in this boot block. This FIT is used by the micro-FSD, the mini-FSD and then by the FSD or redirector.

A number of default FIT files are provided with OS/2 Warp Server and WorkSpace On-Demand. These files are located in the IBMLANRPLFITS directory:

- DEFAULT20.FIT                for Remote IPL OS/2 2.0 clients
- DEFAULT21.FIT                for Remote IPL OS/2 2.1 / 2.11 clients
- DEFAULT30.FIT                for Remote IPL OS/2 Warp Version 3 clients
- DEFAULT40.FIT                for Remote IPL OS/2 Warp Version 4 clients
- DFBB10US.FIT                for WorkSpace On-Demand U.S. English Clients

For WorkSpace On-Demand clients, a .FIT is provided as part of each language client installation. For example: DFBB10nn.FIT, where nn is a two-character language code. Defined in the workstation section of the RPL.MAP file, it is sent to a requester as part of the boot block record.

A FIT file consists of a header record specifying the name of the default network share where files can be found. BBSRV01RPLFILES is an example of a header record where the default network share is RPLFILES on server BBSRV01. The first line of the FIT file must be a UNC name.

The rest of the file consists of file-name mapping records. The mapping records consist of two fields on the same line. The first field represents a filename, and the second represents the actual location of the file on the server. These records consist of a prototype filename or prefix. That means, they would be filenames or prefixes that are to be mapped, follow

ed by a space, and an actual filename or prefix relative to the network share.

A semicolon character (;) is used anywhere on a line of the FIT to begin a comment. Blank lines are ignored. If the prototype matches a proper prefix (part of the path name, ending in \) of the name to be matched; the matched portion is replaced by the actual prefix. If there is an exact match, it is used for substitution. If several prototypes match, the most fully qualified match is selected for substitution.

Network net names can also be included as part of the substitution text; when this is the case, the net name on the first line is ignored. Specifying the network net name also provides the capability to use files and applications located on a server other than the default server.

**Using Wildcard Characters in the FIT File:** Wildcard characters (?) and (\*) are supported subject to the following rules:

- Wildcard characters can appear only in the prototype filename field. They cannot appear in the target server name field.
- Wild card references can be redirected only to the directory level. For example, the following is valid:

```
Z:OS2\*.INI BBSRV01WRKFILESDEFAULTOS2
```

The following is not valid:

```
Z:OS2*.INI BBSRV01WRKFILESDEFAULTOS2ANY.INI
```

- The ? and \* cannot both appear in the same prototype filename field. For example, the following is not valid:

```
Z:OS2ANY???.* BBSRV01WRKFILESDEFALUTOS2
```

**Example FIT Entries:** The following are example FIT file entries.

1. Defines the default Remote IPL server and network share name.

```
BBSRV01RPLFILES
```

2. Translates all references to Z:CONFIG.SYS to MACHINESDEFAULTCONFIG.SYS:

```
Z:CONFIG.SYS MACHINESDEFAULTCONFIG.SYS
```

When this is resolved the default net name is automatically prefixed to the translated name.

```
BBSRV01RPLFILESMACHINESDEFAULTCONFIG.SYS
```

3. Translates all references to Z:OS2 to BB10USOS2:

Z:OS2                      BB10US0S2

Except for explicitly mapped filenames (see line 6), this record results in all Z:OS2\*. \* references being translated to:

BBSRV01\RPLFILES\OS2

4. Subdirectory references from the previous example are also mapped. For example, Z:OS2DLLANYFILE.DLL is translated to:

BBSRV01RPLFILES0S2DLLANYFILE.DL

5. The following is a comment.  
; go to a different share

6. Translates all references to Z:OS2SYSTEMSWAPPER.DAT to BBSRV01WKRFFILESDEFAULTOS2SYSTEMSWAPPER.DAT:

Z:OS2SYSTEMSWAPPER.DAT  
\\BBSRV01\WKRFFILES\DEFAULT\OS2\SYSTEM\SWAPPER.DAT

In this example, the default net name is overridden by specifying the explicit UNC name. The translated filename is to the requester-unique file structure where the requester has read, write, create, delete, and execute access. Swapping here is defined on the remote server in comparison to local swapping.

7. The following translates all other references to be relative to BBSRV01WRKFILESDEFAULT:

Z:                      BBSRV01WRKFILESDEFAULT

Note that the translated path is to the requester-unique file structure on the server for which the requester has read, write, create, delete, and execute authority.

FIT files are stored in the IBMLANRPLFITS directory. Each Remote IPL workstation has its own FIT file stored in this directory. The FIT file is identified by the name given to the requester in the LAN Server Administrator's GUI. For example, the FIT file and location for workstation WSODCLT01, is

IBMLANRPLFITWSODCLT01.FIT

**Constructing the Machine FIT:** Each machine that is defined has its own FIT file defined.

One of the enhancements for Workspace On-Demand client is the method used to build the machine FIT at machine-create-time. Another enhancement is the addition of a user FIT, which will be described in the next section.

The method for constructing a IBMLANRPLFITS<machine-name>.FIT file has been modified. Now when a RIPL client definition is created, the IBMLANRPLFITS<machine-name>.FIT is constructed using the following procedure:

1. Copy the IBMLAN\RPL\FITS\DFBB10nn.FIT (where nn is a two-character language code) into the IBMLANRPLFITS<machine-name>.FIT file.
2. Update all machine name references to be the name of the RIPL client that is being created. That is, change all the occurrences of the word DEFAULT to be the machine name.
3. Check for the existence of the file  
IBMLANRPLMACHINESBB10.nnmachineclass.MC  
machineclass.FIT. If the file exists, it is appended to the file  
IBMLANRPLFITS<machine-name>.FIT.

The machineclass.FIT file typically contains hardware-specific file mapping references (for example: SVGA video support).

#### 3.2.1.4 User-Specific FIT Files

WorkSpace On-Demand introduces support for a two-level dynamic FIT. The first level remains unchanged from the FIT described above, that is, the machine FIT. The second level was added to allow the FIT to be modified to add user-specific information to the machine FIT. The design allows this user-specific FIT to be dynamically changed without rebooting the machine.

##### Note

The user-specific FIT files may only be used with the PMLOGON shell. A customized shell, or an alternate version of OS/2 that does not invoke the PMLOGON application, will not be aware of the user FIT file.

When a user logs on to a WorkSpace On-Demand client, the client searches for any user-specific FIT extensions. If any are found, the in-memory copy of the <machine-name>.FIT file is updated to add the user-specific FIT entries. The copy of IBMLANRPLFITS<machine-name>.FIT on the hard disk is never updated to include user-specific FIT extensions.

The user-specific FIT extensions are inserted following the required UNC reference in the first record of the <machine-name>.FIT and prior to any of the other records. The format for the user-specific FIT is the same format any other FIT file in OS/2.

The user-specific FIT file is located on the domain controller at:

IBMLAN\DCDB\USERS\<user\_name>\<user\_name>.FIT

A user FIT, for example, might contain the following entry:

Z:NETSCAPENETSCAPE.INI BBSRV01WRKFILESUSER01NETSCAPENETSCAPE.INI
--

Figure 9. <User>.FIT Entry - NETSCAPE.INI Redirection

In this case, adding the user FIT allows a different NETSCAPE.INI file to be kept for each user.

The present RIPL design already guarantees the machine names and user names are unique. The shares used in the user FIT can be to any share on the network to which the user has access.

The machine FIT must also include the following line, mapping to the location of the user FIT.

Z:DCDBUSERS	BBSRV01IBMLAN\$DCDB \USERS
-------------	----------------------------

Figure 10. <User>.FIT Entry - User FIT File Redirection

The USERS<username> directory is created in the normal way when a user is added to the domain, using either the NET USER command or the Administrator's GUI. As part of this process, the directory is given an ACL to permit only user access.

In addition, you must not try to access this share (BBSRV01IBMLAN\$) when a user is not logged-on. If an access attempt is made before a logon, the machine ID is used to make the connection, and since the machine ID connections are not torn down at logoff time (or logon time), you will not be able to access the user's FIT file.

### 3.2.2 NDISDD.PRO Control File

The NDISDD.PRO file defines the network adapters that are supported for Remote IPL. It is an ASCII file that contains a definition record for each adapter type. When a WorkSpace On-Demand workstation is defined through the GUI, the network adapter listbox contents are controlled by the NDISDD.PRO file.

The listbox (for WorkSpace On-Demand definitions) has four filters:

1. The NDISDD.PRO record must have all four fields specified.
2. The NIF file specified in field 3 of the NDISDD.PRO record must be present in the d:IBMLANRPLBB10.USIBMCOMMACHS directory.



3. Field 4 of the NDISDD.PRO record is used to construct a server record ID. RPL.MAP is searched looking for a server record that is enabled and contains a matching server record ID.
4. If a matching server record is found, the CNF file specified in the server record is read and the existence of each file referenced by the CNF file is verified.

When an adapter is selected, the server uses the information in the third and fourth fields to build the CONFIG.SYS and PROTOCOL.INI file entries for the WorkSpace On-Demand workstation being defined.

Each definition record consists of four fields. The last two fields have been added in WorkSpace On-Demand. Table 5 describes the NDISDD.PRO fields and their layout:

<i>Table 5. Description of the NDISDD.PRO File</i>	
Field	Description
Field 1	This field specifies the DOS NDIS device driver name associated with an adapter.
Field 2	This field specifies the subdirectory name, xxxxxxxx, that contains the adapter-specific configuration files needed to remote boot the RIPL client. The directory d:\IBMLAN\RPL\IBMCOM\xxxxxxx\ contains the configuration files needed to RIPL OS/2 clients. The directory d:\IBMLAN\RPL\DOS\xxxxxxx\ contains the configuration files needed to RIPL both DOS and OS/2 clients. <i>d</i> is the drive where the \IBMLAN\RPL directory has been installed.
Field 3	This field specifies the NIF filename for the OS/2 device driver associated with an adapter.
Field 4	This field specifies the .CNF filename ID reference for server record IDs in RPL.MAP. This value is used in the construction of the server record identifier for WorkSpace On-Demand workstation records. It specifies the adapter-unique portion of the server record ID for WorkSpace On-Demand definitions.

**Note**

Fields 3 and 4 were added for WorkSpace On-Demand support and must be specified for the adapter to be usable by a WorkSpace On-Demand workstation.

Figure 11 on page 44 shows the IBM Token-Ring entry from the default NDISDD.PRO. In the example shown in Figure 11 on page 44, the DOS NDIS driver is listed in the first field IBMTOK.DOS. The second field gives the directory containing the DOS drivers for this adapter, d:IBMLANRPLDOSTOKENRNG, and the directory containing the OS/2 drivers is d:IBMLANRPLIBMCOMTOKENRNG. The name of the OS/2 NIF file is IBMTOK.NIF

from the third field, and in the forth field the CNF filename reference for the server record ID is R\_BB10\_US\_OTKNTR.

IBMTOK.DOS	TOKENRNG	IBMTOK.NIF	OTKNTR
------------	----------	------------	--------

Figure 11. NDISDD.PRO with IBM Token-Ring Entry

### 3.2.3 Other RPL Files and Directories

The complexity associated with Remote IPL is largely created by the many files that are used for configuration. During the course of this section, we will attempt to provide some understanding of the directory structure as well as the files and directories that are created when WorkSpace On-Demand is installed. We will also examine which of the files are used when a new workstation is defined.

When OS/2 Warp Server is installed with RIPL support, it creates two subdirectories:

IBMLANRPL

IBMLANRPLUSER

When WorkSpace On-Demand is installed over OS/2 Warp Server, a number of additional directories are created within the RPL and RPLUSER directories. WorkSpace On-Demand allows for multiple language versions to be supported on the same machine. To allow for multi-language support, directories have the following naming convention:

IBMLANRPLBB10.<nn>

where nn is the two-character language identifier.

#### Note

During the course of this redbook, we assume that the U.S. English version is installed, and we use BB10.US to refer to the U.S. English version of WorkSpace On-Demand.

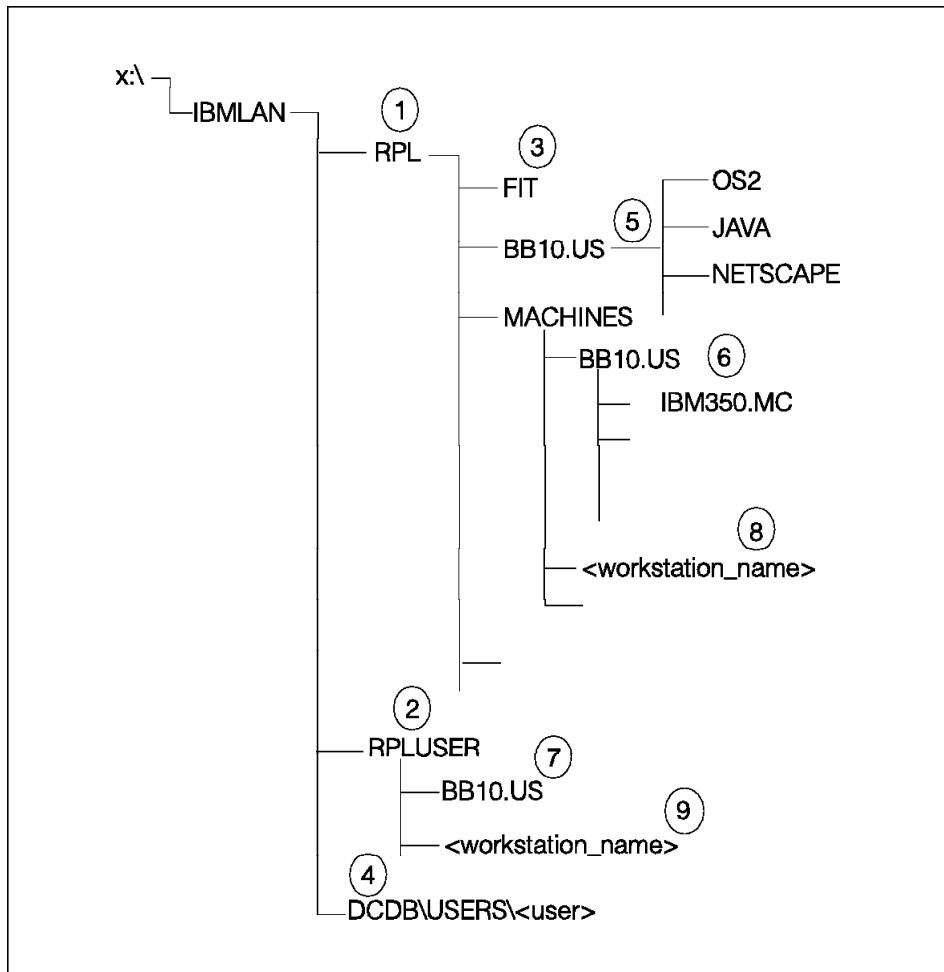


Figure 12. Directories and Files

Figure 12 shows a summary of the directories and files that are required to provide Remote IPL support.

1. The RPL directory contains common read-only files that are required by the server and the Remote IPL workstations. For example:

\IBMLAN\RPL\BB10.US\OS2

\IBMLAN\RPL\BB10.US\MMOS2

Application files could also reside here. For example, if you selected Netscape during installation, the directory it installs into is:

\IBMLAN\RPL\BB10.US\NETSCAPE

Other important files contained in the IBMLANRPL directory are the RPL.MAP and the Boot Block Definition (.CNF) files.

**Note**

When the remote boot service is started, this directory is shared under the net name RPLFILES.

2. The RPLUSER directory usually contains all the Remote IPL workstation unique files. Most of these files are read/write, for example:

`\IBMLAN\RPLUSER\<defaultnc>`

**Note**

When the remote boot service is started, this directory is shared under the net name WRKFILES.

3. All machine FIT files are located in the FIT file directory. This includes the default FIT files as well as the FIT files created for each workstation when it is defined.
4. User FIT files are stored in the user directory within the DCDB.
5. Within the IBMLAN directory, there are a number of BB10.US subdirectories. This particular BB10.US subdirectory contains the actual read-only operating system and application files. Device drivers, executables and dynamic link libraries are among some of the file types located here.
6. This BB10.US subdirectory (IBMLANRPLMACHINESBB10.US) contains the various Machine Class templates. For example, some of the Machine Classes that are shipped with WorkSpace On-Demand are:
  - `\IBMLAN\RPL\MACHINES\BB10.US\ISAVGA.MC`, the generic ISA Bus personal computer with VGA display adapter Machine Class template
  - `\IBMLAN\RPL\MACHINES\BB10.US\IBM350.MC`, the predefined machine class template for the IBM Personal Computer 350 model 6272-880
  - `\IBMLAN\RPL\MACHINES\BB10.US\IBM730.MC`, the predefined machine class template for the IBM Personal Computer 730 model 6877-KAZ
7. This BB10.US directory contains the files that are designated read/write type files. Templates of these files are located in this directory and then copied into the workstation directories when it is created.

8. This is the first of the two workstation directories. This directory contains files that the workstation would only have read access and not write access. Files such as the workstation's CONFIG.SYS, PROTOCOL.INI, and so forth are contained here.
9. This is the second workstation directory that contains files that the workstation would normally read and write access. Files such as the OS2.INI, OS2SYS.INI, and so forth would be contained here.

### 3.2.4 Workstation Files

In this section, we describe the files and directories associated with a workstation definition. Workstations can be defined using the GUI or from the command line. When a Remote IPL workstation is defined, two directories are created to maintain that workstation's specific files. The two directories are:

IBMLANRPLMACHINES<requester\_name>

IBMLANRPLUSER<requester\_name>

where <requester\_name> is the name displayed on the LAN Server Administrator's GUI.

Once the two directories are created, a number of files are placed in them, enabling the workstation to boot as it has been defined.

For the purposes of this example, we will be using the workstation name BBCLTL01. For example, the directories created for the Remote IPL workstation BBCLTL01 are:

\\BMLAN\\RPL\\MACHINES\\BBCLTL01

\\BMLAN\\RPLUSER\\BBCLTL01

The workstation's files are created by the server when the workstation is defined. For example, as a first step, the server copies all files from the IBMLANRPLUSERBB10US directory to the workstation's read/write directory within the RPLUSER directory. Some of the files will be modified. Table 6 on page 48 lists some of the more significant files, their location and source.

<i>Table 6 (Page 1 of 2). Files Associated with the RIPL Client</i>	
<b>IBMLANRPLMACHINES\BBCLTL01</b>	
<b>File</b>	<b>Source</b>
CONFIG.SYS	<p>This file is constructed.</p> <p>The CONFIG.SYS file in the machine class directory contains all the necessary drivers required to support that Machine Class. It normally does not contain network device drivers.</p> <p>For example, if a workstation is defined based on the IBM PC 350 machine class and an IBM Token-Ring adapter was selected, the server builds the network entries for the CONFIG.SYS for the IBM Token-Ring adapter and appends them to the CONFIG.SYS in the IBMLANRPLMACHINESBB10.USIBM350.MC directory.</p>
BBCLTL01.INI	<p>This INI file is copied out of the Machine Class subdirectory. It is then modified based on the values you enter in the LAN Server Administrator's GUI when creating a workstation.</p> <p>This file is in the same format as the OS/2 INI files, but they are unrelated.</p>
IBMCOMPROTOCOL.INI	<p>This file is constructed.</p> <p>The server uses the THINLAPSR.EXE (similar to the original THINLAPS.EXE) to construct a PROTOCOL.INI file. Input to THINLAPS is the PROTOCOL.INI file in the Machine Class subdirectory and the adapter selected in the LAN Server GUI. The card information is extracted from the NDISDD.PRO file.</p>
IBMLANIBMLAN.INI	<p>This file is constructed by taking the file from Machine Class directory and adding the workstation name entered in the GUI.</p>
<b>IBMLANRPLUSERBBCLTL01</b>	
AUTOEXEC.BAT	This file is copied from IBMLANRPLUSERBB10.US
IBMLANACCOUNTSNET.ACC	This file is copied from IBMLANRPLUSERBB10.US
OS2OS2.INI	<p>RPLMACHINESBB10.US&lt;machine_class&gt;OS2</p> <p>This file is cloned exactly as it appears in the above directory.</p>
OS2OS2SYS.INI	<p>RPLMACHINESBB10.US&lt;machine_class&gt;OS2</p> <p>This file is cloned exactly as it appears in the above directory.</p>

Table 6 (Page 2 of 2). Files Associated with the RIPL Client	
IBMLANRPLMACHINES\BBCLTL01	
File	Source
OS2MDOSWINOS2 ATM.INI, CONTROL.INI, MOUSE.INI, MSD.INI, P9000RES.INI, P9X00RES.INI, PROGRAM.INI, STARTUP.GRP, SYSTEM.INI, WIN INI, WOS2ACCE.GRP, WOS2MAIN.GRP	RPLUSERBB10.USOS2MDOSWINOS2  These files are cloned exactly as they appear in the above directory. The SYSTEM.INI and WIN.INI are taken from the RPLMACHINESBB10.US<machine_class> OS2MDOSWINOS2 directory

\* EBOOKIE (BKMTAGS.DEF)





---

## Chapter 4. WorkSpace On-Demand Planning

This chapter describes the required steps to plan for the installation of WorkSpace On-Demand. During the course of this chapter, we cover the hardware and software prerequisites as well as some rule-of-thumb planning information.

It is imperative that you review this chapter before beginning the installation of WorkSpace On-Demand.

For your planning convenience, a WorkSpace On-Demand Product Assurance Checklist is available to help you plan and manage your WorkSpace On-Demand implementation efforts. It is available from the IBM Personal Solutions System Center's Website at: <http://pssc.sl.dfw.ibm.com> in the P.S.S.C. Technical Documents section.

---

### 4.1 Software Prerequisites

As discussed in 2.3, "WorkSpace On-Demand Components" on page 13, all WorkSpace On-Demand software is installed on the WorkSpace On-Demand server. One exception to this statement is the WorkSpace On-Demand Administration Client, which can be installed either on the WorkSpace On-Demand server or on a separate workstation. In any case, there is prerequisite software that must be in place prior to installing WorkSpace On-Demand software. The following sections discuss these software prerequisites.

#### 4.1.1 On the Target WorkSpace On-Demand server

The following software must be installed on the target server prior to installing the WorkSpace On-Demand feature:

- IBM OS/2 Warp Server, OS/2 Warp Server Advanced, or OS/2 Warp Server Advanced - SMP, with the following options installed:

#### **Note on OS/2 Warp Server**

Because it does not support the 386 HPFS file system, OS/2 Warp Server is not generally recommended for use with WorkSpace On-Demand except under controlled, limited circumstances. See Section B.1, "OS/2 Warp Server SYS0071 Errors When Attempting to Start an Application" on page 255, for information regarding resource limitations when using OS/2 Warp Server and WorkSpace On-Demand. Also see Section 4.7.1, "WorkSpace On-Demand Manager Server Performance Considerations" on page 64, for a discussion of the benefits of the 386 HPFS file system when used with WorkSpace On-Demand.

- Remote IPL Support

This is a required component and must be installed.

- File and Print Sharing

This is a required component and must be installed.

- LAN Server Administration Tool

Required if you plan to use the server as an *Administrator Workstation*.

- FixPak 22 or later if you are installing WorkSpace On-Demand on OS/2 Warp Server or on OS/2 Warp Server Advanced. This is for base OS/2 fixes.

#### **Note**

If available, FixPaks later than 22 should be installed. FixPak 22 is *not* a requirement for OS/2 Warp Server SMP.

#### **4.1.1.1 Year 2000 Readiness on WorkSpace On-Demand server**

The WorkSpace On-Demand components are Year 2000 Ready out-of-the box; however FixPak 32 for Warp V3.0 and LAN fixes IPX8266 (for Warp Server and Warp Server Advanced) or IPX8504 (for Warp Server Advanced - SMP) must be applied prior to installing the WorkSpace On-Demand Manager.

#### **4.1.2 On the Target WorkSpace On-Demand Administration Client**

The following software must be installed on the machine used for the WorkSpace On-Demand Administration Client:

- IBM OS/2 Warp Version 4
- File and Print Client. This is a required component.

- LAN Server Administration Tools

#### **4.1.2.1 Year 2000 Readiness on the WorkSpace On-Demand Administration Client**

To achieve Year 2000 Readiness of the WorkSpace On-Demand Administration Client, you need to install the appropriate fixes to the LAN components (see Section 4.1.1.1, “Year 2000 Readiness on WorkSpace On-Demand server” on page 52, for details) plus the appropriate base Warp 4 FixPack - FixPak 4.

### **4.1.3 On the WorkSpace On-Demand Client**

There is no user-installed software on this machine.

- BIOS modified to boot machine from the network

#### **4.1.3.1 Year 2000 Readiness on the WorkSpace On-Demand client**

The WorkSpace On-Demand client is Year 2000 Ready out-of-the-box.

---

## **4.2 Hardware and Network**

The following is a list of the supported RIPL adapters:

### **ISA Adapters**

- 3COM EtherLink 16 (3C507)
- 3COM EtherLink III (3C509)
- 3COM EtherLink III-Combo (3C509-COMBO)
- 3COM EtherLink III-Combo (3C509B-COMBO)
- 3COM EtherLink III-TP (3C509-TP)
- 3COM EtherLink III-TP (3C509B-TP)
- 3COM EtherLink III-TPO (3C509-TPO)
- 3COM Fast EtherLink XL PCI (3C905-TX)
- IBM EtherJet 10/100 BASE-T Adapter with Wake On LAN (85H9928)
- IBM LAN Adapter for Ethernet (48G7169)
- IBM LAN Adapter for Ethernet CX (60G6015)
- IBM LAN Adapter for Ethernet TP (60G6005)
- IBM Token-Ring Network 16/4 Adapter
- IBM Token-Ring Network 16/4 Adapter II
- IBM PCI Token-Ring

- IBM PCI Token-Ring Wake On LAN
- IBM Auto 16/4 Token-Ring ISA Adapter (92G7632)
- IBM Auto 16/4 Turbo Token-Ring ISA Adapter (72H3296)
- IBM AutoWake 16/4 Token-Ring Adapter (55H6810)
- IBM Token Ring ISA-16 Adapter (60G3994A)
- IBM Auto 16/4 Turbo WOL Token-Ring ISA Adapter
- MADGE Smart 16/4 AT PLUS Ringnode (52-03)
- MADGE Blue Plus 16/4 ISA (62-03)
- MADGE Blue Plus 16/4 ISA PnP (62-04)
- MADGE Straight Blue ISA Plus Blue Box (62-02)
- Standard Microsystems Ethercard Elite 16 Ultra (8216)
- Standard Microsystems Ethercard Elite 16C Ultra (8216C)
- Standard Microsystems Ethercard Elite 16T Ultra (8216T)
- Standard Microsystems EtherEZ 10Base T ISA (8416T)
- Standard Microsystems EtherEZ ISA (8416B)
- Standard Microsystems EtherEZ ISA (8416BTA)
- Standard Microsystems TokenCard Elite (8115T)

### Micro Channel Adapters

#### Note

Although Micro Channel machines are not supported as a Machine Class that ships with WorkSpace On-Demand, it is possible to create your own Machine Classes to provide support for Micro Channel machines. See Chapter 8, “WorkSpace On-Demand Machine Classes” on page 149, for a complete discussion of how to create Machine Classes.

- 3COM Etherlink/MC adapter
- IBM LAN Adapter/A for Ethernet
- IBM LANStreamer MC 32 Adapter
- IBM PC Network Baseband Adapter/A
- IBM PC Network Broadband Adapter II/A
- IBM PS/2 Adapter/A for Ethernet Networks
- IBM Token-Ring Network Adapter/A

- IBM Token-Ring Network 16/4 Adapter/A

Support has also been added for the following adapters:

- IBM Etherjet
- Crystal Semi 8920 Chip Set

---

## 4.3 Network Topology

WorkSpace On-Demand uses the RIPL protocol to enable the clients to gain access to the code. The nature of this process makes it heavily dependent on the network. The RIPL process uses NetBIOS, which means that it is not routeable. WorkSpace On-Demand therefore requires that the WorkSpace On-Demand server and WorkSpace On-Demand client be within the same *physical* LAN.

### Note

A physical LAN is defined as one or more LAN segments connected by bridge(s). It is possible to separate a WorkSpace On-Demand client from its WorkSpace On-Demand server by bridges. However, LAN bridges have no flow control mechanism, and when they experience extremely heavy traffic, they can become *congested* and discard frames instead of forwarding them across the bridge. For this reason, we recommend that you limit the number of bridges, or hops, between a WorkSpace On-Demand client and its WorkSpace On-Demand server whenever possible. Ideally, a WorkSpace On-Demand server and the WorkSpace On-Demand clients it supports should be on the *same* LAN segment.

### 4.3.1 WorkSpace On-Demand and Network Capacity

WorkSpace On-Demand clients make their heaviest demands on network resources during the initialization, or boot, process (see Section 2.3.1.4, “Application Roaming” on page 15, for a discussion of the initialization process). During this process, the WorkSpace On-Demand server sends the Machine Class hardware image and user-specific environment to each WorkSpace On-Demand client machine requesting initialization.

This transfers approximately 14 MB of data, in several thousand frames, to each WorkSpace On-Demand client. This type of network loading can stress network components, particularly during periods when multiple WorkSpace On-Demand clients are concurrently requesting initialization from the WorkSpace On-Demand server.

Because network computing makes heavy demands on the networking infrastructure, it is important to consider the following issues and confirm if your network will efficiently support WorkSpace On-Demand, or whether you need to make modifications to your network prior to installing WorkSpace On-Demand.

- Bandwidth Contention

What are the current usage peaks? Will periods when WorkSpace On-Demand clients are logging on or off of the network exacerbate existing bandwidth contention?

- Broadcasts Enabled

Are the bridges, or routers that support bridging, configured to forward all-routes broadcasts?

- Bridge Congestion

Do bridges currently fail to forward frames because they are congested (see bridge statistics)? What will be the impact of adding WorkSpace On-Demand clients?

- Frame Size

What is the smallest frame size configured for your bridges? Is it large enough to efficiently accommodate RIPL requests?

Are the frame sizes of your workstations and bridges *coordinated* to work together efficiently?

- Hop Count

Are the WorkSpace On-Demand servers located on the same LAN segment as the WorkSpace On-Demand clients they support? If not, can the distance between them, in terms of hop count or bridges, be reduced to the minimum?

---

## 4.4 Applications Supported by WorkSpace On-Demand

Applications developed for network computing differ from client/server applications in that their component logic is provided by different network entities. For example, in a typical client/server application, the presentation logic is often proprietary and is always provided by the client. Business logic is often provided by the client, but in the three-tiered client/server model, it is provided by another entity like an application server. In any case, in the client/server application model, data logic is always provided by a server.

The network computing application model also provides presentation logic at the client, but differs from the client/server model in that the presentation

logic, or user interface, is based on *open standards*, such as HTTP, rather than on a proprietary interface like a Windows or OS/2 desktop. Business logic, like data logic, is *always* provided by an entity other than the client. A good example of a *true* network computing application would be an application similar to IBM's Host On-Demand. This application uses a Web browser to provide a user interface that runs a 100% Pure Java application supplied from a remote server. Using Host On-Demand, 3270 mainframe data is accessed through another, gateway, server. In this case, both the application or business logic and host data are supplied by other servers.

Based on this brief explanation and the nature and type of applications that ship with the product, it should be apparent that WorkSpace On-Demand is an excellent choice for migrating end-users from client/server to network computing.

#### **4.4.1 Applications Shipped with WorkSpace On-Demand**

WorkSpace On-Demand provides the following application software. The WorkSpace On-Demand installation creates the required directory structures and makes provisions for configuring parameters like IP addresses, subnet masks, and so forth. Detailed installation instructions for the following applications are provided as README files:

- TCP/IP
  - TFTP
  - FTP
  - Telnet
  - SMTP
- Pcomm lite
- Netscape Navigator 2.02
- Java 1.1.1

#### **4.4.2 Applications Supported by WorkSpace On-Demand**

WorkSpace On-Demand provides *installation aid* for the following application software. Installation aid consists of detailed instructions that ship with the product as README files for the following products:

- Lotus Notes 4.51 Full Client
- Cytrix Windows 3.1 Client

#### **4.4.3 Applications Compatible with WorkSpace On-Demand**

The following applications have been tested and are known to work as network-public applications in conjunction with WorkSpace On-Demand:

- DB2/2 Client Application Enabler
- Microsoft Office 97
- Microsoft Office 96 for Windows 3.1

- Lotus SmartSuite 96 for Windows 95
- Lotus SmartSuite 96 for Windows 3.1
- Lotus SmartSuite 96 for OS/2
- Visual Foxboro (Win32s application.)
- Star Office

#### 4.4.4 Application-Specific Environment Variables

WorkSpace On-Demand provides the capability to define both systemwide and user-specific application environments. These include the PATH statements, LIBPATH statements, DPATH statements, and environmental variables necessary to insure successful execution of the applications configured for each WorkSpace On-Demand end-user.

#### 4.4.5 The Simplified, or Restricted, Shell: WorkSpace On-Demand Application Considerations

As discussed in 2.3.1.1, “The WorkSpace On-Demand Shell” on page 14, the WorkSpace On-Demand user interface restricts end-users to executing only those applications that have been configured for them. This limitation may require additional planning in order to insure that the facilities necessary to support any given application are available to the end-user. For example:

- No command prompt (unless *explicitly* configured by system administrator)

Is the application packaging completely self-contained?

- No templates folder on the desktop

Are the folders/directories required by application output defined, configured and available to the end-user?

- No workplace shell desktop

Do the applications attempt to register with the desktop?

- No systemwide help facility

Do the applications provide adequate contextual help facilities?

- No system object

User preferences must be configured from the server.

- No Lockup

Users are expected to log off.

- No archive settings

These are not really needed in a server-based environment.

- Cannot change monitor resolution



This is controlled by the administrator.

---

## 4.5 Migration Considerations

It is imperative that you know what directories and files are created by the application's installation process. Other required information includes:

- INI files that may be created or modified
- Application migration considerations
- Management infrastructure
- Network infrastructure

---

## 4.6 Resource Requirements

This section discusses the hardware resources, such as CPU type, hard disk capacity and RAM, as well as other resources created by configuration parameters, such as the number of request buffers and sessions required by the WorkSpace On-Demand product.

### 4.6.1 Processor

We recommend a 90 MHz Pentium processor or equivalent as the *minimum* processor for a WorkSpace On-Demand server. While WorkSpace On-Demand will run on any Intel, or compatible, '486 CPU of at least 66 MHz, the OS/2 Warp Server base code will exploit features of the Pentium hardware, if it is present. In cases where multi-function server support is required, see Section 2.3.2.2, "Multi-Function Server Support" on page 16, or where client-loading is heavy, processors with higher clock speeds should be considered. SMP should be considered in cases where multi-function support is required *and* when the supported applications are written to exploit parallel processing, as in DB2/2, for example.

### 4.6.2 Memory and Hard Disk Required for Remote IPL Services

A major component of WorkSpace On-Demand is an enhanced RIPL services capability. See Section 2.4, "RIPL Functional and Usability Enhancements" on page 17, for a discussion of RIPL enhancements in the WorkSpace On-Demand product. The installation of a Remote IPL server requires slightly more memory and more hard disk space than a server that does not support Remote IPL. The following table lists the memory and hard disk space recommendations for a WorkSpace On-Demand server. These recommendations are in addition to any other application requirements, such as when CM/2, DB2/2, or other applications are installed on the server in addition to the WorkSpace On-Demand Manager.

**Note**

These numbers reflect testing done at the ITSO.

*Table 7. Additional Memory Requirements for Remote IPL Service*

Memory	DOS	+ 0.7 MB	DOS Remote IPL service
	OS/2	+ 0.8 MB	OS/2 Remote IPL service
	Both	+ 1.0 MB	Both DOS and OS/2 Remote IPL service
Hard Disk	DOS	+ 16.7 MB	Includes DOS LAN Support Program, Remote IPL copy of DOS LAN Services, DOS 6.3, and one DOS image
		+ 0.7 MB	For each additional DOS image
	OS/2	+ 71.2 MB	Includes DOS LAN Support Program Remote IPL copy of User Profile Management, LAPS, and OS/2 LAN Requester, a Remote IPL copy of OS/2 2.11, and support for one Remote IPL workstation
		+ 0.4 MB	For each additional Remote IPL workstation
		+1-10 MB	Swap space for <i>each</i> WorkSpace On-Demand client configured to swap remotely

A WorkSpace On-Demand client that has been configured to swap remotely on the WorkSpace On-Demand server should have at least 16 + MB of RAM installed. WorkSpace On-Demand clients configured to support local swapping need slightly less RAM. See Section 4.7.2, “WorkSpace On-Demand client Performance Considerations” on page 67, for a discussion on WorkSpace On-Demand client RAM requirements.

### 4.6.3 Hard Disk Required for WorkSpace On-Demand

The memory requirements provided for WorkSpace On-Demand clients are guidelines only and may not be appropriate for a specific environment due to the number, size, and complexity of supported user applications. Remember, the goal is to provide sufficient memory at the WorkSpace On-Demand client to *keep swapping activity to the absolute minimum*.

Therefore, the amount of server disk space allocated for each WorkSpace On-Demand client’s swap space cannot be determined without considering the applications and the amount of memory configured for each WorkSpace On-Demand client. Memory-constrained clients will require more swap space than non-memory-constrained clients, and hard disk planning should take this fact into account.

In addition to the files needed to support Remote IPL services, installing WorkSpace On-Demand requires additional hard disk space on the WorkSpace On-Demand server. This disk space is needed to accommodate the WorkSpace On-Demand Manager and the applications and utilities that can be optionally installed on each WorkSpace On-Demand client. The table that follows documents the hard disk requirements of these WorkSpace On-Demand components.

Table 8. Disk Requirements for WorkSpace On-Demand System Components		
Component	Disk Space	Required Disk Space
CMD line interface, WorkSpace On-Demand Manager, Remote IPL services upgrade, Java and Netscape	28.72 MB	90 MB
WorkSpace On-Demand client	45.43 MB	
Predefined Machine Classes and printer drivers	15.85 MB	
Optional Applications and Utilities		
Java 1.1.1		11.08 MB
DOS Support		1.42 MB
WIN/OS2 Support		21.51 MB
REXX		2.09 MB
Multi-Media Support		10.48 MB
Optional System Utilities		0.71 MB
TCP/IP		8.62 MB
PCOMM Lite		7.42 MB
Netscape 2.02		5.57 MB
TOTAL DISK SPACE REQUIRED		159.1 MB

**Note**

The disk space sizes in this table were measured with a beta version of the product. If you install each component selectively, the total size may vary. TCP/IP, PCOMM Lite and Netscape 2.02 use common component such as MPTS. The size in this table for TCP/IP includes the disk space size for the common component, but PCOMM Lite and Netscape 2.02 does not include this disk space size. If you install PCOMM Lite or Netscape 2.02 selectively, add 6 MB to each component.

#### 4.6.4 Memory Required for WorkSpace On-Demand Image

As of the publication of this redbook, the WorkSpace On-Demand image occupies between 12 and 14 MB of RAM. You should plan to add this much additional memory to the WorkSpace On-Demand server to insure that client requests are serviced from cache. Please note that we recommend using the 386 HPFS file system configured with enough cache to service at least 95% of all disk read requests from the 386 HPFS cache. See Section 4.7.1, "WorkSpace On-Demand Manager Server Performance Considerations" on page 64, for more details on the 386 HPFS file system and cache.

#### 4.6.5 Network Resources

This section deals with the parameters that effect the number of clients, users and applications that can be supported by a WorkSpace On-Demand server.

The Remote IPL service requires additional network-related resources. Some of the configuration parameters in the IBMLAN.INI and PROTOCOL.INI files need to be changed. The values attributed to these parameters depend on the number of WorkSpace On-Demand clients supported by the WorkSpace On-Demand server.

The following guidelines take into account only the resource requirements necessary to support Remote IPL. Add the resource requirements for Remote IPL to any other requirements, such as fault tolerance, local security, communication with local IPL requesters and other servers, and so forth.

The following parameters in the Server and Remoteboot sections of the IBMLAN.INI file might need adjustment for your environment after installation of the Remote IPL service. For more information about these parameters and their interdependency on other parameters, refer to *LAN Server Network Administrator Reference Volume 2: Performance Tuning*, S10H-9681.

In the following information,  $n$  stands for the number of supported WorkSpace On-Demand clients;  $m$  stands for the number of both WorkSpace On-Demand clients and DOS Remote IPL requesters, if both are supported by a single WorkSpace On-Demand server.

Table 9. Relevant Remote IPL Parameters		
Server Section in IBMLAN.INI		
Parameter	Value	Description
NUMREQBUF	4 x n	Description, Maximum value: 408
MAXOPENS	n x (65 + max files opened per workstation)	n x (65 + maximum number of user files opened per workstation, including applications and data files) Maximum value: 8000 Note: This parameter is not needed for a Remote IPL 386-HPFS server.
MAXSESSOPENS	100+max user files per workstation	100 + maximum number of user files opened per workstation, including applications and data files Note: This parameter is not needed for a Remote IPL 386-HPFS server.
Remote Boot Section in IBMLAN.INI		
MAXTHREADS	6	For large environments, increasing this parameter might decrease performance. Specify a value that is not greater than 6.
Protocol Section in PROTOCOL.INI File		
NCBS	4 x n	
SESSIONS	m	

#### Note

A single WorkSpace On-Demand client logged on as GUEST with no applications configured for its desktop requires a single session with four connections and 81 open files. This information is based on a beta version of the product.

## 4.7 Performance Planning for WorkSpace On-Demand

This section discusses the issues related to WorkSpace On-Demand performance from two perspectives: server and client.

#### Note

See section 4.3, “Network Topology” on page 55, for a discussion of the network issues that directly affect WorkSpace On-Demand performance.

### 4.7.1 WorkSpace On-Demand Manager Server Performance Considerations

When you consider that all WorkSpace On-Demand clients obtain their applications and data from the WorkSpace On-Demand server, the importance of providing a robust and responsive server platform becomes obvious. The following issues should be considered in order to obtain maximum performance and responsiveness from the WorkSpace On-Demand server.

- DISK SPACE

Because WorkSpace On-Demand clients receive all of their system and application code from the server, client performance is directly affected by the speed, responsiveness and throughput of the server’s disk subsystem. For this reason, the WorkSpace On-Demand server should be equipped with the most robust disk subsystem, in terms of access time and total data throughput, that is practical for your installation. There is also the issue of data integrity to consider which may dictate implementing RAID on your WorkSpace On-Demand server(s). Some of the factors relating to disk subsystem selection that you may wish to consider include:

- Controller Type - (SCSI recommended)

The type of disk subsystem you choose to implement also effects the overall performance of your WorkSpace On-Demand installation. We recommend a SCSI controller for the following reasons:

- Bandwidth. (*Wide SCSI 16-bit* or *Ultra SCSI 32-bit* data path)
- *Tagged* command queuing (overlapping disk I/O)

- Disk Performance and Memory Considerations

Total disk performance is an important consideration in optimizing overall system performance. In general, a greater throughput rate is the more critical selection criterion; however, reduced access time contributes to overall system performance as well.

- Sustained throughput rate
- Total access and seek times

- File systems

- 386 HPFS - (recommended)

- FAT
- HPFS

The choice of which file system to implement is also an important performance consideration. We recommend implementing an HPFS 386 file system for several reasons. The most important reason is that the 386 HPFS file system provides the capability of configuring an 386 HPFS disk cache large enough to accommodate a client's software image and its application(s). Loading a client's software image and applications from RAM instead of disk is a great performance enhancement. A typical WorkSpace On-Demand's image is approximately 14 MB in size.

As a general rule-of-thumb, you should plan on 48 MB as a minimum memory requirement, with 64 MB as a more typical WorkSpace On-Demand server configuration. However, as you will see in the discussion that follows, optimal memory size should be determined by the size of the 386 HPFS cache, assuming that you choose to follow our recommendation and implement a 386 HPFS file system on your WorkSpace On-Demand server. The amount of memory available to a WorkSpace On-Demand server determines how much memory can be allocated to the 386 HPFS cache.

We recommend that the size of the 386 HPFS cache should be made large enough so that at approximately 95% of all *READ* requests are serviced by the cache. You can use the /STATS parameter of the CACHE386.EXE command to query cache-hit statistics.

For additional information on the HPFS386 cache, refer to the *LAN Server Network Administrator Reference Volume 2: Performance Tuning*, S10H-9681.

The percentage of cache read hits will increase as the 386 HPFS cache increases in size so that more and more of the WorkSpace On-Demand image is contained in within the cache. Things that increase the size of the WorkSpace On-Demand image include:

- Number of supported machine classes
- Number of supported clients
- Number of supported users
- Number of resource definitions, aliases, network printers, home directories, and so forth
- Number of supported applications
- Complexity and size of supported applications
- Size and complexity of FIT files

Therefore, on a robust WorkSpace On-Demand server, it is quite possible that 64 MB of memory is not sufficient to insure that 95% of disk read requests are serviced from cache; so you should plan your WorkSpace On-Demand server's memory requirements accordingly.

- RAID

If data integrity and ease of recovery are important considerations, you should consider implementing RAID on your WorkSpace On-Demand server(s). While all the benefits and implications of implementing RAID are beyond the scope of this chapter, you should also consider that there may be some performance implications when implementing RAID as well. For example:

- RAID Level

- Disk Mirroring, Disk Duplexing, Parity, and so forth.

- Data Striping

- Mirrored Reads

- Write Policy

- Write-Through vs. Write-Back

- Additional Server Memory

How much additional memory is necessary to support a WorkSpace On-Demand server? There are several issues to consider:

- 386 HPFS Cache

- Sufficient memory to insure 95% cache hit-rate for read requests

- RIPL Support

- See Section 4.6.2, "Memory and Hard Disk Required for Remote IPL Services" on page 59, for the additional memory required for RIPL support on a WorkSpace On-Demand server.

- WorkSpace On-Demand clients

- See Section 4.6.2, "Memory and Hard Disk Required for Remote IPL Services" on page 59, for the additional memory required to support *each* WorkSpace On-Demand client.

- WorkSpace On-Demand client Support for Remote Swapping

See Section 4.6.2, "Memory and Hard Disk Required for Remote IPL Services" on page 59, for the additional hard disk required to support *each* WorkSpace On-Demand client.

- Network Interface Card (LAN Adapter)

- Streaming LAN Adapter(s)



We recommend the use of 32-bit Streaming LAN adapters because of their enhanced data throughput.

- Multiple Adapters per LAN Segment

See Section 4.3, “Network Topology” on page 55, for our earlier recommendation for locating WorkSpace On-Demand servers on the same physical LAN as the WorkSpace On-Demand clients that they service.

Use multiple LAN adapters *on the same segment* to provide a *load-balancing* capability, if needed.

- CPU

- Pentium 90 MHz (or compatible) or faster

See Section 4.6.1, “Processor” on page 59, for a discussion of suggested CPU specifications.

#### 4.7.2 WorkSpace On-Demand client Performance Considerations

Insuring that the WorkSpace On-Demand client is provided with sufficient memory to keep *swapping* to a minimum is the most effective means of improving performance. The next most effective means of improving performance is to provide a hard disk for use as a local swapping device.

From an end-user’s perspective, the perception of how well a given WorkSpace On-Demand client’s performs, depends, for the most part, on the responsiveness of the WorkSpace On-Demand server and the network infrastructure supporting it.

- Network Adapters

See Section 4.2, “Hardware and Network” on page 53, for the list of network adapters supported by WorkSpace On-Demand.

- Wake On LAN

Because WorkSpace On-Demand makes heavy demands on network bandwidth when it is initializing multiple WorkSpace On-Demand clients, you should consider the use of *Wake On LAN* network adapters. With this type of adapter, a system administrator can program groups of WorkSpace On-Demand clients to initialize themselves at *staggered* times prior to the start of the business day. This insures that *boot storms* are avoided by managing the initialization process and having WorkSpace On-Demand client workstations available for logon when the end-users arrive for work.

- Memory

- Workspace On-Demand client configured to support a *local* swapping device  
12 MB *minimum*
- Workspace On-Demand client configured to support a *remote* swapping device  
16 + MB *minimum*

**Note**

This parameter is application-dependent. The characteristics of the application(s) configured for a specific end-user may invalidate these recommendations. See Section 4.6.2, “Memory and Hard Disk Required for Remote IPL Services” on page 59, for more information on Workspace On-Demand client memory requirements.

- Hard Disk

Optional, but recommended for improved performance when used as a local swapping device.

**Note**

It is important to remember that the hard disk must be formatted before it can be used as a local swapping device. The choice of file system (HPFS or FAT) does not measurably affect performance. If the disks are unformatted and the Workspace On-Demand client is equipped with a floppy disk drive, you should boot the Workspace On-Demand client with a DOS image diskette and proceed to format the hard disk. For those Workspace On-Demand clients without a floppy disk drive, you should preformat the hard drive prior to installation.

- CPU

Intel or compatible 33 MHz '486 *minimum*

---

## 4.8 Capacity Planning and High-Availability for Workspace On-Demand servers

As a general rule, the issues of performance and capacity planning are closely related. Network planners routinely take actions that directly effect network and/or server performance whenever they modify the capacity parameters of a given system. For example, as the number of client requesters for a given server is increased, server responsiveness degrades accordingly.

For this reason, as well as eliminating a single-point-of-failure, network planners often configure backup domain controllers to service client logon/verification requests in the event that the primary domain controller is unavailable to service the request in a timely manner. Because the tasks of logon/verification are not particularly resource intensive, the backup domain controller function is often implemented on an additional server in the domain, for instance implementing a multi-function server as opposed to a dedicated backup domain controller. This strategy fits nicely with Workspace On-Demand's ability to support multi-function servers.

#### 4.8.1 Workspace On-Demand servers, Load-Balancing and Fail-Over Services

Just as a backup domain controller will service a requester's attempt to log in to a domain in the event that the primary domain controller is either busy or unavailable, remote logon services can be configured on an additional Workspace On-Demand server to provide the same function for RIPL clients. In this way, network designers can provide an effective means of ensuring Workspace On-Demand server responsiveness during periods of heavy demand, such as load balancing, as well as eliminating a single-point-of-failure in the network, such as a fail-over capability.

#### 4.8.2 Creating a Back-up Workspace On-Demand server

Prior to implementing this type of configuration, there are a few things that you should consider:

- The mechanism by which a Workspace On-Demand client locates a Workspace On-Demand server is by *broadcasting* a RPL FIND frame. The Workspace On-Demand server responds with a RPL FOUND frame that is *explicitly addressed* to the Workspace On-Demand client's MAC address, as are all the subsequent frames exchanged between them during the RIPL process. In this manner, a Workspace On-Demand client is serviced by the first Workspace On-Demand server to respond to its RPL FIND request.
- In the Workspace On-Demand server's RPL.MAP file, there can be one, and only one, active workstation record for any given Workspace On-Demand client's MAC address. This workstation record contains the name of the Workspace On-Demand server that will respond to the Workspace On-Demand client's RIPL FIND frame.
- The workstation record is created in the RPL.MAP file when the Workspace On-Demand client is created. This process is *server specific*, which makes it is easy to create duplicate sets of Workspace On-Demand clients for both the primary and backup Workspace On-Demand servers.

- You can use the NET RIPLMACH command with a response file to create multiple WorkSpace On-Demand clients with a single execution. This command provides the ability to specify the WorkSpace On-Demand server name so that you can run it from a WorkSpace On-Demand administrative workstation. If you run it from the console of a WorkSpace On-Demand server, the WorkSpace On-Demand server name is supplied as a default.
- You can also use the GUI to create WorkSpace On-Demand clients, in which case the WorkSpace On-Demand server name is provided by the WorkSpace On-Demand server icon from which you execute the *drag and drop* that creates the WorkSpace On-Demand client.

**Attention!!**

As you will learn later in this book, each WorkSpace On-Demand client's machine name (*Computername* from the IBMLAN.INI file) is also used as a *Username*. In this way, a WorkSpace On-Demand client can log onto the WorkSpace On-Demand server and get access to specific files during the RIPL process, prior to the end-user logging on. Because user names are stored in the DCDB, the machine names on each server **MUST BE UNIQUE**. For example, the primary WorkSpace On-Demand server could utilize a naming strategy such as *MACH1P*, *MACH2P*, *MACH3P*....., and the standby or backup WorkSpace On-Demand server could refer to the same machines as *MACH1B*, *MACH2B*, *MACH3B*..... Keep in mind that these names are not the actual end-user user names that are also stored in the DCDB.

Using this type of strategy, you can provide both a load-balancing and fail-over capability for your WorkSpace On-Demand clients. You can also take advantage of WorkSpace On-Demand's multi-function server support to combine these capabilities into a reduced number of physical machines that provide the benefits of load-balancing and fail-over support in a cost-effective manner for smaller or remote locations.

---

## Chapter 5. Installing WorkSpace On-Demand

This chapter covers the installation of WorkSpace On-Demand Manager and WorkSpace On-Demand Administration Client on machines that meet the software prerequisites mentioned in Section 4.1, "Software Prerequisites" on page 51.

The WorkSpace On-Demand installation can be performed as an attended install or as an unattended install.

- An attended install uses a graphical user interface (GUI) where the installer is prompted for required information and options during the installation.
- An unattended install uses the configuration, installation, and distribution (CID) architecture. After invoking an install command, a response file provides keywords and values so that user intervention is not required.

As installer, you can choose to install all of the WorkSpace On-Demand components or a smaller subset of components as part of a selective install. In a selective install, the installer is allowed to choose the specific WorkSpace On-Demand components to be installed on the server.

---

### 5.1 Before You Install WorkSpace On-Demand

The README file located in the root directory of the WorkSpace On-Demand installation CD-ROM contains the latest information regarding the installation process. Unless otherwise noted in this chapter, consider the information in the README file to override the guidance contained in this chapter. You should read the README file before starting your installation process.

The order in which you install OS/2 Warp Server and WorkSpace On-Demand is important. Following is the order in which products should be installed on a new server.

1. Install OS/2 Warp Server.
2. Install RIPL support.
3. Run RIPLINST for the versions of OS/2 that you want to support in addition the WorkSpace On-Demand from this server.
4. Run GETRPL.

**Note**

When installing multiple versions of OS/2, GETRPL should be run after each execution of RIPLINST.

5. Install LCCM.
6. Install FixPaks
7. Install WorkSpace On-Demand.

**FixPaks**

FixPaks that are included on the WorkSpace On-Demand CD-ROM and are installed with the WorkSpace On-Demand installation include:

- LAN FixPaks:
  - IPx8405 - for OS/2 Warp Version 4
  - IPx8506 - for OS/2 Warp Server and OS/2 Warp Server SMP
- MPTS FixPaks
  - WRx8503 - OS/2 Warp Server SMP
  - WRx8421 - OS/2 Warp Server and OS/2 Warp Version 4

For your convenience, FixPak 29 for OS/2 Warp Server is contained on the WorkSpace On-Demand CD-ROM, but is not installed as part of the WorkSpace On-Demand installation process.

If you are installing WorkSpace On-Demand on an existing server, you will need to start in the middle of the steps above and complete them in order. For example, if you have OS/2 Warp Server with RIPL support installed, you can begin with step 6.

---

## 5.2 Attended Installation of WorkSpace On-Demand Manager Components

This section describes the attended installation process for the WorkSpace On-Demand Manager components. The attended installation takes place in two phases.

You only need to install the WorkSpace On-Demand Manager code on a server from which the WorkSpace On-Demand clients are going to be booted. Under certain circumstances, this might be a multi-function server as described in 2.3.2.2, "Multi-Function Server Support" on page 16.

#### **Backup your system**

It is strongly recommended that the target machine be backed up before installing any WorkSpace On-Demand Manager components.

The WorkSpace On-Demand Manager installation process installs the WorkSpace On-Demand components and FixPaks. The uninstall process will remove the WorkSpace On-Demand components, but it does not remove the FixPak. The only way to return the server to the state it was in before WorkSpace On-Demand was installed is to restore the server from a backup.

#### **Attention**

If you install OS/2 Warp Server components after installing WorkSpace On-Demand, you must reinstall WorkSpace On-Demand because both products share some common code. WorkSpace On-Demand will be overwritten by older OS/2 Warp Server code, and there may be unpredictable results.

### **5.2.1 First Installation Phase**

The WorkSpace On-Demand CD-ROM includes Feature Install, Netscape Navigator 2.02, Java 1.02, and all the necessary FixPaks. These components are required to perform the phase 1 portion of the install and are automatically installed as part of the WorkSpace On-Demand installation if not already present and at the current levels.

Feature Install is installed in the \FIRUN directory on the boot drive. Netscape Navigator will be installed in an administrator-specified directory that is chosen during the install. Java will be installed on an administrator-specified drive that is chosen during the install.

### Feature Install

IBM OS/2 Feature Install is a browser-based installation framework that is used to install some IBM Software Choice features. Feature Install provides OS/2 applications with a common set of installation functions necessary for most installation programs. The Feature Install uses Netscape Navigator for OS/2 and the Netscape Navigator for OS/2 Plugin APIs to facilitate a common look and feel for feature installations.

In addition to enhancing the consistency of the user interface among installation programs, Feature Install supports important administrative installation tasks including the distributed installation of features through a response file and the ability to install a feature onto a RIPL server. To find out more about Feature Install, check out the IBM Software Choice Web page ([http://service.boulder.ibm.com/asd-bin/doc/en\\_us/home.htm](http://service.boulder.ibm.com/asd-bin/doc/en_us/home.htm)).

Before you begin the install, you should stop all network services. This can be done by issuing the NET STOP REQ command. If you forget to stop network services, the install program will stop them automatically for you.

From an OS/2 command line of an OS/2 window or full screen session, enter the drive letter where the WorkSpace On-Demand CD is located, type `install` and press **ENTER**. For example, if the WorkSpace On-Demand CD-ROM on your system is in drive G, enter `G:install`. An IBM WorkSpace On-Demand installation screen will be displayed as shown in Figure 13 on page 75. You navigate through the installation screens by clicking the forward and backward arrows at the bottom of the screen.



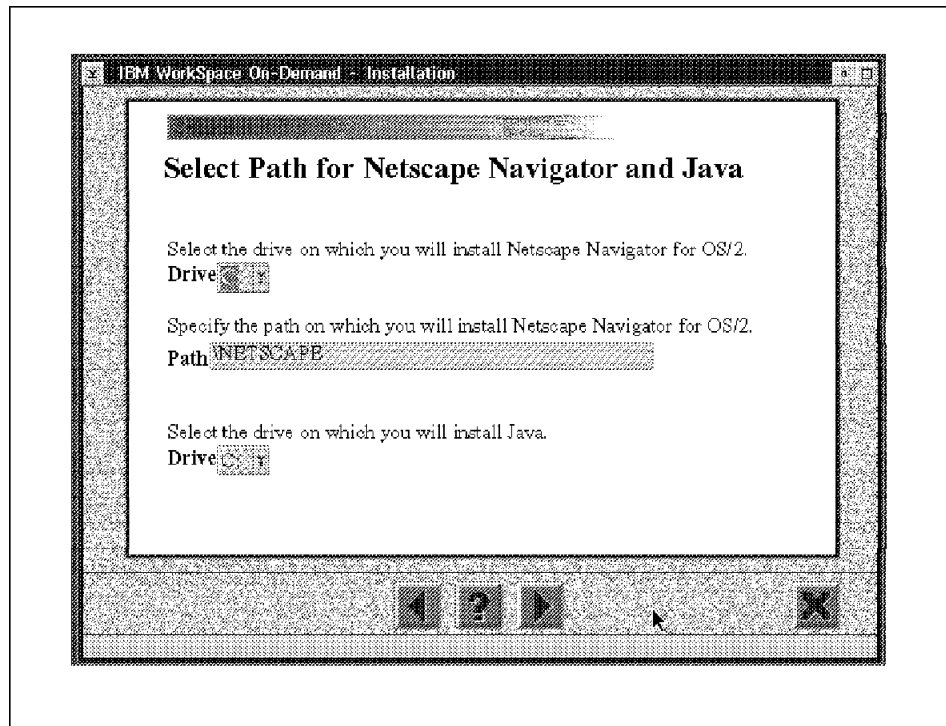


Figure 13. Select Location for Netscape and Java Files

You must choose where to install the Netscape Navigator and Java files. By default, they will be installed on the boot drive in the \NETSCAPE and \JAVAOS2 subdirectories. If you choose to install these products on another drive, make your selection in the fields provided. When you have made your selections, click the forward arrow at the bottom of the screen, and the first phase of the installation will begin to copy files to your hard drive. You can click on the ? button to get help with using these screens.

When the files have been copied to your hard drive, you will be presented with a restart button. Click on the **Restart** button, and the server shuts down and reboots automatically. This completes the first phase of the install.

### 5.2.2 Second Installation Phase

After rebooting the machine, the second phase of the install process starts automatically.

The second phase uses Feature Install through Netscape Navigator to install the WorkSpace On-Demand components that you select. The first panel to be presented is the Netscape - IBM WorkSpace On-Demand Select

Components panel as shown in Figure 14 on page 76. The selectable components are:

- Command Line Interface and API Upgrade
- RIPL Services Upgrade
- Base Operating System

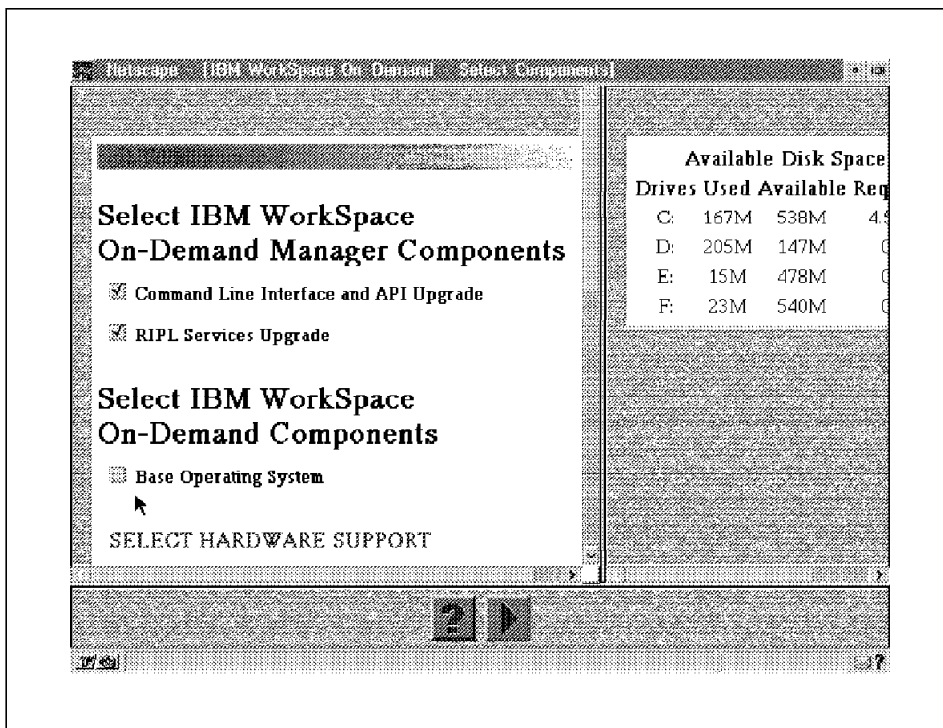


Figure 14. Select WorkSpace On-Demand Components

Using the Netscape - IBM WorkSpace On-Demand Select Component screen, select the features you need on your machine. In most installations, you will want to install all three of these components. Following is a description of each of the installable components:

1. *WorkSpace On-Demand Command Line Interfaces (CLI) and API Upgrade*

Select this option to perform WorkSpace On-Demand functions from a command-line interface and to upgrade the command-line interface (CLI) and the application programming interface (API).

CLI allows you to modify parameters from a command line. Use the command-line interface instead of the GUI to perform LAN Server functions if you are:

- Familiar with IBM networks and the OS/2 LAN Server commands
- Comfortable typing commands without the help information provided by the graphical user interface
- Want to create command files to automate network functions

The API is a programming interface that allows programmers to establish and modify applications. With the right combination of APIs, your application can use almost the entire OS/2 operating system and library of LAN Server functions.

**Note**

Upgrade of the CLI and API is required to enhance both the WorkSpace On-Demand Manager RIPL Services and the WorkSpace On-Demand Manager LAN Server Administration Tool.

## 2. *WorkSpace On-Demand LAN Server Administration Tool Upgrade*

Select the LAN Server Administration Tool Upgrade option if you want to upgrade your previously installed LAN Server Administration Tool.

**Note**

This tool has as a prerequisite the Command Line Interface. You must upgrade the Command Line Interface before you can upgrade this tool.

## 3. *WorkSpace On-Demand RIPL Services Upgrade*

Select the RIPL Services Upgrade option if you want the Remote Initial Program Load Services upgraded on your system. You must have previously installed versions of the RIPL services before you can select this option. If it is not listed as a component that can be upgraded, you must first install it using OS/2 Warp Server Install.

**Note**

This tool has as a prerequisite the Command Line Interface. You must upgrade the Command Line Interface before you can upgrade this tool.

When you have made your selections on the first part of the component selection list, use the slide bar to move up and down to see the complete list. Figure 15 on page 78 shows a second part the Netscape - IBM WorkSpace On-Demand Select Components screen. Here you continue to select the client components you want to install. Since you do not see all of the available selections possible on one panel, be sure

to scroll through all selections using the scroll bar for the left portion of the window.

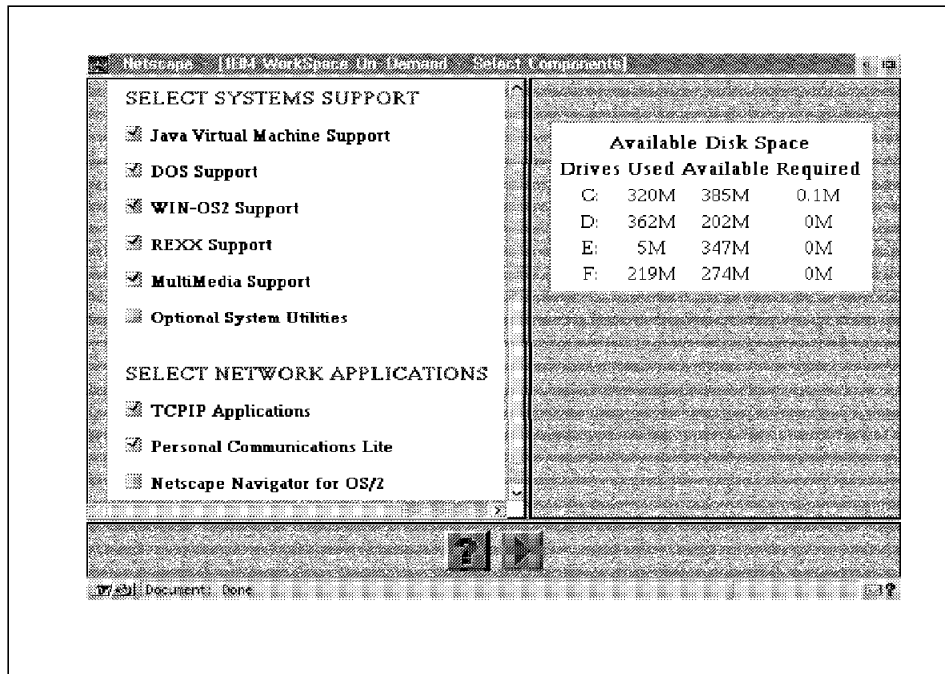


Figure 15. WorkSpace On-Demand Client Components

Following is a description of the other selectable components shown on the Netscape - IBM WorkSpace On-Demand Select Components screen. In most installations, you will want to install all of these components.

#### 4. WorkSpace On-Demand Client Operating System

Select **WorkSpace On-Demand Client Operating System** to install the operating system on your hard disk. The WorkSpace On-Demand Client Operating System contains a subset of the OS/2 Warp 4.0 operating system, MPTS, file and print, and TCP/IP support. This option is required to start the client operating system. If you have previously installed the WorkSpace On-Demand Client Operating System, unselect this option if you only need to install some of the other specific components. You can provide an alternative target drive and path when you don't want to stay with OS/2 Warp Server standards.

#### 5. Predefined Machine Classes

Select **Predefined Machine Classes** to install definitions for specific, predefined hardware. A Machine Class definition contains the supported device drivers and configuration updates needed for selected

hardware configurations. IBM selected several system types and has already preconfigured them. To install these Machine Classes, you must make the selection here. If this option is deselected, only a standard ISA, VGA Machine Class is available with which to configure clients.

**Note**

When selecting Predefined Machine Classes on an initial install or a subsequent install, make sure you also select RIPL Services Upgrade. A Feature Install user exit is run when RIPL Services Upgrade is selected that sets up Machine Class files. If this exit is not run then the administrator will not be able to define WorkSpace On-Demand clients correctly.

6. *Printer Driver Support*

Select **Printer Driver Support** to install all printer drivers on your hard disk. The drivers are installed to the \ibmlan\rpl\bb10.us\os2\drivers directory. The printer drivers are grouped by families of printer driver types. For example, there are drivers for PostScript (pscript) and pcl5 (ibmpcl5). You can manually remove printer driver families that are not needed for your environment. When the Printer Driver Support option is deselected, no printers are copied to your hard disk.

7. *Java Virtual Machine Support*

Select **Java Virtual Machine Support** to install the Java Virtual Machine to your hard disk. This provides the support needed to run Java applications and Java applets on the client.

8. *DOS Support*

If you want to run DOS programs on your WorkSpace On-Demand client, select **DOS Support**. If this option is not installed, the utilities specific to the DOS environment are not installed.

9. *WIN-OS2 Support*

If you want to run 16-bit Windows programs on your WorkSpace On-Demand client, select **WIN-OS2 Support**. If you are going to install WIN-OS2 support, DOS support is required.

10. *REXX Support*

If you want to run REXX programs on your WorkSpace On-Demand clients, select **REXX Support**.

11. *Multimedia Software Support*

If you want to install the files that provide multimedia support, select **Multimedia Software Support**. These enable multimedia capabilities such as Ultimotion video and sound.

#### 12. *Optional System Support*

To install system utilities onto your hard disk, select **Optional System Utilities**.

The system utilities included are:

- xcopy
- attrib
- format
- find
- label
- ansi
- boot
- more
- mode
- tree
- undelete
- pstat
- diskcopy
- fdisk
- setboot
- bldlevel

#### 13. *TCP/IP Applications*

Select **TCP/IP Applications** to install base TCP/IP applications on your hard disk. TCP/IP applications include FTP, Telnet, FTPD, Telnetd, and other management applications and TCP/IP servers. The TCP/IP protocol is included in the WorkSpace On-Demand Client Operating System; depending on the requirements of your applications, the TCP/IP Applications option may not be needed.

#### 14. *Personal Communications Lite*

Select **Personal Communications** to provide entry-level TCP/IP 3270/5250 emulation support on the WorkSpace On-Demand client.

#### 15. *Netscape Navigator*

Select **Netscape Navigator for OS/2** to provide Web browser support.

After selecting all the components you want to install on this server, you initiate the install process by selecting the **Install** button. Feature Install will proceed to copy the WorkSpace On-Demand Manager files associated with the components you have selected from the WorkSpace On-Demand install CD-ROM to the server hard drive.

When the second phase of the attended installation completes, Feature Install will start the Axtive Registration Tool. This tool allows you to complete the registration of this install and license of WorkSpace On-Demand with IBM. As the installer, you must choose from several different registration methods at this time or defer registration until later.

#### Postponed Registration

If you choose to defer the registration until later, then you will need to do the following on the OS/2 Warp Server machine where WorkSpace On-Demand was installed to complete the registration:

1. Go to the **OS/2 Warp Server Desktop**.
2. Open the **OS/2 System** folder.
3. Open the **System Setup** folder.
4. Click on the **WorkSpace On-Demand Registration** icon.
5. Provide the necessary information.
6. Select the registration method desired.

The registration portion of the install begins with the window shown in Figure 16 on page 82. You will proceed to several more screens where you will provide your name, company, job title, address, and phone number. Each screen contains the information you provided previously, and you may go back at any point to modify your entries.

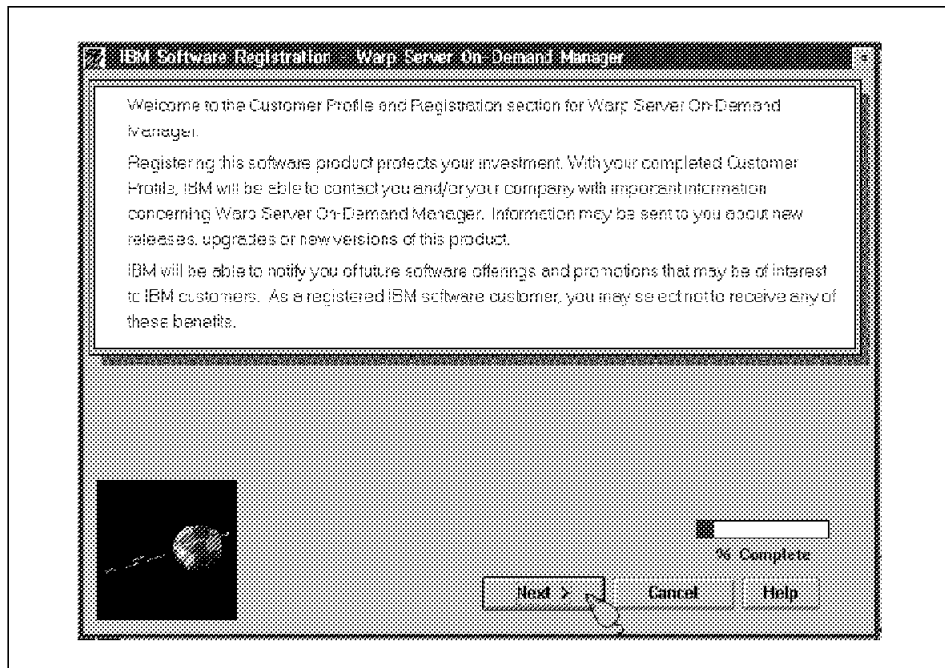


Figure 16. IBM Software Registration - WorkSpace On-Demand

After all the fields are filled in, you can choose one of several ways to submit your registration, as shown in Figure 17 on page 83. Select the radio button that is most convenient for you. The options are:

- **Modem** -- You must select the COM port for your modem and any dial prefix that may be required to complete the call, such as '9' to reach an outside line. Click **Dial Now** and the registration will be sent to IBM once a successful connection is made.
- **Fax** -- This option prints your customer profile formatted for you to fax directly to IBM. A registration number will be faxed back to you at the fax number you provide on the fax form.
- **Mail** -- This option prints your customer profile so that you may send it through the mail. The IBM address is included on this printout.
- **Telephone** -- Information is displayed on the screen, and you will give this information to an IBM operator when you call the 1-800 number listed on the screen.
- **Internet** -- The program will send the registration information to the IBM Registration Center through the Internet. Just click on **connect** to make the connection.



- **Finish Later** -- Click **OK** in the dialog box, and you can register the product at a later date.

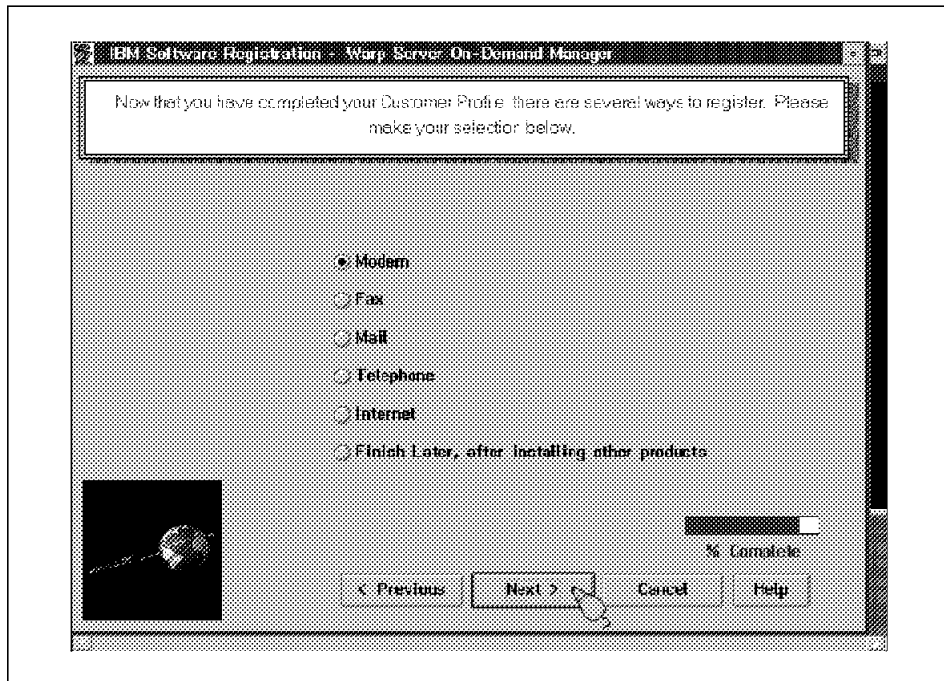


Figure 17. Select Method of Submitting Registration

When you have completed the registration activity you choose to perform, you must shut down and reboot the server machine before continuing with the next step of the install as described in 5.2.3, "Post Installation Procedures" on page 84.

#### What if there are problems?

If you encounter any problems during install, check the following logs:

- C:\OS2\INSTALL\WSODHST.LOG
- C:\OS2\INSTALL\WSODERR.LOG
- C:\OS2\INSTALL\WPINSTAL.LOG

If Feature Install fails, try to rerun the INSTALL.EXE from the WorkSpace On-Demand CD-ROM. The install process is smart enough to detect where it stopped the last time, and it will take over from the previous point of failure.

### 5.2.3 Post Installation Procedures

After the WorkSpace On-Demand Manager code is installed, you must perform an additional operation to enable the server before RIPL requesters (clients) can be configured. Run the RIPL post-install utility (GETRPL.EXE) to build the client-unique default values.

GETRPL migrates RIPL files, creates and updates system INI files, and moves files to the appropriate directories on the server. The GETRPL utility must run on the WorkSpace On-Demand Manager RIPL server after the install of the WorkSpace On-Demand RIPL support to perform post-install setup.

#### Note

You must ensure that the server services are started and you are logged on the server with an ID that has administrator authority before running the GETRPL command.

To run the post-install utility, enter the following command on an OS/2 window or full-screen command prompt

```
GETRPL /I /O:BB10.nn
```

and press **Enter**.

- The /I parameter tells the server to build a new OS2.INI file for the boot code.
- The nn refers to the national language code of the WorkSpace On-Demand version you are installing.

For more information about the RIPL Post-Install Utility (GETRPL.EXE) and its optional parameters, see the online *WorkSpace On-Demand Administrator's Guide*.

---

## 5.3 Unattended Installation of WorkSpace On-Demand Manager Components

WorkSpace On-Demand can be easily installed using the CID structures in a software distribution environment. With the configuration of a response file, you will be able install WorkSpace On-Demand unattended.

There is a sample BBCID.RSP install response file provided with WorkSpace On-Demand that can be used as part of an unattended install. This file can be edited with an ASCII editor to customize the desired values prior to

running an unattended install. BBCID.RSP is located on the WorkSpace On-Demand CD-ROM in the \INSTALL directory.

The unattended install is started through the Command Line Interface using INSTALL.EXE and specifying the appropriate parameters. To manually start the installation process with a response file, type:

```
INSTALL /B:boot_drive /L1:path_log_file /L2:path_log_hst /NN:path_net_nav  
/R:path_response_file /S:source_path /X
```

With the configuration of a response file, you have the ability to deploy new WorkSpace On-Demand Manager clients to remote locations. The additional disk space needed for the distribution server is approximately 200 MB for the entire WorkSpace On-Demand Manager file tree.

You may also want to consider implementing this in a small REXX file as the example shown in Figure 18.

```
/* */  
'c:'  
'cd \build30'  
cmd = 'cmd /c wsodinst /b:c: /l1:c:\build30\bbinst.l1 '  
      '| /l2:c:\build30\bbinst.l2 ' ,  
      '| /S:g:\. ' ,  
      '| /nn:c:\netscape ' ,  
      '| /r2:c:\build30\wsodcid.rsp ' ,  
      '| /rf:c:\build30\ncinfix.rsp ' ,  
      '| /rv:c:\build30\ncinvar.rsp ' ,  
      '| /X '  
  
say "invocation string " cmd  
cmd
```

Figure 18. Unattended Install REXX File

In the example shown in Figure 18, several response files and commands are placed in a subdirectory called \BUILD30. Two log files, BBINST.L1 and BBINST.L2, are written to this subdirectory. Netscape is installed to the C:\NETSCAPE subdirectory. The install will draw its install information from three response files: WSODCID.RSP, NCINFIX.RSP, and NCINVAR.RSP.

See Section 9.5, "The Management of WorkSpace On-Demand Management Servers" on page 216, for a more in-depth look at unattended installation of WorkSpace On-Demand Manager and clients utilizing response files and CID.

---

## 5.4 Attended Installation of the WorkSpace On-Demand Administrator Client

This section describes the attended installation process for the WorkSpace On-Demand Administrator Client components.

OS/2 Warp Version 4 clients are capable of managing Warp Server domains through the LAN Server Administration icon. This is a selectable option during the installation of the OS/2 Warp Version 4 client. An administrator can remotely manage any domain (or servers in that domain) without actually sitting at the domain controller or server. This is also true for the enhancements provided by the WorkSpace On-Demand Manager. To be able to perform remote management of the WorkSpace On-Demand Manager, you need to separately add the WorkSpace On-Demand Manager support to your OS/2 Warp Version 4 client that has the LAN Server Administration support installed.

The installation of the WorkSpace On-Demand administrator client support is very similar to the installation of the WorkSpace On-Demand Manager code on the server. In fact, the very same installation program is used. The procedure is almost identical except that during the second phase of the installation (which uses Feature Install), you only select one option:

**Command Line Interface and API Upgrade.** This updates the LAN Server Administration GUI on the OS/2 Warp Version 4 client with the same enhancements as the GUI on the WorkSpace On-Demand Manager received to support WorkSpace On-Demand.

### 5.4.1 First Installation Phase

The WorkSpace On-Demand CD-ROM includes Feature Install, Netscape Navigator 2.02, Java 1.02, and all the necessary FixPaks. These components are required to perform the first phase of the install and are automatically installed as part of the WorkSpace On-Demand installation if not already present and at the current levels.

Feature Install is installed in the \FIRUN directory on the boot drive. Netscape Navigator is installed in an administrator-specified directory that is chosen during the install. Java is installed on an administrator-specified drive that is chosen during the install.

At the beginning of the installation process, LAN services are stopped automatically. Additionally, you should stop all applications running on your machine. When this first installation phase is completed, you have to reboot your system.

From an OS/2 command line of an OS/2 window or full-screen session, enter the drive letter where the WorkSpace On-Demand CD is located; type

install, and press **ENTER**. For example, if the WorkSpace On-Demand CD-ROM on your system is in drive G, enter G:install. An IBM WorkSpace On-Demand installation screen will be displayed as shown in Figure 13 on page 75. You navigate through the installation screens by clicking the forward and backward arrows at the bottom of the screen.

You must choose where to install the Netscape Navigator and Java files. By default, they will be installed on the boot drive in the \NETSCAPE and \JAVAOS2 subdirectories. If you choose to install these products on another drive, make your selection in the fields provided before proceeding. You can click on the ? button to get help with using these screens.

When the files have been copied to your hard drive, you will be presented with a restart button. Click on the **Restart** button, and the server shuts down and reboots automatically. This completes the first phase of the install.

#### 5.4.2 Second Installation Phase

The second phase uses Feature Install through Netscape Navigator to install the WorkSpace On-Demand components that you select. The first panel to be presented is the Netscape - IBM WorkSpace On-Demand Select Components panel as shown in Figure 19 on page 88. The selectable components are:

- Command Line Interface and API Upgrade
- RIPL Services Upgrade
- Base Operating System

For a description of the components, refer to 5.2, "Attended Installation of WorkSpace On-Demand Manager Components" on page 72. To install the WorkSpace On-Demand administrator client on a OS/2 Warp Version 4 client machine, the only item you need to have selected is the **Command Line Interface and API Upgrade** option.

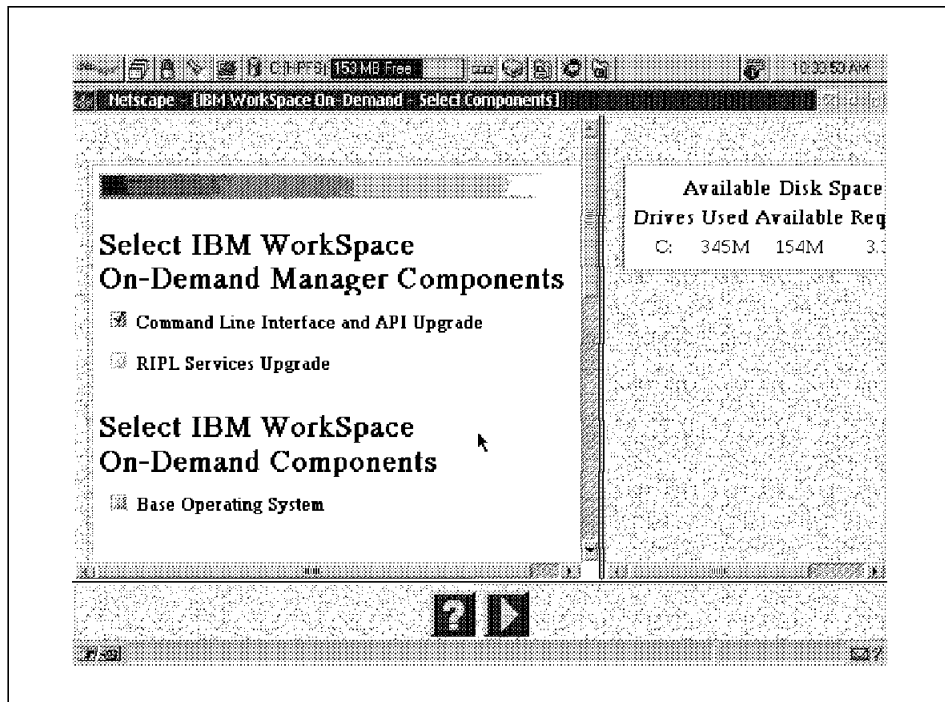


Figure 19. Select CLI and API Upgrade Option for OS/2 Warp Version 4 Client

To start the install on the client, select the **Install** button. Feature Install will proceed to copy the WorkSpace On-Demand Manager files associated with the WorkSpace On-Demand Administrator Client component from the WorkSpace On-Demand install CD-ROM to the client hard drive.

When the second phase of the attended installation completes, Feature Install will start the Active Registration Tool. This tool allows you to complete the registration of this install and license of WorkSpace On-Demand with IBM. As the installer, you must choose from several different registration methods at this time or defer registration until later.

### Postponed Registration

If you choose to defer the registration until later, then you will need to do the following on the OS/2 Warp Server machine where WorkSpace On-Demand was installed to complete the registration:

1. Go to the **OS/2 Warp Server Desktop**.
2. Open the **OS/2 System** folder.
3. Open the **System Setup** folder.
4. Click on the **WorkSpace On-Demand Registration** icon.
5. Provide the necessary information.
6. Select the registration method desired.

When you have completed the registration activity you choose to perform, you must shut down and reboot the client machine before using the WorkSpace On-Demand Administrator Client support, as shown in Figure 20, you just installed.

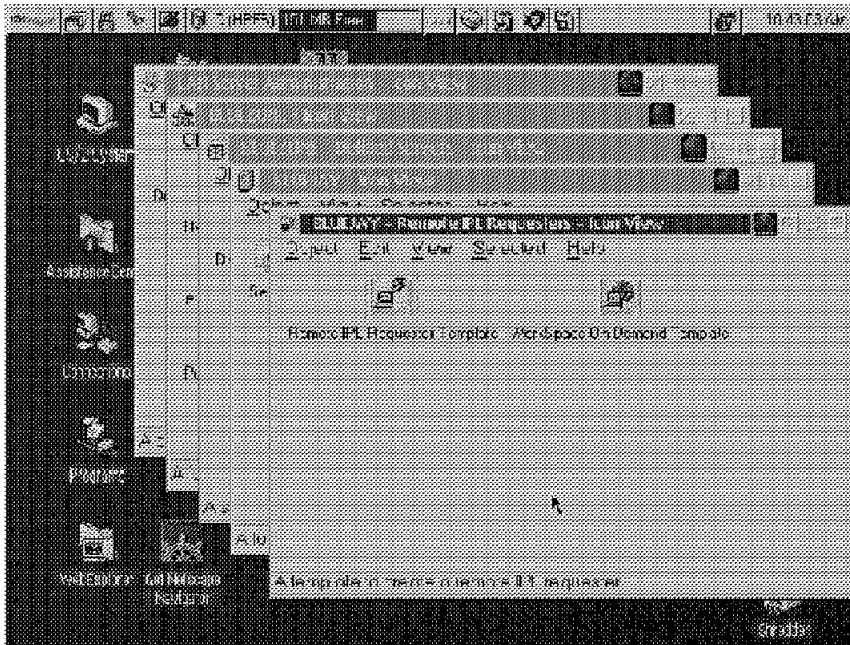


Figure 20. The Administrator GUI on an OS/2 Warp Version 4 Client

---

**Note**

If new OS/2 Warp Version 4 features are installed after the WorkSpace On-Demand Administrator Client machine is installed, you will need to reinstall the administrator client support again on the client machine to insure the WorkSpace On-Demand version of the files common to both OS/2 Warp Version 4 and WorkSpace On-Demand are installed on the client machine.

---

## 5.5 Unattended Installation of the WorkSpace On-Demand Administrator Client

The WorkSpace On-Demand Administrator Client can be easily installed using the CID structures in a software distribution environment. With the configuration of a response file, you will be able to deploy numerous new WorkSpace On-Demand Administrator Clients to your remote locations.

There is a sample BBCID.RSP install response file provided with WorkSpace On-Demand that can be used as part of an unattended install. This file can be edited with an ASCII editor to customize the desired values prior to running an unattended install for the WorkSpace On-Demand Manager as well as the WorkSpace On-Demand Administrator Client. BBCID.RSP is located on the WorkSpace On-Demand CD-ROM in the \INSTALL directory.

Section 9.5.1, "Unattended Installation of WorkSpace On-Demand" on page 217, gives an example of performing an unattended install of WorkSpace On-Demand using response files.

---

## 5.6 Uninstalling WorkSpace On-Demand

You can remove WorkSpace On-Demand from the Manager server by executing the uninstall feature. This is found in the Uninstall Features Folder, which is found in the System Setup folder, as shown in Figure 21 on page 91. Click on the **Remove WorkSpace On-Demand** icon to begin the uninstall process.



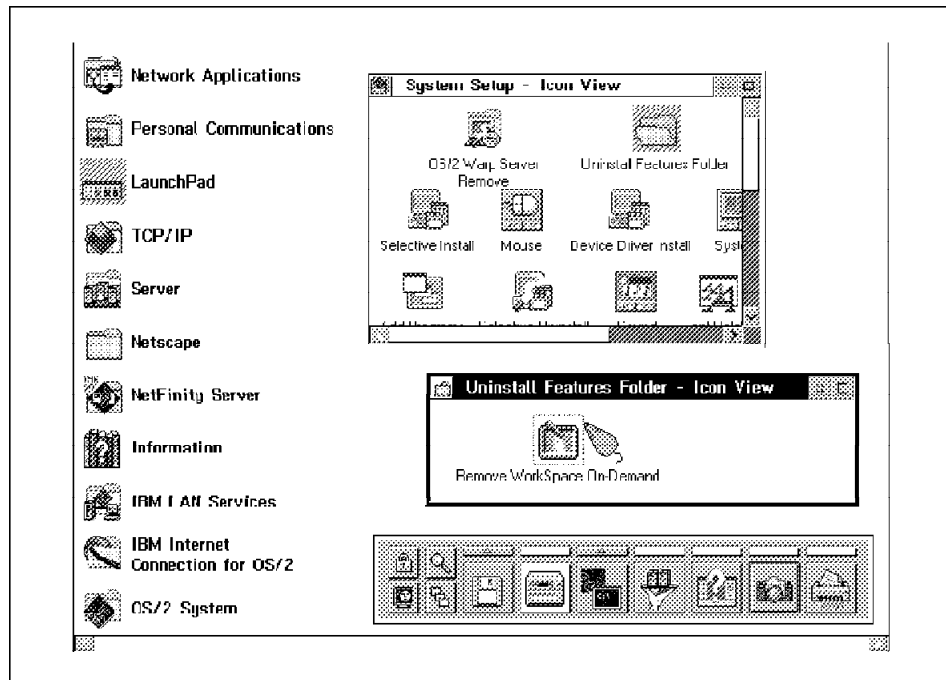


Figure 21. The Remove WorkSpace On-Demand Feature

You can completely remove the WorkSpace On-Demand product or selectively remove portions of it. As shown in Figure 22 on page 92, you can check which portions of the product you want to remove. Check all three options if you want to completely remove WorkSpace On-Demand from the server.



Figure 22. Selectively Uninstalling WorkSpace On-Demand

Click the forward arrow to advance through the uninstall. The remote boot service will be stopped, and the appropriate files will be removed from the server. An indicator will appear that shows the status of the uninstall process. If you have selected to completely remove the product, you should eventually see the window shown in Figure 23 on page 93.

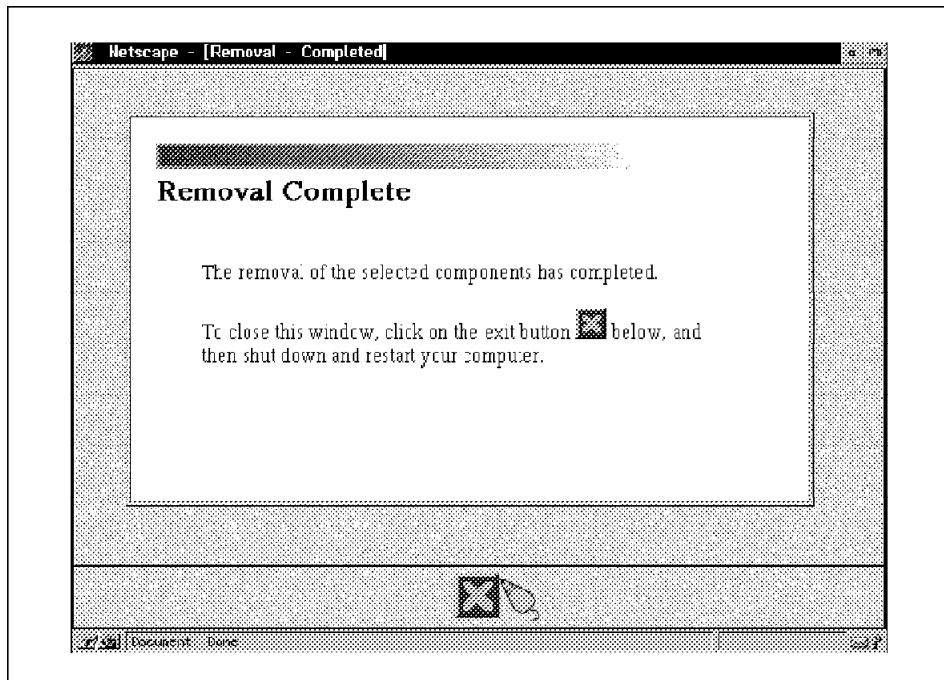


Figure 23. The Removal Complete Window

Shut down and restart the server service and insure that WorkSpace On-Demand has been removed properly. You may see remaining client definitions in the Administrator GUI that were originally created as WorkSpace On-Demand clients. Simply delete these and the system should be returned to the state it was in before you installed the WorkSpace On-Demand manager code.

**Note**

The WorkSpace On-Demand Manager installation process installs the WorkSpace On-Demand components and FixPaks. The uninstall process will remove the WorkSpace On-Demand components, but it does not remove the FixPak. The only way to return the server to the state it was in before WorkSpace On-Demand was installed is to restore the server from a backup. That is why it is strongly recommended that you back up your target machine before installing any WorkSpace On-Demand Manager components as was noted in 5.2, "Attended Installation of WorkSpace On-Demand Manager Components" on page 72.



---

## Chapter 6. Basic WorkSpace On-Demand Administration

This chapter describes setting up an initial environment. At the end of this chapter, you should be able to get the WorkSpace On-Demand client up and running with some basic applications.

Once the installation is completed, no additional software is required to enable the client.

In this chapter, the following server and workstation names are used:

<b>WSODCLT1</b>	WorkSpace On-Demand client
<b>WSODDOM</b>	Domain Controller
<b>WSODSRV1</b>	Server with WorkSpace On-Demand Installed
<b>WSODADM</b>	Administrator Workstation with WorkSpace On-Demand Tools Installed

---

### 6.1 Overview of Client Setup

Once the server software has been installed and all the installation processes are complete, you need to define the client workstation. To enable a new client with some application software, you will need to complete the following steps:

- Define the WorkSpace On-Demand client using the LAN Server Administrator's GUI.
- Define the application software using the GUI.
- Define user details, including the applications the user has access to.
- Optionally modify the end-user interface.

Each of the above steps are described in the sections that follow.

---

### 6.2 The Administration Tools

WorkSpace On-Demand provides both a command line and a graphical user interface (GUI) for administration. The GUI administration tools are integrated into the LAN Server Administration GUI. During the course of this chapter, only the GUI tools are described. The command line tools are described in Chapter 9, "Administering WorkSpace On-Demand in the Enterprise" on page 193.

The GUI tools allow the administrator to control the WorkSpace On-Demand clients as well as existing DOS and OS/2 RIPL clients. Each WorkSpace On-Demand client has specific hardware information, which is kept on the WorkSpace On-Demand server, that allows the client to initialize correctly. The administrator has the ability to display, set, and modify the client machine hardware configuration data and control applications the end-user can access.

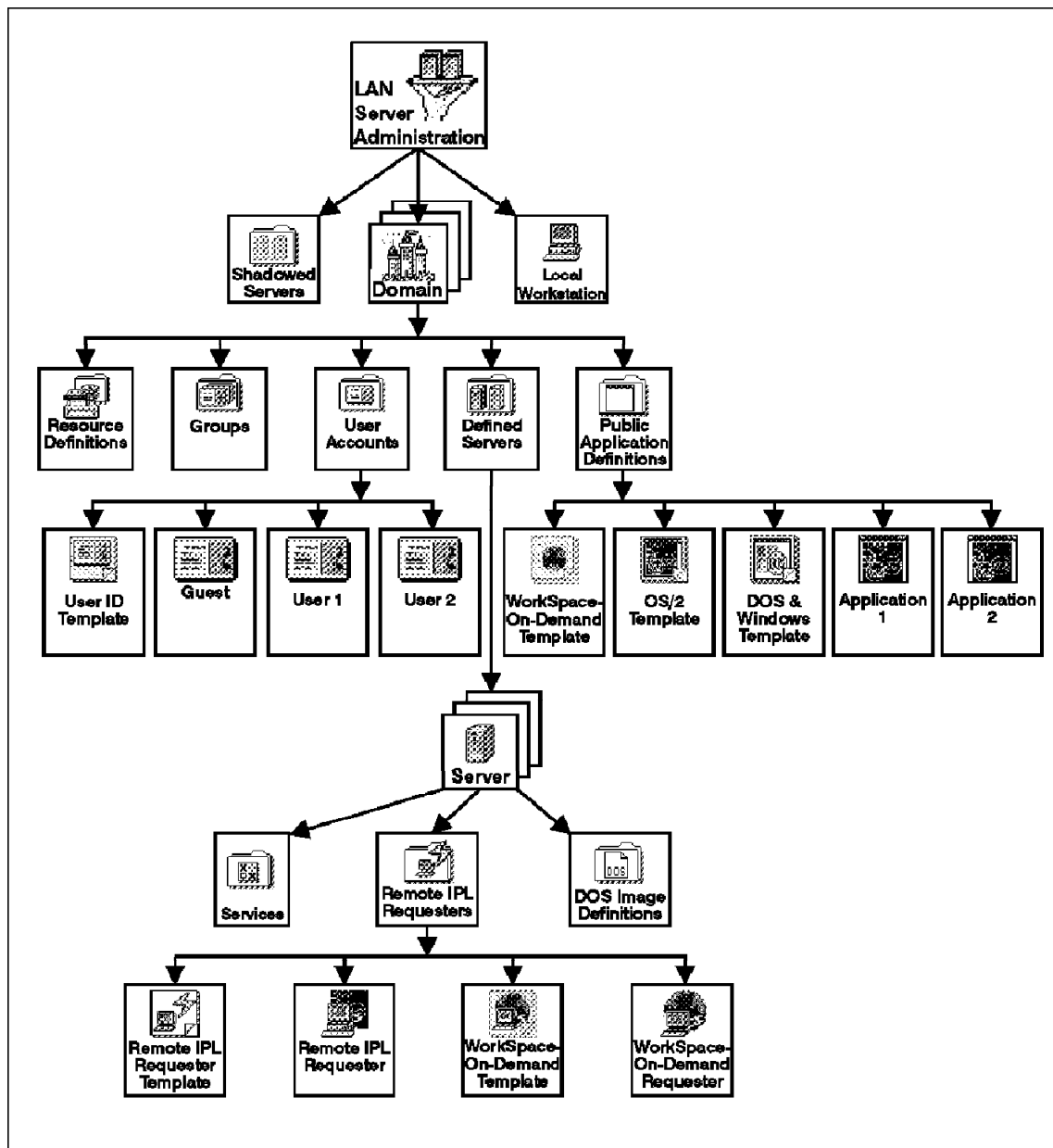


Figure 24. OS/2 Warp Server Administration GUI

The WorkSpace On-Demand Manager functions enhance the existing LAN Server Administration GUI in three functional areas to support the WorkSpace On-Demand Clients. These areas are:

- RIPL Requesters

After installation, a new WorkSpace On-Demand RIPL requester object is created in the Remote IPL requester folder. This is shown in Figure 24. This new object is similar to the existing OS/2 Warp Server Remote IPL object, but allows for more specific data to be entered to handle the WorkSpace On-Demand Client information. An administrator can create, modify, delete, or display a WorkSpace On-Demand requester.

- Applications

All applications that run on WorkSpace On-Demand clients require specific environment information and must be able to change the environment on a per-application basis. This environment information is stored in the modified Application Definition Folder. The flow to the Application GUI is shown in Figure 24 on page 97. An administrator can add, modify, delete, and display any application parameters.

- User Accounts

The user-specific parameters are modified and maintained from the User Accounts notebook. Existing fields within the user objects are used for specific WorkSpace On-Demand information. These fields include a home directory for storing WorkSpace On-Demand user-specific information and an application list, which stores all the applications an end-user can access after logon.

The flow to the user accounts is the same as before and is shown in Figure 24 on page 97.

---

## 6.3 Defining a New Client

The LAN Server Administration GUI offers an easy way to define a WorkSpace On-Demand client. At your Remote IPL server, open the **LAN Services** folder and select the folder **Local Workstation** by double-clicking its icon. Open the **Remote IPL Requesters** folder. An example of the opened folders is seen in Figure 25 on page 99.



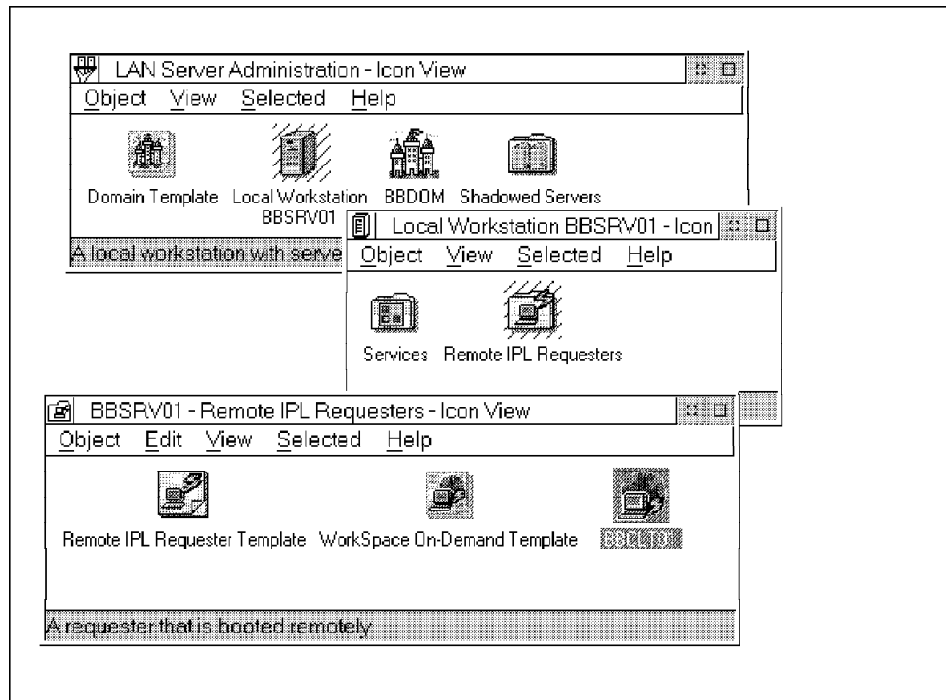


Figure 25. WorkSpace On-Demand Workstation Definition

By dragging the **WorkSpace On-Demand Template** icon to a free place and dropping the template within the same folder, you can define a new Remote IPL requester. The settings notebook is opened.

The sections that follow describe each of the settings notebook pages. Once you have completed the relevant pages, select **Set** to save the changes and close the notebook or select **Apply** to save the changes without closing the notebook.

### 6.3.1 Identity

As Figure 26 demonstrates, you must enter the Machine ID and Description (optional) to identify the client.

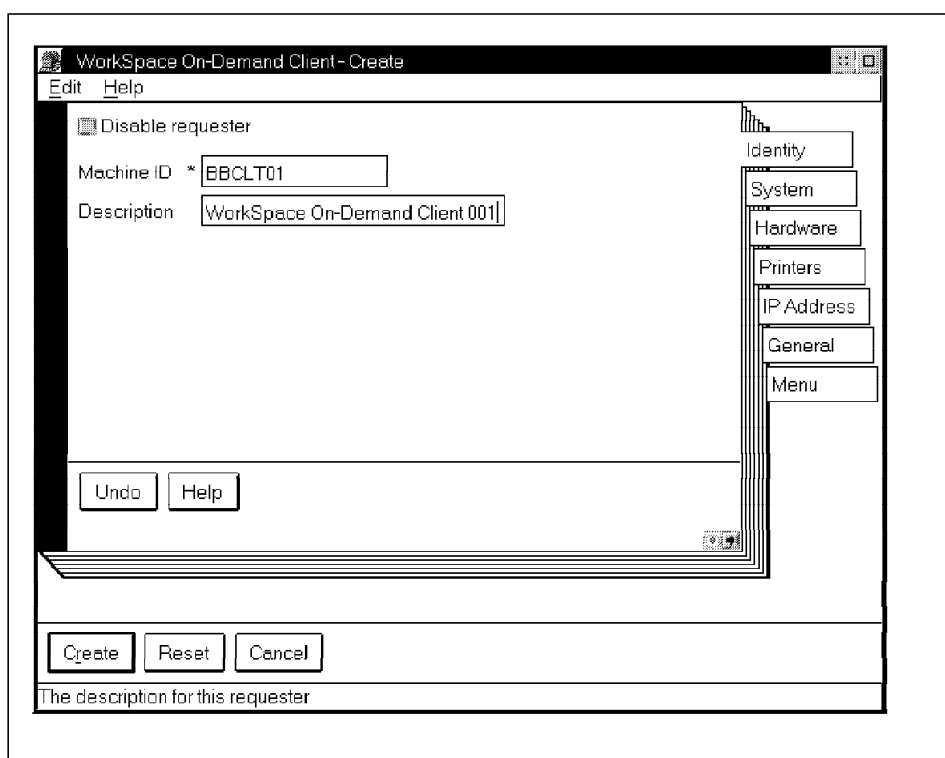


Figure 26. Identity Page of Notebook

1. You usually want to enable the Remote IPL requester when creating the requester. If you do not want the Remote IPL requester enabled, select **Disable Requester**. Disabling the requester makes it unavailable for loading from the server.
2. Complete the **Machine ID** field. The machine ID is an alphanumeric string of up to 15 characters that is used by the administrator to identify the Remote IPL requester. In our example, we have chosen WSODCLT1 as the Machine ID.
3. Complete the **Description** field. The description is an alphanumeric string of up to 48 bytes that is used by the administrator to describe the Remote IPL requester.

### 6.3.2 System

As shown in Figure 27, you must select the **Boot drive ID** and **Operating system**.

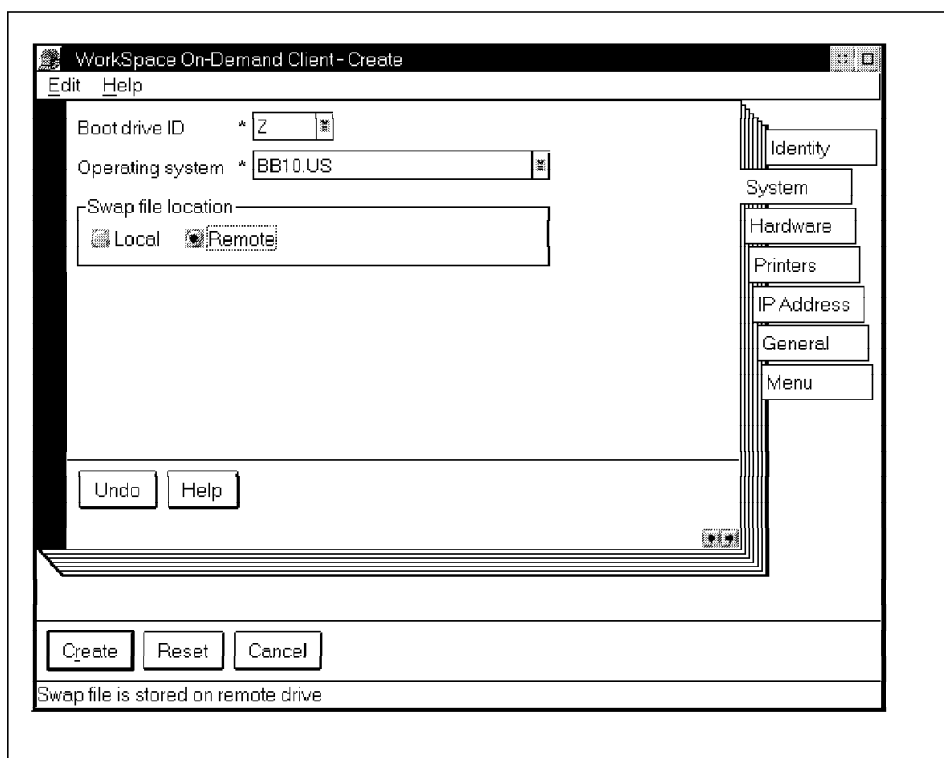


Figure 27. System Page of Client Definition Notebook

1. Select the **Boot drive ID** from the drop-down list. The default is **Z:**.

**Note**

If you select Z, you will not be able to use the "next available drive" option when assigning drives.

2. Select the **Operating system** from the drop-down list. If you have multiple language versions of WorkSpace On-Demand installed, each version will appear in the drop-down list. OS/2 and DOS will not be an option here. If you wish to define an OS/2 or a DOS workstation, use the Remote IPL requester template.
3. Choose the location for the swap file; either **Local** (on the RIPL requester) or **Remote** (on the server). If you select to swap locally, you should ensure that a local disk will be present and accessible on the

workstation. The local swapping supported by WorkSpace On-Demand is to IDE drives only.

### 6.3.3 Hardware

As shown in Figure 28, you must provide the Network adapter address and select the appropriate value for the Machine class overrides, if necessary.

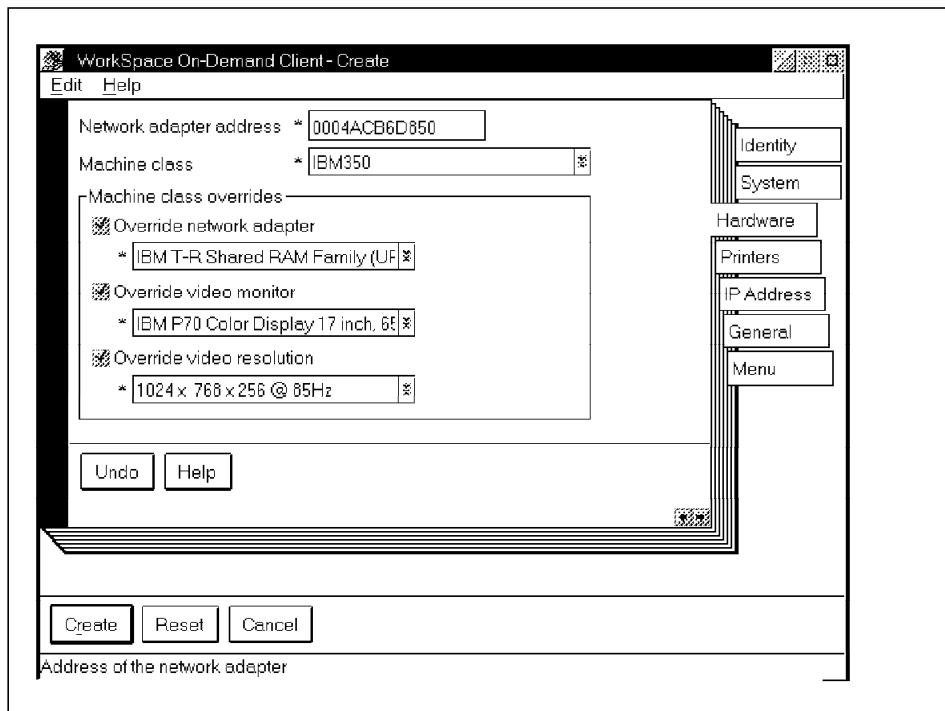


Figure 28. Hardware Page of Client Definition Notebook

1. Complete the **Network adapter** address field. The network adapter address is a unique address of the network adapter card and is obtained by turning on the Remote IPL requester machine and obtaining the 12-character string following the AA entry on the display.
2. Select the **Machine class** from the drop-down list.  
  
The Machine class is the description of the hardware environment for a particular client machine. The Machine Class that we are using in this example is for an IBM PC 350 machine.
3. The Machine class overrides group provides the ability to choose other network adapters and video selections than provided in the current Machine class definition. To override any of the default settings:

- Select the **Override network adapter checkbox**, and choose an entry from the drop-down list. In our example, we have selected the IBM T-R Shared RAM Family adapter.
- To change the video monitor, select the **Override video monitor checkbox**, and then choose an entry from the drop-down list.
- To change the video resolution, select the **Override video resolution checkbox**, and then choose an entry from the drop-down list.

**Note**

For certain Machine classes, you will be able to change the video resolution and the video monitor. The ISAVGA class does not allow you to change either.

### 6.3.4 Printers

To define a printer for the client, you must complete the information on the printer panels shown in Figure 29 on page 104. From the printer page, you can define network and local printer access for the Remote IPL requester.

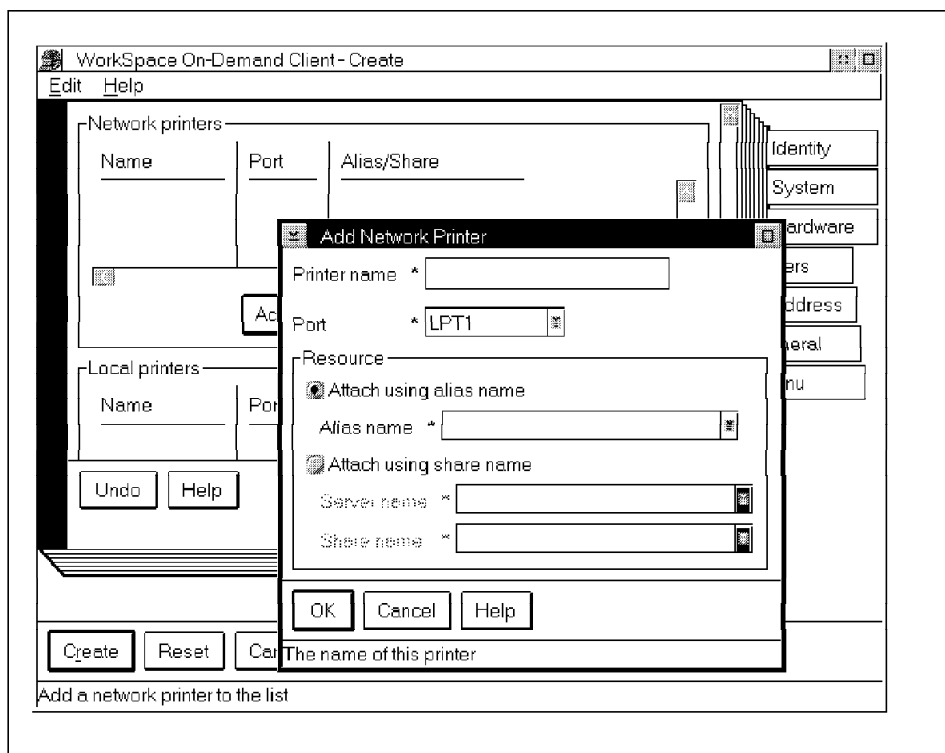


Figure 29. Printer Page of Notebook

To add a network printer, complete the following steps.

1. Select **Add**.
2. On the Add Network Printer pop-up:
  - a. Complete the **Printer name** field.
  - b. Select the desired **Port** from the drop-down list.
  - c. Select the desired **Attach using alias name** or **Attach using share name** radio button. If **Attach using alias name** is selected, select the printer **Alias name** from the drop-down listbox, or enter a valid alias name. If **Attach using share name** is selected, select the **Server name** and **Share name** from the drop-down list or enter a valid name in each field.
  - d. Click on **OK**.

If you have a local printer attached first, ensure that the parallel or serial port that the local printer is attached to is enabled. Then complete the following steps to access the local printer from the client:

1. Select **Add**.
2. On the Add Local Printer pop-up:
  - a. Complete the **Printer name** field.
  - b. Select the desired **Port** from the drop-down list.
  - c. Select the desired **Printer driver** from the list.
  - d. Select **OK**.

### 6.3.5 TCP/IP Configuration

If the WorkSpace On-Demand client will use TCP/IP, configure the TCP/IP settings through the IP Address Notebook, as shown in Figure 30.

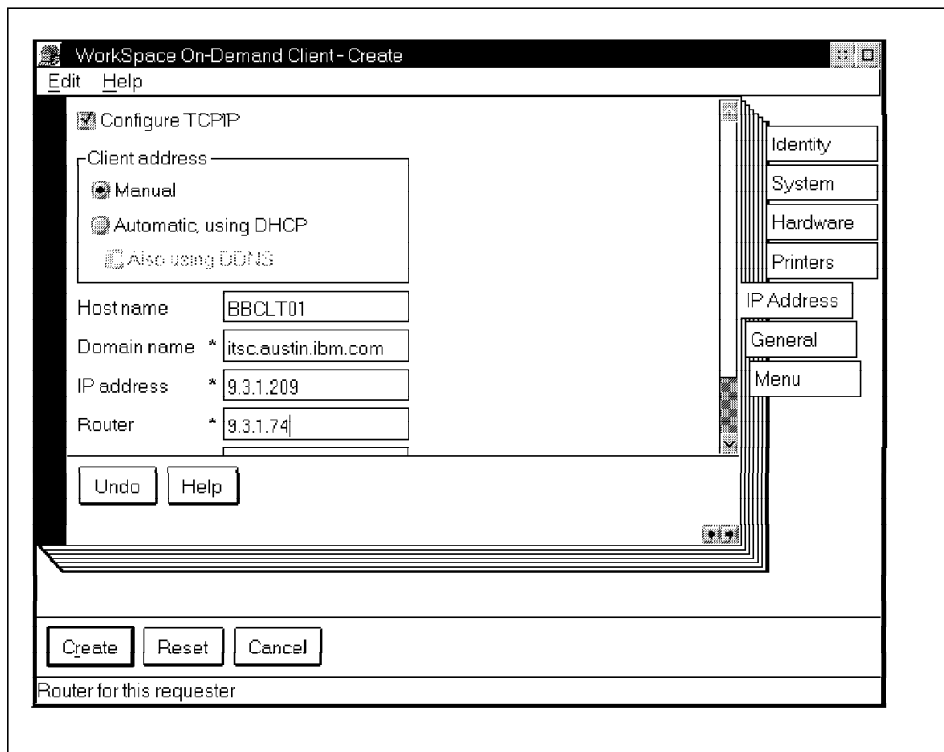


Figure 30. TCP/IP

1. Select **Configure TCP/IP** if you want to perform TCP/IP configuration. All other entries are disabled until you select this checkbox.
2. For client address, select **Automatic, using DHCP** if your network uses automatic addressing through the dynamic host configuration protocol, or select **Manual**. If automatic is selected, you may also optionally select **Also using DDNS** if you want to use Dynamic Domain Name

Services. If **Manual** is selected, you must complete the **IP address** and **Subnet mask** fields.

#### Dynamic Domain Name Server (DDNS) Clients

If you will be configuring clients to use the DDNS client function, in order to ensure the client hostname is registered with the DDNS server, one of two actions will be necessary on the part of the administrator:

1. Upgrade your DHCP server with the DHCP5ZIP FixPac (see information below about how to obtain this FixPac) to allow you to use the "DNS proxy A record update" function. This new DHCP function allows a DHCP server to add the 'A record' to the DDNS domain file on behalf of a DHCP client. The hostname entered during Workspace On-Demand client configuration will be automatically registered with the DDNS server during the client boot process. If the client hostname needs to be changed, it is done via the client configuration panels at the RPL server. Because the client hostname is essentially 'owned' by the DHCP server, the client cannot change the hostname using the DDNSCFG.EXE program, thus giving the administrator more control over the client's configuration.
2. If you are not using the "DNS proxy A record" update feature of Warp Server's DHCP server, the DDNS configuration program will need to be executed ONCE at each client workstation after the initial boot following client definition. This can be accomplished by defining the DDNSCFG.EXE program (located in the \MPTN\BIN directory of the client) as an application available to the client on the client's desktop. This program will need to be executed again if the client hostname is changed. However, if this program is left available for the client to execute, the administrator will have no control over client hostname configuration.

#### DHCP5ZIP FixPac

Implementing this function requires minor modifications to the DHCP configuration file. Refer to the readme file contained in the FixPac for further information on the function.

This FixPac is available on the Internet by going to the URL at: <http://service5.boulder.ibm.com/pspfixpk.nsf>, then selecting **APAR tracking - All Fixpaks**; then search for DHCP5ZIP.



3. Complete the **Host name** field. This field specifies the host name of the RIPL requester machine and can contain up to 32 alphanumeric characters.
4. Complete the **Domain name** field. This field specifies the domain name of the RIPL requester machine and can contain up to 40 alphanumeric characters.
5. Complete the **IP address** field. This field specifies the IP address of the RIPL requester machine and can contain an IP address of up to 15 numeric characters (for example 123.123.123.123).
6. Complete the **Router** field. This field specifies the router address of the RIPL requester machine and can contain an IP address of up to 15 numeric characters (for example 123.123.123.123).

Select the **Create** button to create the new client. After a short amount of disk activity, you should be ready to remote boot the client machine.

#### TCP/IP 4.1 and DDNS

TCP/IP Version 4.0 is shipped with WorkSpace On-Demand. See Appendix B.12, "Upgrade to TCP/IP for OS/2 Version 4.1" on page 264. It contains information on setting up TCP/IP for OS/2 Version 4.1 on your WorkSpace On-Demand clients.

See Appendix B.13, "Consideration for TCP/IP Configuration Using DDNS" on page 267 for additional information on using DDNS and TCP/IP on your WorkSpace On-Demand clients.

### 6.3.6 Workstation Definition

When you select to create the new workstation, the server does the following:

- A new record is added in the workstation records section of the RPL.MAP file. This is shown in Figure 31:

```
; workstation records
0004ACB6D850 WSODCLT1 ~ FITS\BB10CLT01 WSODSRV1 Z ~ ~ ~ ,,, ~ R_BB10_US_OTKNTR ~ ~ ~
```

Figure 31. Workstation Record Entry for a WorkSpace On-Demand Client

- A FIT file (WSODCLT1.FIT) is created for the machine in the IBMLANRPLFITS directory.
- The following two directories are created for the workstation:

IBMLANRPLMACHINESWSODCLT1  
IBMLANRPLUSERWSODCLT1

Files that are required by the machine are placed in the above directories.

- The server creates a user ID for the workstation. The server then sets access control on the above directories.
- The workstation ID has RWCXDA rights to the IBMLANRPLUSERWSODCLT1 directory.
- The workstation ID is also added to the RPLUSER group, which has RX access rights to the IBMLANRPL directory.

**Note**

If the file system of the Remote IPL server is HPFS, the workstation's name length is allowed to be up to 15 characters.

---

## 6.4 Adding Application Software

This section focuses on setting up simple OS/2 applications. Windows 3.x, Windows 95, DOS, and Java applications are discussed in Chapter 7, "Advanced WorkSpace On-Demand Administration" on page 123. For a more complete discussion of the above features, refer to *IBM Up and Running! OS/2 Warp Server*, S25H-8004, and read the chapter titled "Setting Up an OS/2 Program As an OS/2 Public Application".

Defining an application for use by a WorkSpace On-Demand client is much the same as setting up a public application with OS/2 Warp Server. A number of enhancements have been added to the public application facility. These enhancements allow more environment information to be saved with the application, thus allowing for application roaming.

Application roaming provides the user access to his/her application from any WorkSpace On-Demand workstation of the same Machine Class. Previously, although the application was installed on the server, environment information had to be added to the workstation's files. This meant that users had access to certain applications only from particular workstations. The enhancements made to the public applications allow for enhanced flexibility.

All applications have different installation and setup processes. This makes it extremely difficult to provide a standard set of instructions that works for most applications. This section is intended to explore the application setup

process, and we will therefore confine the instructions only to simple applications.

We describe setting up a few simple applications. Detailed application setup is described in chapter Chapter 7, “Advanced WorkSpace On-Demand Administration” on page 123.

In order to make applications available to end-users using WorkSpace On-Demand, the administrator must install the applications on a server. End-users cannot install their own applications. To make an application available, the administrator must:

1. Install the application.
2. Create an alias for the directory containing the application.
3. Define the application.
4. Set application parameters.
5. Assign the application to an end-user.
6. Set user-specific application parameters (optional).

#### 6.4.1 Defining a Simple OS/2 Application

The example we use here is the OS/2 PMCAMERA. This application consists of the following files:

PMCAM200.DLL  
PMCAM200.EXE  
PMCAM200.HLP  
PMCAMERA.INI

1. Decide where to install your application on the server. We created a directory on a drive which we refer to as d:OS2APPS. A second directory was created for the DLL files, d:OS2APPSDLL. The installation step for PMCAMERA is simply to XCOPY the application files to their locations:

```
xcopy PMCAM.DLL d:os2appsdll  
xcopy PMCAM200.EXE d:\os2apps  
xcopy PMCAM.HLP d:\os2apps\hlp  
xcopy PMCAM200.INI d:\os2apps
```

2. Before you actually define the public application for PMCAMERA, you must first create a directory alias that points to the subdirectory.

This is done within the Resource Definitions Folder in the LAN Server Administrator’s GUI. In the Directory Alias - Create settings notebook, complete the **Identity page**.

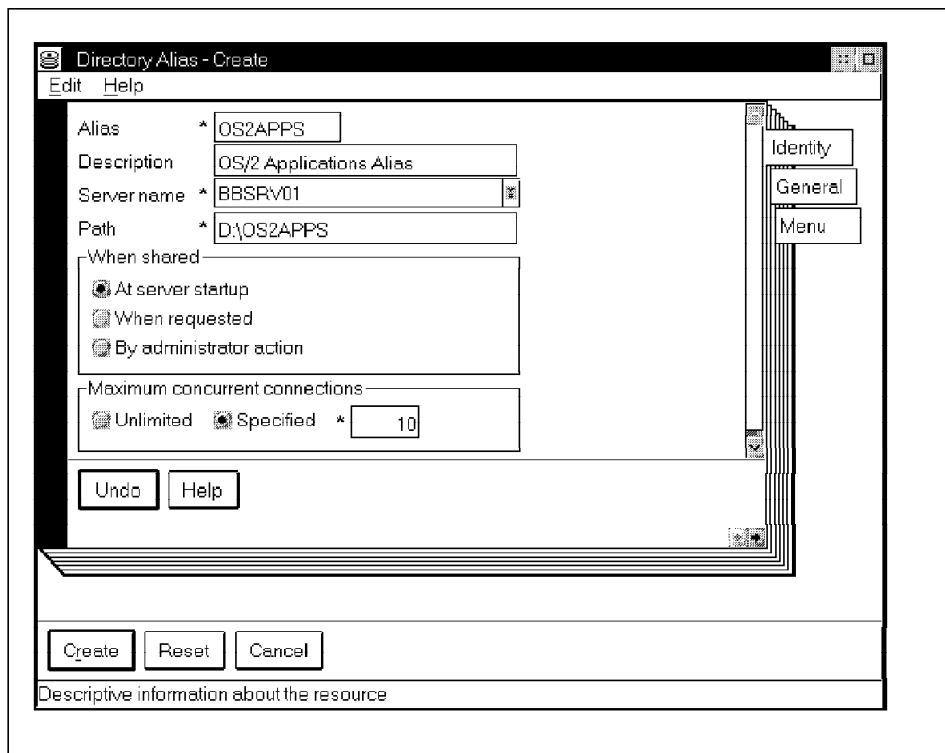


Figure 32. Directory Alias - Create Notebook

- Figure 32 shows the directory alias parameters used for the OS/2 Application Directory Alias. Select **Create**.

The following window is displayed:

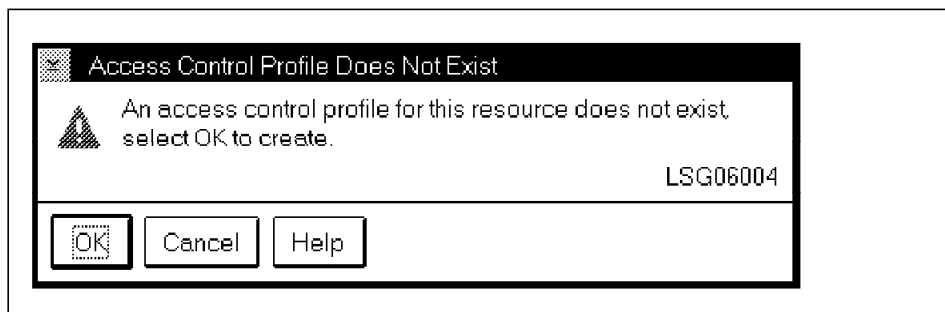


Figure 33. Access Control Profile Does Not Exist Window

- Select **OK**.

The Access Control Profile - Settings View notebook is displayed.

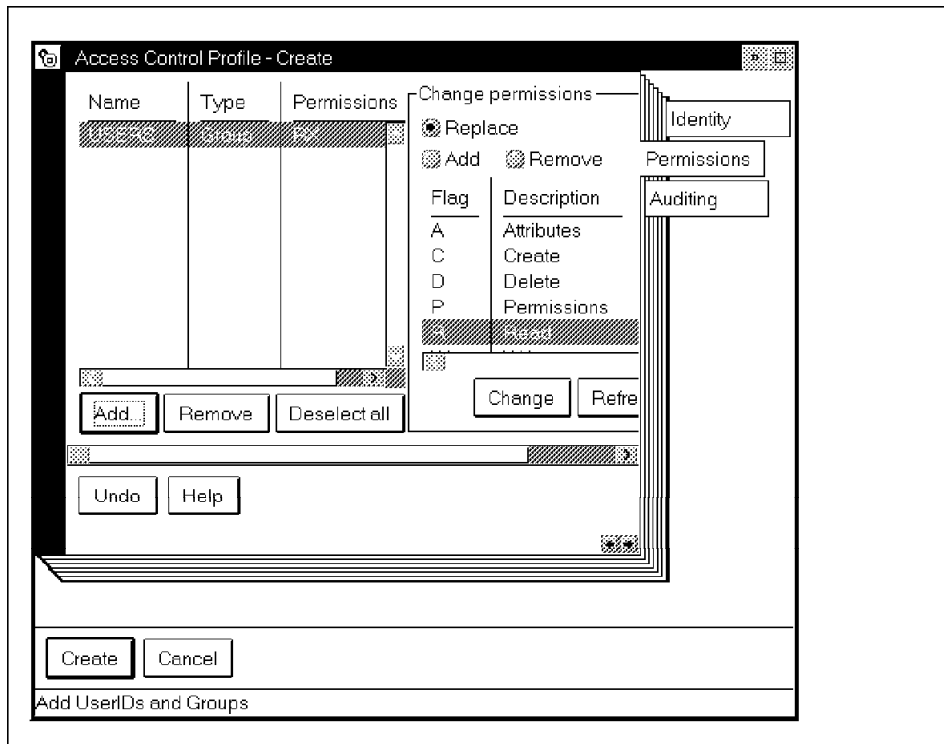


Figure 34. Access Control Profile - Settings View Notebook

5. Complete the **Permissions** page and grant all Users at least Read and Execute rights. As in previous LAN Server releases, the USERS group is a special group to which all defined users belong. Select **Create**.

The following window is displayed:

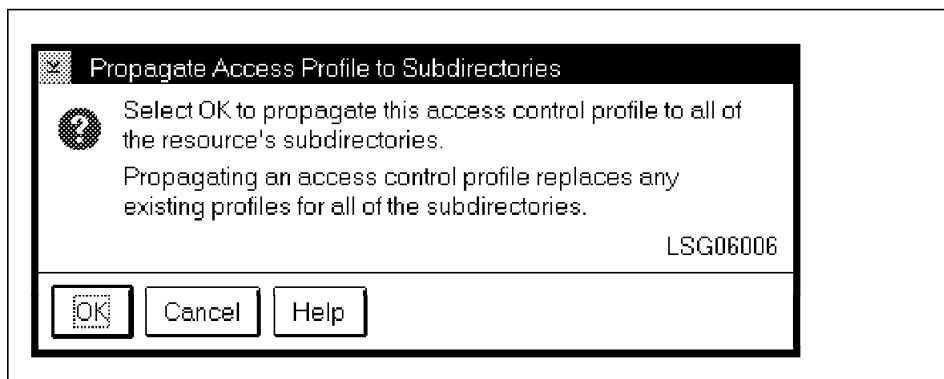


Figure 35. Propagate Access Profile to Subdirectories Window

6. Select the **OK** button.

Since you just created the file alias for OS/2 Applications, you can now continue by creating a public application (also called network application) for PMCAMERA.

7. To do so, double-click on the **Public Application Definitions** object.

The Public Applications Definitions window is opened. It is shown in Figure 36.

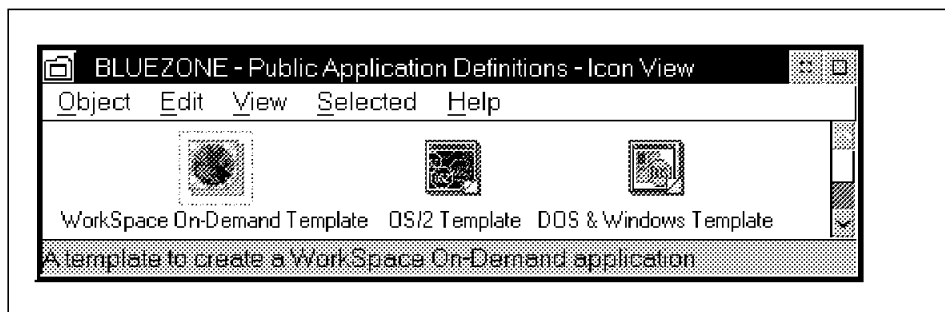


Figure 36. Public Application Definitions Folder

8. In the Public Applications Definitions folder, drag and drop a **WorkSpace On-Demand Template** to a free area of the folder.

The WorkSpace On-Demand Application Definition - Create settings notebook is displayed as shown in Figure 37 on page 113.

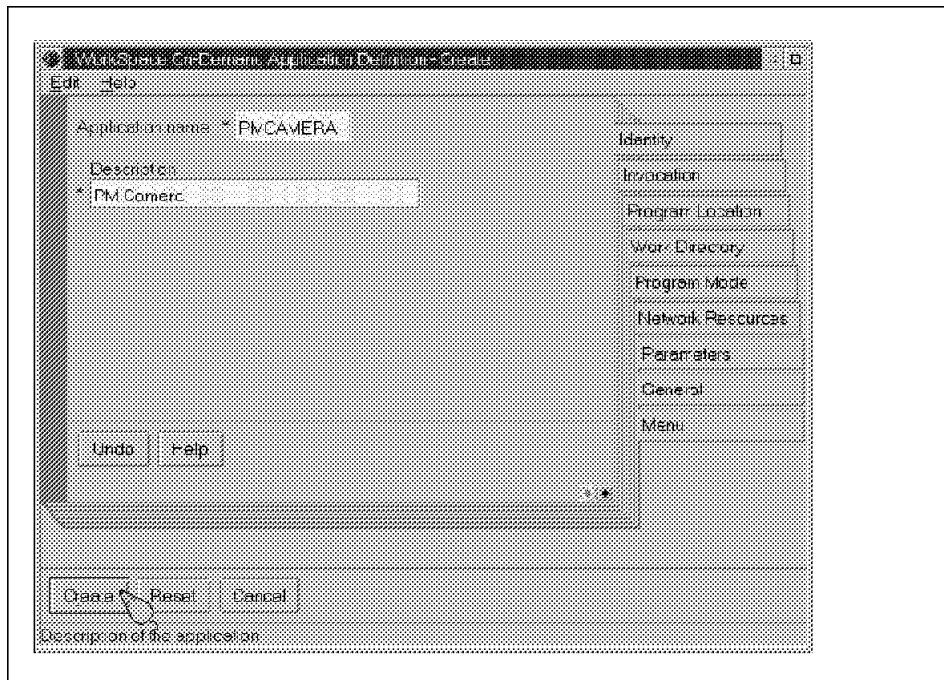


Figure 37. Workspace On-Demand Application Definition - Create Settings Notebook

9. On the Identity page, give the application a descriptive name and some descriptive text.
10. On the Invocation page, specify the command used to launch the application. In our example, it is PMCAM200.EXE. Additional parameters can also be specified, or the user could be prompted for additional parameters.
11. On the Program Location Page, you need to specify the **Alias** directory that you had defined earlier for the application. The path location to the main executable also needs to be specified.

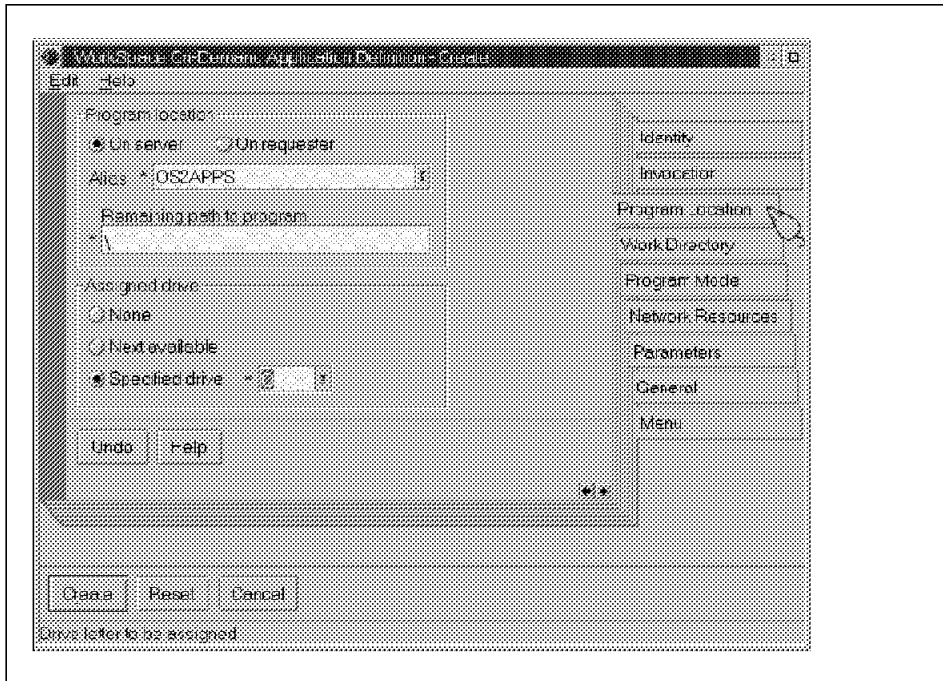


Figure 38. Workspace On-Demand Application Definition - Program Location Page

Most applications do not require a drive letter assignment to the directory alias where the program resides. The default is to not assign a drive under the Program Location tab. However, some application programs require a drive assignment and will not start correctly unless one is assigned. The error message that displays when you attempt to start the application will state that one or more files cannot be found or located. To avoid this situation, you can assign a drive letter to the directory alias as a current assignment and specify this drive.

12. Most applications also do not require a working directory to the application program path. The default is to not assign a working directory under the Work Directory tab.



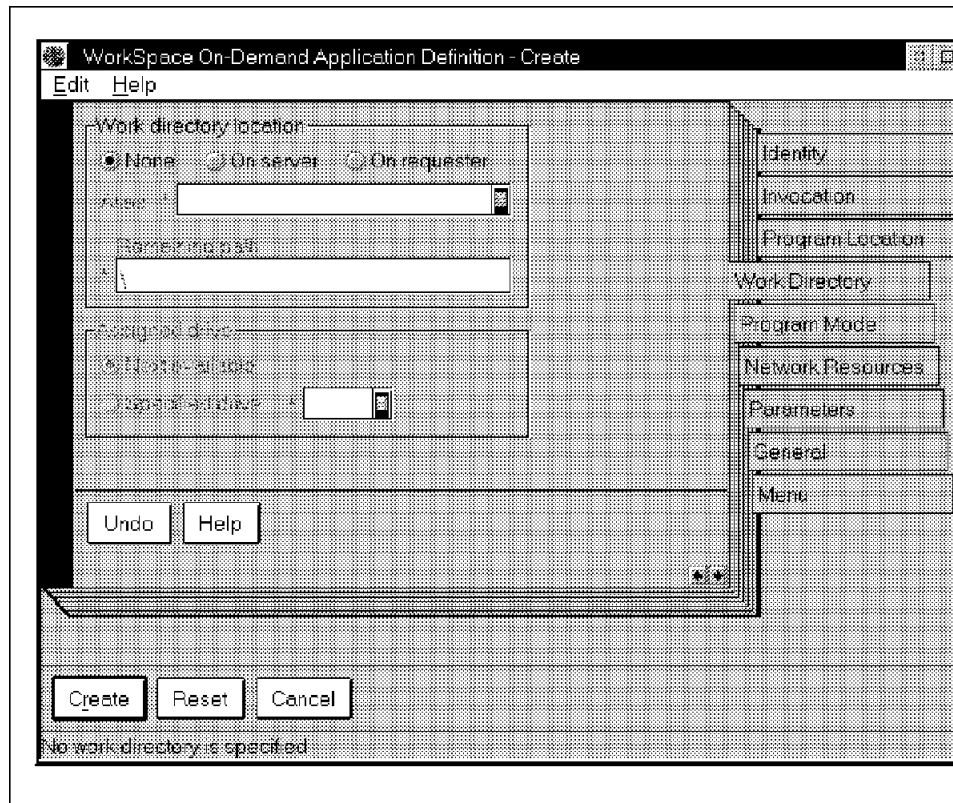


Figure 39. WorkSpace On-Demand Application Definition - Work Directory Page

However, some programs require a working directory and will not start correctly unless one is assigned. The error message that displays when you attempt to start the application will state that one or more files cannot be found or located. To avoid this situation, you can assign a working directory using the Work Directory settings page.

13. On the Program Mode Page, you need to specify what type of application you are defining. In this example, it is an OS/2 Presentation Manager Application (OS/2 PM).
14. Figure 40 on page 116 shows the Network Resources page. Using this page, you can dynamically assign network resources when this application is selected.

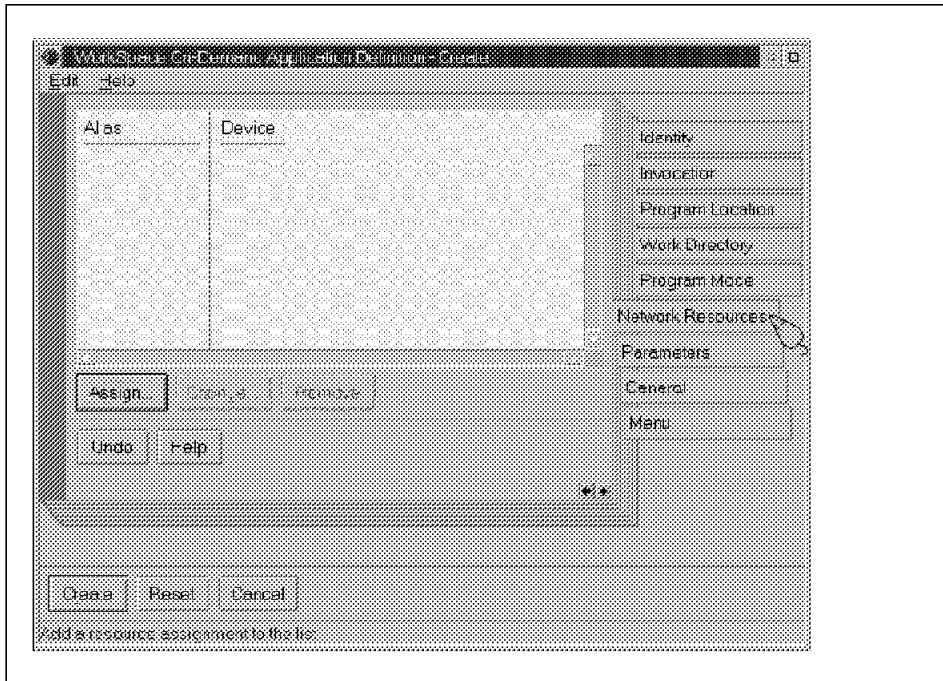


Figure 40. Workspace On-Demand Application Definition - Network Resources

15. The **Parameters** page defines how the application is started. This is shown in Figure 41 on page 117.

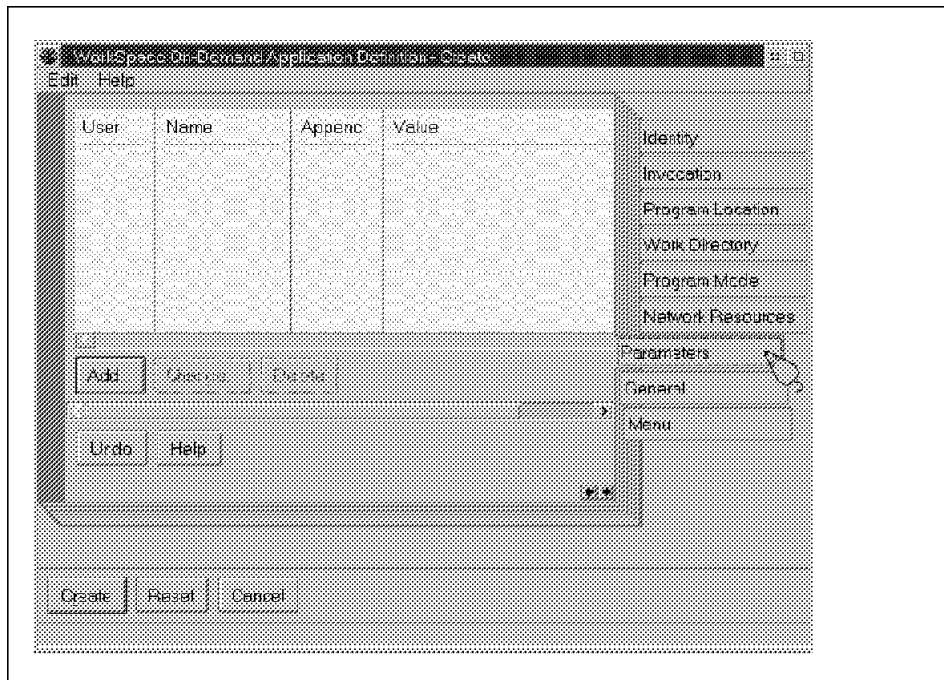


Figure 41. WorkSpace On-Demand Application Definition - Parameters

The Application Parameters page displays the environment variables including their names, the location where the value is appended to the variable, the value, and whether the parameter is user-specific. From here, you can add, change or delete parameters. You can also read a set of parameters from a file instead of manually entering them each time. The following rules apply:

- A parameter name can be used more than once if it is defined with a different append value. Parameters specified more than once that have different append values are processed in this order:
  - a. REPLACE value
  - b. BEGIN value
  - c. END value

If you repeat a parameter name and the same append value, an error message is displayed that instructs you to correct the problem before adding that application's parameter.

- Most environment variables can be specified as application parameters; however, LIBPATH is an exception. LIBPATH cannot be changed or replaced. LIBPATH can be specified with the BEGIN or END append flag to allow adding to the front or back of the existing

LIBPATH. Additionally, BEGINLIBPATH produces the same results as specifying LIBPATH with the BEGIN append flag and ENDLIBPATH produces the same results as specifying LIBPATH with the END append flag. See the *IBM OS/2 Command Reference* for more information on using LIBPATH. The online version is located in the OS/2 Base Operating System folder in the Information folder on the Desktop.

To add an application parameter:

- a. Select the **Add** push-button. The Add Application Parameters pop-up is displayed.
- b. Choose either **Configuration** to display a drop-down list of known environment names and then choose one, or choose **Application specific** and then enter an application-specific name.
- c. Choose the position for appending the environment parameters. Select **Begin** to place the parameter value at the beginning of a currently existing environment variable of the same name. Select **End** to place the parameter value at the end of a currently existing environment variable of the same name. Select **Replace** to replace an existing environment variable with the value you specify.
- d. Enter the value of the parameter in the **Value** field.
- e. Select the **User specific** checkbox if this application parameter is allowed to be a user-specific parameter. Specifying an application parameter as user-specific allows it to be defined on a per-user basis. See setting up user-specific application parameters to override this value for each end-user. If this value is not overridden for a given user, the application parameter value is used.

In the PMCAMERA example, the following parameters needed to be added:

- 1) LIBPATH, Begin, WSODSRV10S2APPSDLL
- 2) HELP, ?, WSODSRV10S2APPSHLP

#### **User settings**

For the purposes of this example, all users have a common INI file. If you wanted users to maintain their own INI file, then you would need to create a user FIT file and put an entry in it for the INI file.

- f. Select **OK**.

16. Select **Set** or **Apply**.

---

## 6.5 User Setup

After an application is installed and defined, it must then be assigned to an end-user before it can be used. This is done after an application is installed on a server and set up as a public application.

Additionally, an application can be removed from an end-user when the end-user no longer requires access to the application. The application is still available for other users. Removing the application from one end-user does not uninstall the application.

To assign an application to an end-user:

1. Open **LAN Services** from the Desktop.
2. Open **LAN Server Administration**.
3. Open the appropriate domain object.
4. Open **User Accounts**.
5. Open the object for an existing user ID.
6. Select the **Applications** page. To assign or add a public application for this end-user: Select **Add**. The Add Public Application pop-up is displayed from the menu.

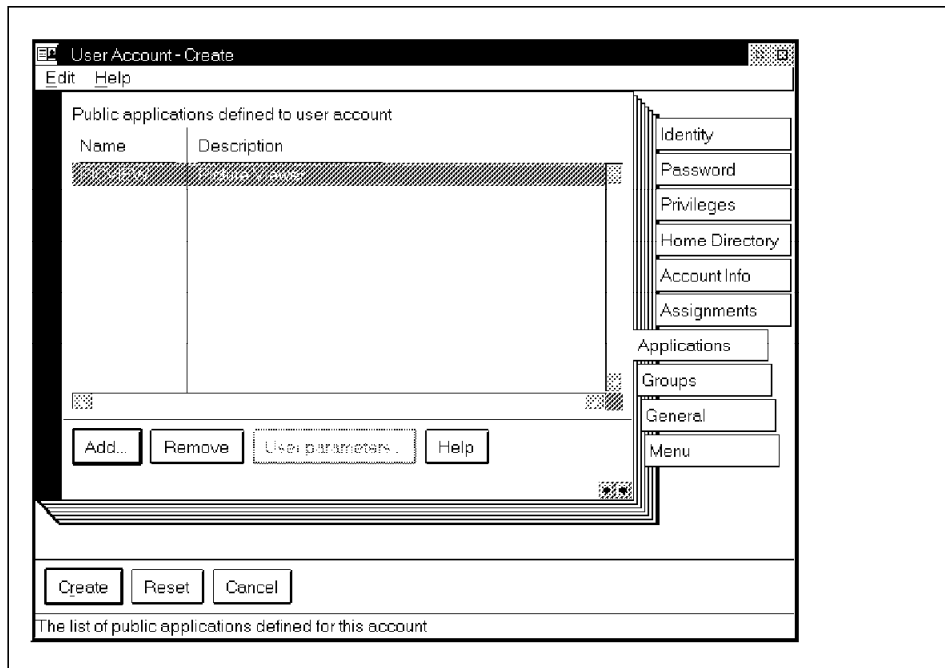


Figure 42. User Notebook - Applications

7. Select one or more applications to add for this end-user.
8. Select **Add**.

Alternatively, you can simply use the mouse to drag an application object and drop it onto the user or group to which you wish to assign it.

After network applications have had their environment information or application parameters defined, these network applications can also have certain application parameters set on a per-user basis. This may be useful where, for example, one end-user wants a different color definition used when running the same application. The administrator can set these user-specific application parameters for each end-user.

User-specific application parameters are those application parameters that the administrator defined as user-specific when initially setting up application parameters. Only this subset of application parameters are allowed to be set as user-specific applications parameters. If a user-specific or override value is not specified, the generic application parameters are used when the user runs the application.

To set user-specific application parameters:

1. Open **LAN Services** from the desktop.
2. Open **LAN Server Administration**.
3. Open the appropriate domain object.
4. Open **User Accounts**.
5. Open the object for an existing user ID.
6. Select the **Applications** page.
7. Select the desired application from the public application list.
8. Select **User-Specific Application Parameters**. The User-Specific Application Parameters pop-up is displayed showing the parameter name, where appended, and the value.

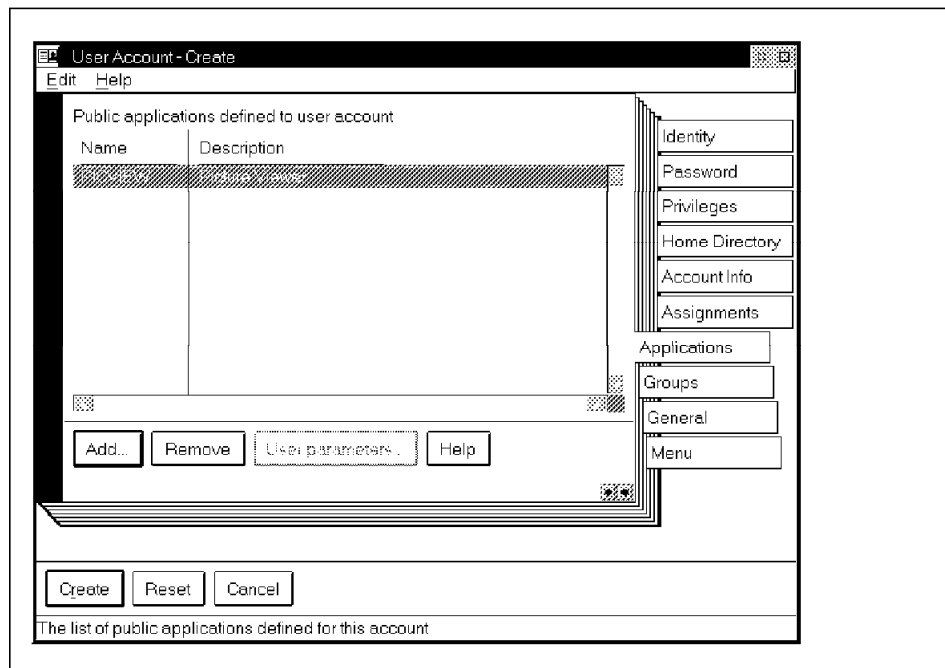


Figure 43. User Notebook -

To add user-specific application parameters:

- a. Select an application parameter from the **User default parameters** list.
- b. Select **Add**.

This adds the entry to the **User-specific parameters** list and displays the Set User-Specific Value pop-up.

- c. Enter the **value** for the application parameter.
- d. Select **OK**.



---

## Chapter 7. Advanced WorkSpace On-Demand Administration

This chapter covers the administration of the WorkSpace On-Demand server after the initial WorkSpace On-Demand client has been set up.

---

### 7.1 Logon Options

You can modify certain options that control the client logon process. These options are controlled in the CONFIG.SYS file with the RUNWORKPLACE=PMLOGON statement.

If you modify the CONFIG.SYS file in the Machine Class directory, all subsequent RIPL requesters (client machines) that are of that Machine Class type will have the change. This CONFIG.SYS file is located in the IBMLANRPLMACHINESBB10.USyyyyyyyy.mc directory on the server root drive, where yyyyyyyy.mc is the target machine class directory.

If you want the change to affect only one RIPL requester, modify the CONFIG.SYS in that machine's directory. This CONFIG.SYS file is located in the IBMLANRPLMACHINES<machine\_name> directory on the server root drive, where <machine\_name> is the target RIPL requester.

The PMLOGON parameters are optional. If PMLOGON is specified in the RUNWORKPLACE= statement without any parameters, all defaults are used. When the logon window is displayed, there is no user ID or password displayed, and the domain that is displayed is obtained from the client IBMLAN.INI file.

The following parameters can be specified on the PMLOGON command. They are displayed in the Logon window, and can be changed by the end-user:

- /U:<UserID> - User ID
- /P:<Password> - Password
- /D:<Domain> - Domain name

The following parameters can be specified on the PMLOGON command and are displayed in the Logon window, but cannot be changed by the end-user:

- /UF:<UserID> - Fixed user ID
- /PF:<Password> - Fixed password
- /DF:<Domain> - Fixed domain name

These parameters have the following prerequisites:

- /AUTO - Automatic logon, the Logon window is not displayed and password must be specified with the /PW switch.
- /BMP:<filespec> - This specifies a fully qualified path and file name for the bitmap displayed in the Logon window.

The following should be noted:

1. If multiple instances of the same parameter are specified (for example, /U:usera /U:userb), the last parameter is used, and all previous ones are ignored.
2. If multiple fixed and non-fixed instances of the same parameter type are used (for example, /U:usera /UF:userb /U:userc) , the last-fixed parameter is used (for example, /UF:userb), and all others are ignored.
3. The /AUTO parameter can be used with /U or /UF, with /P or /PF, and with /D or /DF. If multiple instances of the same parameter type are specified, the same rules apply as in the two previous notes.
4. If /AUTO is specified and there is insufficient information to attempt a logon (user ID is not specified), the Logon window is displayed and the /AUTO parameter is ignored.
5. If /AUTO and /PW are specified and there is insufficient information to attempt a log on (for example, either the user ID or password are not specified), the Logon window is displayed, and the /AUTO parameter is ignored. /PW is processed. You should be cautious about using the /AUTO parameter:
  - If /AUTO is specified and end-users log on successfully, when they log off from the desktop, they are automatically logged on again.
  - If /AUTO is specified and the end-user password has expired, the logon is no longer automatic. Any other end-user can change the password at this point by completing the Change Password window on the requester or client machine. This permits access by one end-user to another end-user account.
  - If /AUTO is specified and the end-user changes the password, the /AUTO parameter is not honored until the next time the requester or client machine is started. For example, if the end-user logs on and the password is expired, the end-user changes the password when prompted. When the end-user logs off (if /AUTO was specified), the logon fails because the password specified on the RUNWORKPLACE= statement and the password just changed do not agree. You must update the RUNWORKPLACE= statement to the new password.

### 7.1.1 Logon Bitmap

You can set the client Logon window to display any desired bitmap image. This allows you to customize the look of the Logon window for end-users. You can set all RIPL requesters (client machines) of the same Machine Class to use the same bitmap image or you can set any individual RIPL requester to display its own bitmap image.

The Logon window bitmap is controlled by the `RUNWORKPLACE=PMLOGON` statement in the client `CONFIG.SYS` file. If a bitmap is specified, that bitmap is scaled and displayed in the Logon window. If a bitmap is not specified, the `PMLOGON.BMP` file in the `\OS2` directory is used.

To change the client Logon bitmap:

1. Edit the desired Machine Class or RIPL requester `CONFIG.SYS` file using an ASCII editor. See *Modifying the Client Logon Options* for more information on the directory path of the `CONFIG.SYS` files.
2. Modify the `RUNWORKPLACE=Z:\OS2\PMLOGON.EXE` statement to include the `/BMP:filespec` parameter. The `filespec` parameter is the fully qualified path and file name of the bitmap to be displayed on the Logon window. For example:  
  
`RUNWORKPLACE=Z:OS2PMLOGON.EXE /BMP:Z:BITMAPSUSER1.BMP`
3. Save the `CONFIG.SYS` file.
4. Place the desired bitmap in the directory that is specified in the `RUNWORKPLACE=` statement.

#### Note

If `/BMP` is specified with a file name only and not a fully qualified path, the `DPATH` is used to locate the bitmap. If the bitmap file is not specified with the `/BMP` parameter, or if the specified bitmap file name is not valid or cannot be located, the `PMLOGON.BMP` file is used in the `\OS2` directory. If the `PMLOGON.BMP` bitmap file cannot be located, a default bitmap provided with the WorkSpace On-Demand product is displayed.

### 7.1.2 Changing the Background Bitmap

The background bitmap for the WorkSpace On-Demand client's desktop is specified in the file `OS2.INI` in the directory `\IBMLAN\RPLUSER\<clientname>\OS2`. When the client workstation is created, this file is taken unchanged from `\IBMLAN\RPL\BB10US\<machine_class>\OS2`. To change the background bitmap that the user sees after logging on, you will need an INI-file editor.

As always, when manipulating INI-files, it is best to make a safe copy of the file first. The name of the bitmap file is stored in the PM\_SystemBackground section; the key to edit is DefaultDesktopBackground.

You can change the background bitmap either for a single machine or for an entire class. If you modify the INI file in the \IBMLAN\RPL\BB10.US\<machine\_class>\OS2 directory, any clients you subsequently create will take the new INI file; if you modify the INI file in the \IBMLAN\RPLUSER\<client\_name>\OS2 directory of an existing client, only that machine will take the new file.

---

## 7.2 Printing

WorkSpace On-Demand provides for the use of local printers, networked printers, or both, from the same client workstation.

### 7.2.1 Network Printing

The first requirement for network printing with WorkSpace On-Demand is to ensure that the printer to be used is shared and that the user has at least Create access to it. You should also make sure that you have created a Printer object on the server. When these requirements are satisfied, you can complete the steps described in 6.3.4, “Printers” on page 103. This will create, if it does not already exist, a directory for the printer driver in the \OS2\DRIVERS subdirectory of the \IBMLAN\RPL\<operating\_system> subdirectory. For example, the driver for a printer that emulates an HP Laserjet will go in the subdirectory \IBMLAN\RPL\<operating\_system>\OS2\DRIVERS\LASERJET. You should bear this structure in mind when installing updated versions of printer drivers on the WorkSpace On-Demand server.

So that the user can use the network printer as soon as they log on, you will need to give the user a logon assignment for the printer share or alias.

The WorkSpace On-Demand shell allows the user to manipulate the printer object on the desktop by means of mouse button 2 and the context menu. For example, if they have several printer objects, the user can then choose which is to be the default. The user can also see details of, or cancel, their own print jobs.

### 7.2.2 Local Printing

Defining a local printer in the Client notebook causes a local printer icon to appear on the client’s desktop. The setup utility may prompt you to install

additional drivers for the printer, if it is of a type not already defined on that server.

---

## 7.3 File Index Tables

The File Index Table (FIT) provides a mechanism for a client workstation to redirect file I/O requests to the Workspace On-Demand server. The FIT is loaded into the client's memory, and whenever a file access is requested by the WorkSpace On-Demand client operating system or an application running on the client, the FIT is checked to resolve the request. A sample FIT entry is shown in Figure 44.

Z:NETSCAPENETSCAPE.INI MYSERVERNETSCAPENETSCAPE.INI
---

*Figure 44. Sample Fit Entry*

The example shown in Figure 44 redirects a file access request for the file NETSCAPE.INI, that the client system assumes is located in the NETSCAPE directory on its drive Z:, to the NETSCAPE alias on the server MYSERVER.

There are three basic types of File Index Table:

- Client machine FIT
- User FIT
- Default user FIT

The following sections explain the three types, the differences between them, and the way that they interact in the client workstation.

### 7.3.1 Client Machine FIT

The client machine FIT is loaded into the client workstation's memory when the WorkSpace On-Demand client operating system is loaded. By default, this FIT contains file redirection information for the client operating system itself, including device drivers and other machine-specific files.

Each client workstation that you define on your WorkSpace On-Demand server has its own unique machine FIT file. The client machine FIT file is created automatically when you define a client, and is assembled from two components:

- A WorkSpace On-Demand FIT file named DFBB10US.FIT, located in the IBMLANRPLFITS directory
- Hardware-specific redirection information, unique to the machine class for this client. These redirection commands are stored in a

machine-class FIT file, located in the  
IBMLANRPLMACHINESBB10.US<machine\_class> directory.

These two files are merged together when you define a client, to produce a client-specific machine FIT file. The client machine FIT file is stored in the IBMLANRPLFITS directory on the WorkSpace On-Demand server.

As an administrator, you may wish to add file redirection information for additional files or applications that will be accessed by all the client workstations on your server. You can do this by editing the WorkSpace On-Demand machine FIT file DFBB10US.FIT to add the records you require. These additional records will then be incorporated into all client-specific machine FIT files that you subsequently create using this WorkSpace On-Demand base machine FIT file.

**Note**

If you add your information to the DFBB10US.FIT after you have created a client machine you will need to delete and recreate the client machine to make these changes effective.

### 7.3.2 User FIT

The user FIT is loaded into the client workstation's memory when a user logs on to the client workstation using the standard simplified shell (PMLOGON.EXE). The user FIT contains file redirection information for the applications that are unique to that user.

This FIT is normally generated manually by a system administrator for each user when the user's user ID is created on the WorkSpace On-Demand server. The user FIT file for a particular user is located in the IBMLANDCDBUSERS<user\_id> directory, which is created automatically when you define the user ID on your WorkSpace On-Demand server.

### 7.3.3 The Default User FIT

If you do not create a user FIT file for a particular user, WorkSpace On-Demand will use a default user FIT for that user. This default user FIT is contained in the BB10USDU.FIT file, which resides in the IBMLANDCDB directory. The default user FIT supplied with WorkSpace On-Demand is shown in Figure 45 on page 129.

```

;Default user FIT
;   ?: is replaced with the client's boot drive
;   <DCSERVER> is replaced by the server name of the Domain Controller
;   <USER> is replaced by the user id

?:\NETSCAPE\MAIL          \\<DCSERVER\IBMLAN$\DCDB\USERS\<USER>\MAIL
?:\NETSCAPE\NEWS          \\<DCSERVER\IBMLAN$\DCDB\USERS\<USER>\NEWS
?:\NETSCAPE\SECURITY      \\<DCSERVER\IBMLAN$\DCDB\USERS\<USER>\SECURITY
?:\NETSCAPE\NETSCAPE.INI  \\<DCSERVER\IBMLAN$\DCDB\USERS\<USER>\NETSCAPE.INI
?:\NETSCAPE\NETSCAPE.!!!  \\<DCSERVER\IBMLAN$\DCDB\USERS\<USER>\NETSCAPE.!!!
?:\NETSCAPE\NETSCAPE.###  \\<DCSERVER\IBMLAN$\DCDB\USERS\<USER>\NETSCAPE.###
?:\NETSCAPE\COOKIES.TXT   \\<DCSERVER\IBMLAN$\DCDB\USERS\<USER>\COOKIES.TXT
?:\NETSCAPE\*.HTM        \\<DCSERVER\IBMLAN$\DCDB\USERS\<USER>
?:\NETSCAPE\*.DB         \\<DCSERVER\IBMLAN$\DCDB\USERS\<USER>
?:\NETSCAPE\*.HST        \\<DCSERVER\IBMLAN$\DCDB\USERS\<USER>
?:\NETSCAPE\*.TMP        \\<DCSERVER\IBMLAN$\DCDB\USERS\<USER>

```

Figure 45. Default User FIT File - BB10USDU.FIT

The default user FIT file contains entries for Netscape Navigator. However, you can add definitions for any other applications, just as you would to a user-specific FIT file.

There are three variables used in the default user FIT file. You can use these variables in a user-specific FIT file in place of a hard-coded boot drive, domain controller and user ID. This capability allows you to create “standard” user FIT files that can be used by groups of users who have common application requirements.

### 7.3.3.1 Combining Multiple User FITs

In order to reduce the amount of work involved in creating user FIT files, you can use a combination of user-specific and default user FITs. The user-specific FIT can contain entries unique to that user, and can then invoke the default user FIT. Figure 46 illustrates this technique.

```

;User-specific FIT entries

Z:\TOOLS\IMAGEDIT          \\SERVER01\TOOLS\IMAGEDIT

#include BB10USDU.FIT

```

Figure 46. Sample User FIT File Including the Default User FIT

The user FIT shown in Figure 46 contains a single redirection request for a user-specific application, then uses a #include statement to invoke the default user FIT. Note that, regardless of where the #include statement is

located in the user FIT file, the default user FIT entries are always loaded *after* the user-specific entries.

**Note**

If you do not place a `#include` statement in a user FIT file, the default user FIT file will *not* be read into memory, and *only* the user-specific entries in the user FIT file will be invoked.

You can use the `#include` statement to incorporate any FIT file, not just the default user FIT file. The FIT file to which the `#include` statement refers must reside in the IBMLANDCDB directory, and must conform to the standard 8.3 naming convention. This facility allows you to nest FIT files, and gives you the ability to define standard FIT files for commonly used applications, and incorporate them into a user's FIT using `#include` statements.

**Note**

There cannot be more than one optional `#include` statement in a single FIT file. This means that you must give careful consideration to how you structure your nested FIT files.

Given that there is a user-specific and a default user FIT file, an administrator has four options for implementing file remapping with FIT files at the user level.

1. Use only the default user FIT:

No action required; this is how the system functions by default.

2. Use a user-specific FIT file:

Create a user-specific FIT file, `<user_id>.FIT`, for the user ID in `IBMLANDCDBUSERS<user_id>`. The system now uses this FIT file for this user ID instead of the default user FIT file.

3. Use a user-specific FIT *and* the default user FIT:

As in number 2 above, but also use the `#include` statement to append the default user FIT redirection instructions.

4. No FIT mappings

Use number 2 above, but omit any redirection instructions and the `#include` statement.



#### Points to Consider

As supplied, the default user FIT file redirects Netscape user files to the user's own subdirectory in the Domain Controller Database. This arrangement insures that in a multi-server domain. As long as a user logs on to the same domain, his/her default user FIT will be available regardless of which server validates the logon and user-access permissions. However, there are at least two points to consider with this arrangement:

- By placing user data in the DCDB directories, you increase the amount of network traffic due to the DCDB replication process inherent in a multi-server environment. This additional network traffic could negatively impact performance and responsiveness, particularly on heavily loaded networks and/or servers.
- As defined in Figure 45 on page 129, the user will have write access to the directory that defines his/her user ID.

Alternatives include implementing user home directories, user-specific aliases or directing user files to a subdirectory in another tree that is *not* under the DCDB directories (for example, IBMLANRPLUSER) where they can be kept separate from essential system files. Information on implementing these techniques can be found in *LAN Server Network Administrator Reference Volume 3: Network Administrator Tasks*, S10H-9680.

### 7.3.4 FITs in Memory

As already mentioned, the machine FIT is loaded into the client workstation's memory when the WorkSpace On-Demand client operating system is loaded. The machine FIT is maintained in memory at all times until the client workstation is shut down or rebooted.

When a user logs on to the client workstation, the WorkSpace On-Demand server attempts to locate a user FIT for that user. If a user FIT exists, it is downloaded to the client workstation. If a user FIT does not exist, the WorkSpace On-Demand server downloads the default user FIT to the client.

When the client receives the user FIT, it merges its contents with those of the machine FIT already in memory. The user FIT entries are placed at the beginning of the in-memory FIT, followed by the machine FIT entries.

The in-memory FIT acts like a table. Entries are examined in sequential order until a match is found, at which point the access request is resolved

and the I/O is redirected accordingly. This means that if the user FIT and the machine FIT contain conflicting entries, the user FIT entry will take precedence over the machine FIT entry.

You can have the currently active FIT displayed to you at a WorkSpace On-Demand client workstation. See B.4.2, “Using RDRDEBUG to View the Client FIT” on page 260 for details on how this can be accomplished.

---

## 7.4 Setting Up Applications Under WorkSpace On-Demand

Section 6.4, “Adding Application Software” on page 108, explains the method for setting up a simple OS/2 application to run on a WorkSpace On-Demand client. The aim of this section is to guide you through enabling some more complex applications, and to show you what is involved in setting up any other applications you may want to run under WorkSpace On-Demand

### 7.4.1 OS/2 Applications

Several ingredients need to be in place to configure an OS/2 public application to work in WorkSpace On-Demand.

- The application code installed on the WorkSpace On-Demand server or in another server accessible to the user.
- An alias for the application code directory structure
- At least Read and Execute rights to that directory for the user, propagated to all the directory’s subdirectories
- A Public Application object for the program
- The appropriate environment variables to allow the client machine to reach the application’s data and code files (defined in the Parameters page of the Public Application notebook, as shown in Figure 41 on page 117).
- A user definition on the Applications page of the User notebook, as shown in Figure 43 on page 121.

The procedure for defining these is described in detail in Section 6.4, “Adding Application Software” on page 108.

As stated in Section 6.4, “Adding Application Software” on page 108, there is no universal method that will make any application work under WorkSpace On-Demand. A small investment of time in getting to know the structure of each application, where it stores its files and which are general and which are user-specific, will be well repaid in time saved in configuring the application for WorkSpace On-Demand.

In addition to the essentials listed above and in Section 6.4, “Adding Application Software” on page 108, there are several key points to consider when planning to use an application under WorkSpace On-Demand.

- Look at the Program object created by the application’s installation program. Note the invocation line, and any parameters it specifies. Does the object specify a working directory?
- What is the application’s directory structure? Does it have a single code directory, or separate \DLL and \HELP subdirectories? Does it install code files elsewhere, for example in \OS2\DLL? Each directory the application accesses will require a line in a FIT file to enable the application under WorkSpace On-Demand. If certain files in a directory are to be redirected to a different directory from the rest of the files in that directory, you will need an additional line in the FIT file to control this.
- Try installing the application on a stand-alone OS/2 workstation. Note the entries it makes in CONFIG.SYS. You may need to add these as parameters in the Application Definition notebook.
- Look for any files that are updated by the user after installation. Consider whether each WorkSpace On-Demand user will require his/her own copy of a file or whether a single, central copy will do. What access rights will users need to the files. Remember that this may not be the same as the access they want!

#### **7.4.2 Enabling the OS/2 Applications Supplied with WorkSpace On-Demand**

In this section, we discuss enabling the predefined applications, which are:

- Netscape Navigator Version 2.02
- Java Version 1.0x
- Personal Communications Lite for OS/2

#### **7.4.3 Netscape Navigator Version 2.02**

Netscape Navigator fits, in most respects, the pattern for an OS/2 application as described in 6.4, “Adding Application Software” on page 108. You will, of course, need to ensure that you have activated TCP/IP on the client machine for which you intend to define Navigator. The environment variable extensions to define on the Parameters page of the Application notebook are:

PATH=Z:\NETSCAPE	(append to END)
LIBPATH=Z:\NETSCAPE	(append to END)
HELP=Z:\NETSCAPE	(append to END)

*Figure 47. Parameters for Netscape Navigator*

However, Navigator users will, in most cases, need to write to a personal cache directory, bookmark file and NETSCAPE.INI file. A standard installation puts all these files in the same directory as Netscape's executable files, so to give the user Write access to the whole directory would be unwise. Furthermore, leaving the user files in the NETSCAPE directory would prevent each user from having his/her own copy of the files. In effect, there could be only a single set of bookmarks and cached pages. While there may be cases where this would be desirable, it would not be acceptable to the majority of users.

To get around this, we need to keep each user's copies of these files in a separate, user-specific directory to which that user has Write and Create access. Netscape Navigator is not designed specifically for a multi-user environment and so expects to find the user files in a standard path, determined at installation time. By modifying the user's FIT file, we can allow Navigator to use its standard paths, while WorkSpace On-Demand redirects the files to the user's specific location. The user FIT file extensions are shown in Figure 48 on page 135.

WorkSpace On-Demand looks for a user FIT file in the IBMLANDCDBUSERS<username> directory after the user logs on. If it finds one, it appends the contents to the client machine's FIT that it already holds in memory, with the result that the settings in the user FIT effectively override their equivalents in the client machine FIT.

In this example, the server name is BLUEJAY and the user FLYDBIRD. The administrator created the directory \BMLAN\RPLUSER\USERS\FLYDBIRD and shared it as FLYFILES. The user FLYDBIRD has read, write, create, delete and execute access to the subdirectory tree. Note that this method requires a different share or alias for each user; see 7.4.4, "Netscape Navigator - The Quick Way" on page 135, for a more general method.

```

; NETSCAPE support
Z:\NETSCAPE                BB10.US\NETSCAPE
Z:\NETSCAPE\BOOKMARK.*     \\BLUEJAY\FLYFILES\NETSCAPE
Z:\NETSCAPE\CACHE          \\BLUEJAY\FLYFILES\NETSCAPE\CACHE
Z:\NETSCAPE\COOKIES.*      \\BLUEJAY\FLYFILES\NETSCAPE
Z:\NETSCAPE\NETSCAPE.INI   \\BLUEJAY\FLYFILES\NETSCAPE\NETSCAPE.INI
Z:\NETSCAPE\NETSCAPE.!!!   \\BLUEJAY\FLYFILES\NETSCAPE\NETSCAPE.!!!
Z:\NETSCAPE\NETSCAPE.###   \\BLUEJAY\FLYFILES\NETSCAPE\NETSCAPE.###
Z:\NETSCAPE\SECURITY       \\BLUEJAY\FLYFILES\NETSCAPE\SECURITY
Z:\NETSCAPE\MAIL           \\BLUEJAY\FLYFILES\NETSCAPE\MAIL
Z:\NETSCAPE\NEWS           \\BLUEJAY\FLYFILES\NETSCAPE\NEWS
Z:\NETSCAPE\LICENSE        \\BLUEJAY\FLYFILES\NETSCAPE\LICENSE
Z:\NETSCAPE\PLUGINS        \\BLUEJAY\FLYFILES\NETSCAPE\PLUGINS

```

Figure 48. User FIT Extensions for Netscape Navigator

#### 7.4.4 Netscape Navigator - The Quick Way

The method described in 7.4.3, “Netscape Navigator Version 2.02” on page 133 gives a useful insight into how to configure a complex application requiring redirection of multiple user files and various combinations of access permissions. There is, however, a quicker way.

WorkSpace On-Demand supplies a default user FIT file, called BB10USDU.FIT. See 7.3.3, “The Default User FIT” on page 128, for a complete discussion of this feature. The default user FIT file contains three generalized names: ‘?’ for the client’s boot drive, <dcserver> for the server name and <user> for the user name. If you do not specify a customized user FIT file, WorkSpace On-Demand uses this file to redirect the Netscape user files to the appropriate directories.

We can make life even easier by not specifying an alias for the Netscape code directories. WorkSpace On-Demand installs Netscape in \IBMLAN\RPL\BB10.US\NETSCAPE, effectively a subdirectory of the drive known to the WorkSpace On-Demand client as Z:, the client’s boot drive. We can therefore consider the Netscape files to be located on the requester, even though they are physically situated on the server.

##### 7.4.4.1 A Word on NETSCAPE.INI

The first time Navigator is run, it will attempt to create a NETSCAPE.INI file in the user directory to which it is redirected. Before Navigator can be used to access any Internet or intranet sites, however, certain values, such as Proxy addresses or Socks servers, must be configured. You may choose to give your users the values they will need to do this for themselves, but it will probably save you some effort to do this for them. Since large groups of users will be using the same values, you can create an initial

NETSCAPE.INI in the \BMLAN\RPL\BB10.US\NETSCAPE directory, and then copy it to each user's user directory. Then, when the user starts Navigator for the first time, s/he will be able to go straight to the site s/he wants to look at, without having to configure the browser first.

#### **7.4.5 Java Runtime Version 1.0x**

The WorkSpace On-Demand installation utility installs Java support in the \BMLAN\RPL\BB10.US\JAVAOS2 directory, in the place where Netscape Navigator expects to find it. Since there are no user files to redirect, Java is effectively ready to run, provided you use Navigator for browser access.

The Java-specific environment variable CLASSPATH is already defined in the supplied CONFIG.SYS file for each Machine Class.

#### **7.4.6 Personal Communications Lite**

A simple Personal Communications Lite setup is quite easy to configure under WorkSpace On-Demand since there are no files to which the users require Write access. Although Personal Communications Lite provides a utility to allow the user to set up his/her own .WS configuration file, in practice most WorkSpace On-Demand administrators will prefer to define a central .WS file for their users, along with preset keyboard profiles and preset color definitions.

Again, Personal Communications Lite is installed in the \BMLAN\RPL\BB10.US\ directory tree, known to the WorkSpace On-Demand client as its boot drive, Z:, so you can define the application as being located on the requester.

#### **7.4.7 DOS and Windows Applications**

The same general principles apply to setting up a DOS or Windows application under WorkSpace On-Demand as to any other. There are a few special considerations, however.

There is no need to use OS/2 Warp Server's DOS and Windows Template to create the application. Although this will work, it is better to use the WorkSpace On-Demand template, which gives you the option to specify parameters for the application.

You will need to pay particular attention to the way your application handles the Windows INI files. In many cases, you will have to make a policy decision on whether to allow your users unrestricted access to their own copies of the INI files, by means of FIT file extensions, or whether to give them only Read access to a central set of INI files. Once again, there is no substitute for knowing your application!

#### 7.4.7.1 DOS and WinOS2 Settings

WorkSpace On-Demand supports the use of DOS and WinOS2 settings, although it has no equivalent to OS/2's Add Programs facility to define them automatically. You can define any setting as an Application-Specific entry on the Parameters page of the Application Definition notebook. The following example will set IDLE\_SECONDS to 60 and DOS\_BACKGROUND\_EXECUTION to OFF.

(Name)	(Value)
IDLE_SECONDS	60
DOS_BACKGROUND_EXECUTION	0

You can find a list of standard settings for many applications in the \IBMLAN\RPL\BB10.US\OS2\INSTALL\DATABASE.TXT file.

### 7.4.8 Special WorkSpace On-Demand Launch Parameters

WorkSpace On-Demand supports three special parameters that define the way an application is launched. They may be specified on the Parameters page of the Application Definition notebook, just like any other Application-Specific parameter.

#### 7.4.8.1 WSOD\_LAUNCH\_MINIMIZED

When this is specified with a value of 1, the application is launched minimized.

#### 7.4.8.2 WSOD\_LAUNCH\_NOCLOSE

When specified with a value of 1, this parameter causes a DOS or OS/2 VIO window not to close when the application terminates. This may prove useful in determining the cause of a problem with the application definition because you will be able to see any error messages the program generates. It is usually best removed once the program is working correctly.

#### 7.4.8.3 WSOD\_LAUNCH\_SESSION

This parameter gives the administrator full control over the type of session in which WorkSpace On-Demand will launch the application. Use this with care, because the application may not work at all if you choose the wrong type of session.

The values that WSOD\_LAUNCH\_SESSION can take, and the session types that it will launch, are as follows:

Value	Session type
1	Full-screen OS/2
2	Windowed OS/2
3	OS/2 PM
4	Full-screen DOS
7	Windowed DOS
10	WinOS2 Real Mode
12	WinOS2 Auto
15	WinOS2 Standard, seamless, separate VDM
16	WinOS2 Standard, seamless, common VDM
17	WinOS2 Enhanced, seamless, separate VDM
18	WinOS2 Enhanced, seamless, common VDM
19	WinOS2 Enhanced, full-screen
20	WinOS2 Standard, full-screen

---

## 7.5 Running Java Applications from the WorkSpace On-Demand Desktop

Java for OS/2 runs Java applications on the desktop in the applet viewer, APPLET.EXE. By making APPLET.EXE available as a public application, you can make Java applications available to WorkSpace On-Demand users.

You will need to configure a separate invocation of APPLET.EXE for each Java application you intend to make available this way, in each case passing the name of the HTML file defining the application as a parameter on the Invocation page of the Application notebook, as shown in Figure 49 on page 139.



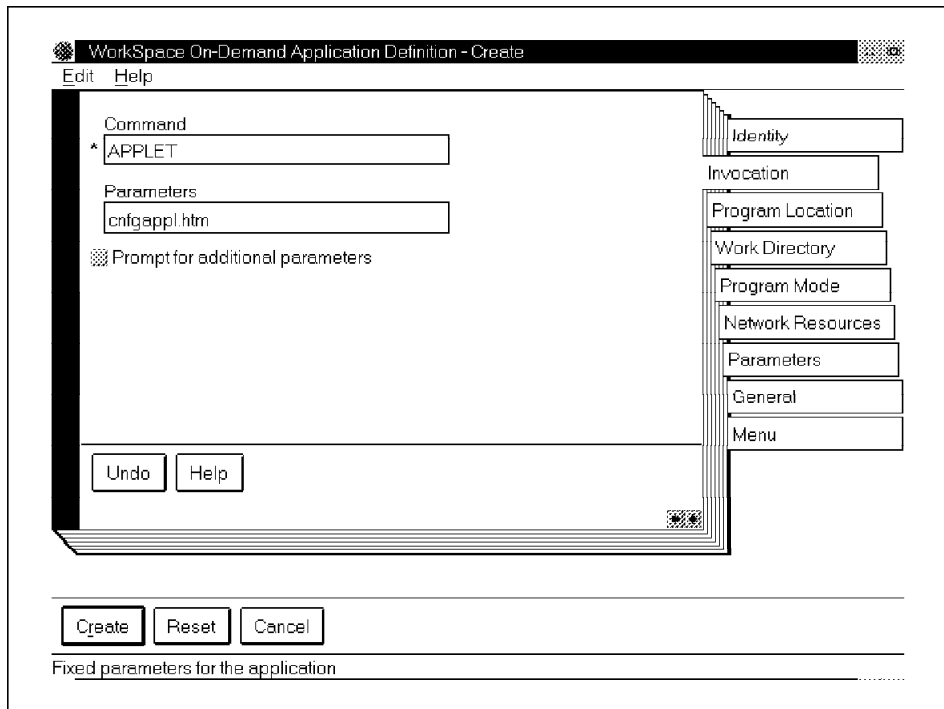


Figure 49. Invocation Page for the Java Configuration Applet

As in the case described in 7.4.6, “Personal Communications Lite” on page 136, you will probably not need to give your users Write access to any files to make Java for OS/2 run. Rather, by running the CNFGAPPL utility applet once on the server, you can create a single, central copy of the PROPERTY file, from which all the users can then read their Java configuration values.

## 7.6 Adding a Network Adapter

Although WorkSpace On-Demand supports a number of adapters, only a few of them are configured and enabled at the time the product is installed. In this section, we describe how to enable other adapters as well as how to add unsupported adapters to your list.

### 7.6.1 Enabling Supported Adapters

To enable server records that correspond to other LAN adapter types, the RPL.MAP file must be updated.

Besides the IBM Token-Ring Adapter, Remote IPL officially supports a number of additional NDIS LAN adapters. For more information about LAN

adapters, refer to <http://www.software.ibm.com/os/warp/pspinfo/nic.htm>. or to <http://www.software.ibm.com/os/warp/workspace/nics/index.htm>.

Although these adapters are supported, you must enable that support. WorkSpace On-Demand only delivers the OS/2 device drivers. You must manually copy the DOS driver .NIF file, .DOS file, and its appropriate .MSG file from the device driver diskette, option diskette, or diagnostics diskette. These diskettes are delivered with the LAN adapter. The appropriate files should be copied to the IBMLANDOSLANLSPDOS directory. Some adapters may not have .MSG files or .NIF files. The most critical file is the device driver itself, that is the .DOS file. The .MSG and .NIF files, if present, can have a different name than the device driver file.

The only DOS NDIS driver delivered with OS/2 Warp Server is the IBMTOK.DOS driver that is delivered with the LAN Support Program 1.38. As part of WorkSpace On-Demand, the following additional DOS NDIS drivers are provided:

<b>EL90X.DOS</b>	3Com Fast EtherLink/EtherLink XL
<b>ELNK16.DOS</b>	3COM EtherLink 16
<b>ELNK3.DOS</b>	3COM EtherLink III
<b>IBM16TR.DOS</b>	IBM Token-ring 16/4 II
<b>IBMEINDI.DOS</b>	IBM EtherJet ISA Ethernet Adapter
<b>IBMENI.DOS</b>	IBM LAN Adapter for Ethernet
<b>IBMFE.DOS</b>	IBM 100/10 PCI Ethernet Adapter
<b>IBMTRP.DOS</b>	IBM PCI Token-Ring Adapter
<b>SMARTND.DOS</b>	Madge Smart 16/4 Ringnodes
<b>SMC8000.DOS</b>	SMC Ethernet ISA Adapter
<b>SMC8100.DOS</b>	SMC TokenCard Elite

All other DOS NDIS drivers must be manually installed in the appropriate directories from the device driver diskette, option diskette, or diagnostics diskette.

For example, to enable support for the IBM LANStreamer Adapter, do the following:

1. In the \DOS subdirectory of the device driver diskette, find the files:

<b>IBMTTRDB.DOS</b>	DOS NDIS Device Driver
---------------------	------------------------

**IBMTRDB.NIF**      DOS NDIS Network Information File for IBM DOS Environments

**LT6.MSG**          DOS NDIS Message File

2. Copy all the above files to the Remote IPL subdirectory: IBMLANDOSLANLSPDOS.
3. In the \IBMLAN\RPL\RPL.MAP file, enable the appropriate server records by removing a semicolon in column one.

The entries for WorkSpace On-Demand with the LANStreamer adapter are:

```
YYYYYYYYYYY BB1USLS.CNF 3 10 N  WORKSPACE"ON-DEMAND"1.0"IBM"LANSTREAMER"MC"32"TOKEN"RING  "  ,,,  " R_BB10_US_OTKLS  "  "
```

Figure 50. Enabling LANStreamer MC32 Support in the RPL.MAP File

Because the IBM LANStreamer MC32 Adapter is supported for Remote IPL, all of the necessary configuration files to create clients for this adapter are present. However, you must enable the server records. You can determine the appropriate server records to enable by reading the descriptive comment field (field 7) in the server record.

4. Using the normal Remote IPL requester definition procedure, define the Remote IPL requesters using the newly enabled adapter types.

## 7.6.2 Enabling Unsupported Adapters

### Note

1. Enabling an unsupported adapter does not change the support statement for that particular adapter from IBM.
2. This procedure will enable an adapter for WorkSpace On-Demand only. For information on how to enable an adapter for DOS or OS/2, refer to the OS/2 Warp Server online documentation.

To enable an unsupported adapter, the following steps need to be completed:

- Locate the most current DOS and OS/2 device drivers and other files for the selected adapter. This can usually be obtained from the hardware vendor's Web site.
- Create a directory structure for the new drivers.
- Copy the device drivers and the related files into the RPL directory structure.

- Create a PROTOCOL.INI for the DOS environment.
- Create a CNF file for the new adapter.
- Modify the NDISDD.PRO file.
- Enable the associated server record in the RPL.MAP file.

In this section, we describe the procedure for adding a network adapter. This procedure that follows should work for most NDIS-compliant LAN adapters that support Remote IPL.

#### Note

At the time we did the research on adding the unsupported adapter, the Madge adapter was not yet included in the list of supported adapters in the WorkSpace On-Demand product. It has since been added to the list of supported adapters; so you will not need to perform these steps to utilize the Madge adapter in your environment. The steps that you need to follow to enable an adapter are as described in this section. And even though the Madge adapter is supported in WorkSpace On-Demand, we have left it in the redbook as an example of enabling an adapter.

As an example, we will add the Madge Smart 16/4 AT Plus Ringnode Adapter. For the remainder of this section, we will refer to this adapter as the the Madge adapter and the drive designation *D:* as the drive on the server where WorkSpace On-Demand is installed.

1. Copy the OS/2 driver and the NIF file to the D:\BMLANRPLBB10\USIBMCOMMACHS directory:

Drivers have already been copied into this directory during the installation of WorkSpace On-Demand. Check if the drivers are in this directory before starting the copy. For the Madge adapter, the following files were already present in the D:\BMLANRPLBB10\USIBMCOMMACHS directory:

<b>MDGND.OS2</b>	OS/2 NDIS Device Driver
<b>MDGND.NIF</b>	OS/2 NDIS Network Information File for IBM OS/2 Environments

Had the above files not been present, they would have to be copied.

### The .NIF File

The .NIF file contains important information that is used in the Override Network Adapter drop-down list that is part of the hardware page of the client definition notebook.

The example below is from the *MDGND.NIF* that shipped with the adapter.

```
[MDGND]
Type = NDIS_SINGL
Title = "Madge Fastmac Plus OS/2 NDIS MAC driver for
        Smart 16/4 Ringnodes"
...
```

The text in the title field is used to populate the **Override network adapter** drop-down list on the hardware page of the WorkSpace On-Demand client>Create settings notebook. You can modify the text included between the double quotation marks if you wish to provide a more meaningful description. However, the important point is that an adapter-specific .NIF file must exist in D:\BMLANRPLBB10\USIBMCOMMACS, and it must contain a title field. If the adapter you are configuring does not provide this information, you must provide it by creating your own .NIF file.

If a message file exists for the adapter, copy it into the D:\BMLANRPLBB10\USIBMCOM directory. For this particular Madge adapter, we could not locate a message file.

2. Create new subdirectories within the D:\BMLANRPLDOS directory. The name should be descriptive of the adapter being added. For the Madge adapter we created, the following directories:

D:\BMLANRPLDOSMADGE

D:\BMLANRPLDOSMADGEOS2

### Driver Directories

DOS drivers and their associated .NIF files are usually placed in the D:\BMLANDOSLANLSPDOS directory.

We've elected to place the DOS device driver into its own subdirectory so that the directories can be easily accessed and restored from a backup. This might simplify re-creating support for the adapter in the event that migrating to a newer version of WorkSpace On-Demand re-creates the original directory structure.

Your choice of directory location *must* be reflected in the .CNF file, which you will create and modify in a later step.

3. Copy the DOS driver and its associated .NIF into the directory you created. For the Madge adapter, we copied SMARTND.DOS to the D:\BMLANRPLDOSMADGE directory. If there are DOS message files, copy them into this directory as well.
4. Now you must create PROTOCOL.INI files to allow the adapter to load properly. Instead of creating them from scratch, we copy them from the IBM 16/4 Token Ring Network Adapter. The PROTOCOL.INI file we copied was located in the IBMLANRPLDOSTOKENRNG directory.

### Note

This is the PROTOCOL.INI file used to Remote IPL DOS on the WorkSpace On-Demand client workstations early in the remote boot process.

5. For our example, the entries are modified the PROTOCOL.INI file to include the correct driver and binding information as follows:
  - a. The bindings= entry in each section must be:  
`bindings=smartnd`  
  
This specifies that the physical DOS driver that loads is defined in the smartnd section of the protocol.ini file.
  - b. The final section must be:  
`[smartnd]`  
`drivername=smartnd$`  
  
This is the DOS driver section. It specifies that the DOS driver that protocols should bind with in memory is SMARTND.DOS. Comment all the IBM Token-Ring's parameters out. These are not valid for the Madge card. If you require any of these parameters, consult the online documentation.

Figure 51 on page 145 shows the final DOS PROTOCOL.INI file.

```
DOS PROTOCOL.INI used to boot OS/2 clients
This PROTOCOL.INI reduces the DMXJOMOD.SYS 'extra memory'
from 17 KB to 6 KB
This PROTOCOL.INI File is for the Madge adapter.
[PROTMAN_MOD]
    DriverName = PROTMAN$
[DXMJO_MOD]
    DriverName = NETBEUI$
    Bindings = smartnd
[DXMEO_MOD]
    DriverName = DXMEO$
    Bindings = smartnd
[smartnd]
    drivername = smartnd$
;    EARLYRELEASE
;    MAXTRANSMITS = 6
;    RECVBUFS = 2
;    RECVBUFSIZE = 256
;    XMITBUFS = 1
;    XMITBUFSIZE = 2040
```

Figure 51. Madge PROTOCOL.INI File for DOS

6. Copy this modified PROTOCOL.INI to: IBMLANRPLDOSMADGE
7. Now copy the OS/2 PROTOCOL.INI file from  
IBMLANRPLDOSTOKENRNGOS2 to  
IBMLANRPLDOSMADGEOS2.

**Note**

This is the PROTOCOL.INI file used during the first stage of the Remote IPL to load the WorkSpace On-Demand client operating systems on the workstations.

8. Modify the PROTOCOL.INI file to include the correct driver information as follows:
  - a. The bindings= entry in each section must be:  
bindings=smartnd  
  
This specifies that the physical OS/2 driver that loads is defined in the smartnd section of the protocol.ini file.
  - b. The final section must be:  
[smartnd]  
drivername=smartnd\$

This is the OS/2 driver section. It specifies that the DOS driver that protocols should bind with in memory is SMARTND.OS2. Comment all the IBM Token-Ring's parameters out. These are not valid for the Madge card. If you require any of these parameters, consult the online documentation.

Figure 52 shows the final OS/2 PROTOCOL.INI file.

```
; DOS PROTOCOL.INI used to boot OS/2 clients
; This PROTOCOL.INI reduces the DMXJOMOD.SYS 'extra memory' from 17k to 6k
; This PROTOCOL.INI File is for the Madge adapter.
[PROTMAN_MOD]
    DriverName = PROTMAN$
[DXMJO_MOD]
    DriverName = NETBEUI$
    Bindings = smartnd
    PiggyBackAcks = 0
    Sessions = 6
    NCBS = 6
    Names = 6
    Stacksize = 512
    Maxdatarcv = 512
[DXMEO_MOD]
    DriverName = DXMEO$
    Bindings = smartnd
[smartnd]
    drivername = smartnd$
;     EARLYRELEASE
;     MAXTRANSMITS = 6
;     RECVBUFS = 2
;     RECVBUFSIZE = 256
;     XMITBUFS = 1
```

Figure 52. Madge PROTOCOL.INI for OS/2

#### 9. Create a CNF file for the new Adapter

The CNF file defines how the machine boots. Once again, it is best to copy and modify an existing CNF file. For our example, we copied the IBM 16/4 Token-Ring CNF file, BB10USNTR.CNF, in the D:IBMLANRPL directory, to a file we called BB10USMDG.CNF also in the D:IBMLANRPL directory.

10. Modify the CNF file for the new adapter. Figure 53 on page 147 shows the modifications made to the CNF file to support the Madge adapter. The modified line is highlighted. The original line appears above the modified line and is commented out.



```

; OS/2 Boot Block Configuration
; (IBM Token-ring Compatible Adapter-Modified for the Madge adapter)
RPL DOS\RPLBOOT.SYS
DAT DOS\MFSD20.SYS
ORG 1000H
LDR BB10.US\OS2LDR ~ OS2LDR UFSD.SYS MFSD20.SYS
DAT DOS\UFSD.SYS
; org line : DAT DOS\TOKENRNG\OS2\PROTOCOL.INI
DAT DOS\MADGE\OS2\PROTOCOL.INI
DAT C:\IBMLAN\DOSLAN\LSP\DXM.MSG
DAT C:\IBMLAN\DOSLAN\LSP\DOS\LT2.MSG
EXE C:\IBMLAN\DOSLAN\LSP\NETBIND.COM ~ ~ ~
; **NETBIOS and IEEE 802.2*****
; DRV C:\IBMLAN\DOSLAN\LSP\DXMTOMOD.SYS PBA=0~S=12~ST=12~C=14~O=N ~ ~
; DRV C:\IBMLAN\DOSLAN\LSP\DXMEOMOD.SYS ~ 10 ~
; **NETBIOS and IEEE 802.2*****
;
; **NETBIOS*****
DRV C:\IBMLAN\DOSLAN\LSP\DXMJOMOD.SYS ~ 6 ~
; **NETBIOS*****
DRV C:\IBMLAN\DOSLAN\LSP\DXMAOMOD.SYS 001 ~ ~
; org line : DRV C:\IBMLAN\DOSLAN\LSP\DOS\IBMTOK.DOS ~ ~ ~
DRV C:\IBMLAN\RPL\DOS\MADGE\SMARTND.DOS ~ ~ ~
DRV C:\IBMLAN\DOSLAN\LSP\PROTMAN.DOS /I: ~ ~

```

Figure 53. WorkSpace On-Demand Boot Block Definition File BB1USMDG.CNF

Here we load everything exactly as the IBM Token-Ring adapter does. The change is to load the correct Card Driver (SMARTND.DOS) and PROTOCOL.INI. This is the same PROTOCOL.INI file we modified earlier to support the Madge adapter.

#### 11. Modify the NDISDD.PRO file

The list displayed in the GUI is dependent on entries in the NDISDD.PRO file as well as on entries in the RPL.MAP file. The server uses the NIF file specified in the NDISDD.PRO to build the CONFIG.SYS and the PROTOCOL.INI for machines that are defined to use this adapter.

```

;NDIS device driver profile records are composed of 4 fields.
;They are described below.
;Field 1 This field specifies the DOS NDIS device driver name.
;Field 2 This field specifies the subdirectory name that
;         contains the configuration file(s) needed to remote
;         boot the RIPL client.
;         d:\IBMLAN\RPL\IBMCOM\xxx\ contains
;         the configuration files needed to RIPL OS/2 clients.
;         d:\IBMLAN\RPL\DOS\xxx\ contains the
;         configuration file(s) needed to RIPL both DOS and OS/2 clients.
;         d - the drive on which the RIPL component has been installed
;         xxx - the text located in the second field of a given line
;Field 3   NIF file name for OS/2 device driver
;Field 4   CNF file name ID reference for server record ID
IBMTOK.DOS  TOKENRNG  IBMTOK.NIF  OTKNTR
IBMTOK.DOS  TOKENRNG  IBMTURBO.NIF  OTKTTR
SMARTND.DOS  MADGE  MDGND.NIF  OTKMTR
MACETH.DOS  ETHERNET  MACETH.NIF  OET
ELNKM.C.DOS  ELNKM  ELNKM.NIF  OET3EM
ELNKII.DOS  ELNKII  ELNKII.NIF  OET3EI
IBMENI.DOS  ETHILAEI  IBMENI.NIF  OETLAE
F1MAC.DOS  ETHIPS2I
ELNK16.DOS  ELNK16
ELNK3.DOS  ELNK3  EL3IBM02.NIF  OET3E3
SMC8000.DOS  WDPLUS  ETHOS2AT.NIF  OETWDP
IBM16TR.DOS  TOKN164I  IBM160S2.NIF  OTK164
IBMENII.DOS  ETHILAE  IBMENII.NIF  OETAEE
IBMTTRDB.DOS  TOKNLS32  IBMTTRDB.NIF  OTKLS
IBMEINDI.DOS  ETHERJET  IBMEINDI.NIF  OETEJ
E100B.DOS  IBM_E100  E100BE02.NIF  OETE100
ENDS2ISA.DOS  IBM_CLA  ENDS2ISA.NIF  OETCLA

```

Figure 54. WorkSpace On-Demand NDIS Driver Profile File, NDISDD.PRO

In Figure 54, the highlighted line is the one that was added for the Madge adapter. The DOS NDIS driver is listed in the first field, SMARTND.DOS. The directory containing the DOS drivers for this particular adapter is d:\IBMLANRPLDOSMADGE. The name of the OS/2 NIF file is MDGND.NIF, and the CNF file name reference for the server record ID is R\_BB10\_US\_OTKMTR.

12. Modify the RPL.MAP file. The following line was added for the Madge adapter:

```

; server record fields:
YYYYYYYYYYYY BB10MDG.CNF 3 10 N " WORKSPACE"ON-DEMAND"1.0" MADGE"16/4"AT_PLUS"RINGNODE"NON-NDIS " " ,,, " R_BB10_US_OTKMTR " "

```

Figure 55. WorkSpace On-Demand RPL.MAP File, Server Record Entry

13. The new adapter should now be selectable from the drop-down list in the Administration GUI when defining a new WorkSpace On-Demand client.

---

## Chapter 8. WorkSpace On-Demand Machine Classes

This chapter discusses a fundamental concept of WorkSpace On-Demand, the concept of Machine Classes. Because they provide the method of supporting WorkSpace On-Demand clients on disparate hardware platforms, understanding Machine Classes is crucial when implementing WorkSpace On-Demand in your environment .

This chapter examines Machine Classes from three perspectives:

- Overview  
What is a Machine Class and how it is used?
- Implementing Machine Classes: Getting Started  
What steps and procedures are required for *all* types of Machine Classes?
- Creating a Machine Class: Using Examples  
Example of how to create a Machine Class with a methodology for implementing hardware-specific Machine Classes:
  - Video Subsystem
  - SCSI Support
  - CD-ROM
  - Keyboard
  - Serial Mouse

---

### 8.1 Machine Classes: An Overview

Simply stated, a Machine Class is the *structured* collection of device drivers and other support files that defines a WorkSpace On-Demand client's hardware environment.

In this definition, we use the word *structured* because the location of these files on the WorkSpace On-Demand server is important to successfully implementing Machine Classes.

#### 8.1.1 What is a Machine Class?

A change in one component of a workstation configuration will require a different Machine Class in your WorkSpace On-Demand environment. Some of the components which, when changed, can result in a new Machine Class are:

- Video adapters (most important)
- Network adapters
- Monitors (try to use a class of monitors that fit within a single desired resolution/frequency range)
- Bus types (ISA, MCA, PCI, EISA)
- Hard drive connection (SCSI, IDE)
- Keyboards
- Mouse types

If you try to install and support a wide range of different components with all their possible combinations, you will end up with a very large number of Machine Classes to test and administer. For example, let's assume that you plan to implement WorkSpace On-Demand on a variety of PC hardware including:

- Five different graphics adapters
- Five different network adapters
- Two different hard drive types (SCSI and IDE)

Simply taking into account the graphics adapters and network adapters, this results in a large set of Machine Classes, as shown in Table 10.

*Table 10. Matrix of Machine Classes for a Given Hardware Combination. This table shows the resulting combination of Machine Classes if you plan to support five different graphics adapters and five different network adapters.*

	<b>graphics adapter A</b>	<b>graphics adapter B</b>	<b>graphics adapter C</b>	<b>graphics adapter D</b>	<b>graphics adapter E</b>
Network adapter 1	MC1A	MC1B	MC1C	MC1D	MC1E
Network adapter 2	MC2A	MC2B	MC2C	MC2D	MC2E
Network adapter 3	MC3A	MC3B	MC3C	MC3D	MC3E
Network adapter 4	MC4A	MC4B	MC4C	MC4D	MC4E
Network adapter 5	MC5A	MC5B	MC5C	MC5D	MC5E

Note that the table above does not take into account the differences between SCSI and IDE drives. To support both SCSI and IDE hard drives

along with all of the possible combinations of network and graphics adapters, you must double the number of classes shown in the table, resulting in a total of 50 Machine Classes.

As you can see from the example shown in Table 10 on page 150, it is important to minimize the number of different hardware combinations on your client workstations. This will significantly reduce the effort and cost associated with creating, testing and administering Machine Classes in your WorkSpace On-Demand environment.

### **8.1.2 Machine Class File Structure**

We have alluded to the fact that the structure, or location, of Machine Class files is of critical importance when implementing Machine Classes. This subject will be covered in more detail in the getting started and example sections; see 8.2, “Implementing Machine Classes: Getting Started” on page 154 and 8.3, “Creating Machine Classes: An Example” on page 157.

### **8.1.3 Components of a WorkSpace On-Demand Machine Class**

The structured collection of device drivers and other support files that comprise a Machine Class include:

- CONFIG.SYS
- AUTOEXEC.BAT
- WIN.INI
- SYSTEM.INI
- OS2.INI
- OS2SYS.INI
- XXXX.FIT
- Device-specific directories that include the device drivers control and data files needed to support:
  - Video Adapter
  - Sound Card
  - DASD Type
  - CD-ROM
  - Keyboard
  - Mouse
  - Other devices, for instance scanner, badge reader, and so on.

#### Note

In our discussion of Machine Classes, we have not included the Network Adapter nor any of the device, protocol or MAC drivers used by a network adapter. Although the network adapter is a major component of a WorkSpace On-Demand client's hardware environment, configuring support for network adapters, including the creation of .CNF files, is a part of configuring RPL support. Therefore, we address the issue in 7.6, "Adding a Network Adapter" on page 139.

Depending on the hardware supported, the content— or even the existence— of these files will differ among various types of WorkSpace On-Demand client machines. It is the differences in these files that must be *documented* and then incorporated into a new Machine Class in order to enable support for a new WorkSpace On-Demand client hardware platform.

In order to more easily manage and maintain your WorkSpace On-Demand Manager system, we recommend that you not deploy these new files by copying them into the WorkSpace On-Demand RIPL image. As you will see in the examples that follow, we believe that a better strategy is to create new, or private, subdirectories and FIT-file entries that support these new hardware devices. In this way, the Machine Classes that you've created can be easily backed up and restored whenever a new release of WorkSpace On-Demand installed on the WorkSpace On-Demand server.

Unlike conventional RIPL servers, which provide support for a single, generic machine type, a WorkSpace On-Demand server is capable of concurrently supporting multiple Machine Classes.

### 8.1.4 Machine Classes Shipped with WorkSpace On-Demand

WorkSpace On-Demand provides several different Machine Classes with the product. These Machine Classes include:

<b>IBM300GL</b>	IBM Personal Computer 300GL model 6282-38U
<b>IBM300G2</b>	IBM Personal Computer 300GL model 6272-880
<b>IBM350</b>	IBM Personal Computer 350 model 6586-57H
<b>IBM730</b>	IBM Personal Computer 730 model 6877-KAZ
<b>IBM750</b>	IBM Personal Computer 750 model 6887-86B
<b>ISAVGA</b>	Any PC with an ISA Bus and VGA Display Adapter that is supported by OS/2 Warp Version 4

These Machine Classes can be modified, through a GUI, to accommodate variations in the type of network adapter, monitor and video resolution needed by a given Machine Class. However, this flexibility may not be sufficient to adequately define the machines you wish to use as WorkSpace On-Demand clients.

Consider, for example, that each of two IBM PC 730 machines, identical in every aspect *EXCEPT* that each machine has a different video adapter chipset, requires its own separate Machine Class, and you can see how important Machine Classes are to successfully implementing WorkSpace On-Demand. It should be apparent that the Machine Classes shipped with WorkSpace On-Demand may not address all of the WorkSpace On-Demand client hardware configurations needed to support WorkSpace On-Demand in a particular user's environment.

There are ways to expand the types of client machines supported for use as WorkSpace On-Demand clients. One way is to engage an IBM Services Provider or IBM Business Partner to develop and test Machine Classes to meet your specific needs. The other way, which we will explore in the rest of this chapter, is to develop your own WorkSpace On-Demand Machine Classes.

### 8.1.5 Modifying Machine Classes

By using one of the Machine Classes supplied with WorkSpace On-Demand as a prototype, you can create your own Machine Class to accommodate the specific hardware requirements of your WorkSpace On-Demand clients. This subject is covered in more detail in 8.2, "Implementing Machine Classes: Getting Started" on page 154 and in 8.3, "Creating Machine Classes: An Example" on page 157.

### 8.1.6 Machine Class Feasibility Tests

Use the following two tests to confirm if you can *easily* create a Machine Class to support the type of machine you wish to use for a WorkSpace On-Demand client.

#### 1. The Initial Test:

Can you RIPL the target client using the generic ISAVGA.MC Machine Class? See Section 8.1.4, "Machine Classes Shipped with WorkSpace On-Demand" on page 152.

This is a *low-level* Machine Class providing the most basic machine functions.

---

**Note**

When working with the Generic ISAVGA.MC Machine Class, depending on whether or not your target machine has a supported Network Interface Adapter, you may have to configure and enable a working .CNF file with appropriate entries in the RPL.MAP file before proceeding with this test. See Section 7.6, "Adding a Network Adapter" on page 139, for a complete discussion of configuring Network Interface Adapters.

## 2. The In-Depth Test:

Can you successfully install and configure OS/2 Warp Version 4, including the *adapters-of-choice*, on the target client?

The adapters-of-choice refer only to device drivers obtained from the OS/2 Warp Version 4 CD-ROM, the OS/2 Device Drivers CD-ROM or from the OS/2 Web site or Bulletin Board systems.

Successfully executing this test verifies that you can isolate and extract the required device driver, INI and other support files from a working system that are necessary to create a viable Machine Class, one that supports the video adapter, sound card, mouse and other devices that you wish to use.

Successfully executing this test also insures that video adapter and other components chosen for your WorkSpace On-Demand clients are supported by IBM.

---

**Note**

When testing, it may be possible to develop a Machine Class for hardware that fails either of these tests. However, the effort required will be much greater than for those machines which pass these tests. There is also the issue that machines with devices and drivers that are not supported by OS/2 Warp Version 4 (See test #2) may not be supported.

---

## 8.2 Implementing Machine Classes: Getting Started

As mentioned in 8.1, "Machine Classes: An Overview" on page 149, Machine Classes are used to define the hardware environment of the WorkSpace On-Demand clients. The steps necessary to configure hardware support will differ depending on the type of hardware that is to be supported. For example, configuring support for a specific video subsystem



requires the most extensive modifications to device driver directories and configuration files like OS2.INI.

Other supported hardware, such as a SCSI controller, may only require changes to the WorkSpace On-Demand client's CONFIG.SYS file. Prior to executing the steps necessary to configure support for specific devices like these, there are several initial steps common to all types of Machine Classes that must be performed first.

### 8.2.1 Utilities and Tools Needed to Create a Machine Class

Before beginning to create a Machine Class, you should obtain several software tools that you will need. These tools include:

- OS/2 INI editor/viewer
- File Comparison Program
- Disk Comparison Program (optional)
- INI Application Copy Tools (REXX source provided)

See Appendix F, "Tools for Modifying and Managing CONFIG.SYS and INI Files in Machine Classes" on page 281, for the source-code for these tools. These tools copy an application entry, including its keywords and data values, from one \*.INI file to another.

- Network Sniffer Tool

A sniffer tool such as IBM DatagLANce lets you keep track of the exact network traffic between the server and your RIPL client, which will help you discover various problems.

You should also have a good understanding of the OS/2 device driver model. You can find very useful information in the ITSO redbook titled *The Guide to OS/2 Warp Device Drivers*, SG24-4627.

### 8.2.2 Choosing a Prototype Machine Class

The first step in creating a Machine Class is to determine which of the existing Machine Classes included with WorkSpace On-Demand is the appropriate choice for your Machine Class *prototype*. The use of a prototype greatly simplifies the process of creating a Machine Class. There are two rules for selecting the appropriate prototype.

1. If an existing Machine Class exactly matches the video subsystem in the target Machine Class, use it as the prototype Machine Class.

The first rule concerns the type of video subsystem support that is required. Configuring a Machine Class for video support is the most complex and tedious of all the Machine Class possibilities. For this

reason, if you find that your target system's video subsystem exactly matches one of the supplied Machine Classes, you should use it as the prototype and make modifications to the other hardware components as needed.

2. In all other cases, use the *generic* ISA/VGA Machine Class as the prototype.

The generic, or ISA/VGA Machine Class provides support for the most basic hardware capabilities. Starting with these most basic capabilities, a Machine Class can be created to address the hardware requirements of your target WorkSpace On-Demand client machines.

### 8.2.3 Creating a New Machine Class's Directory Structure

#### Attention!

It is **imperative** that you be logged on to your OS/2 Warp Server as an **administrator** before attempting to complete any of the following steps. This is because administrator authority is required to propagate the ACLs needed to make files and directories accessible to RPLUSER. Without them, a Machine Class will fail. If you use an Entry version of OS/2 Warp Server, you have to apply all access profile information to your changes manually.

The first step is to create the directory structure and files needed to support a new Machine Class. In the following example, we use the generic Machine Class, ISAVGA.MC, to create a new Machine Class called CPQ5120E.MC.

Create a new subdirectory for the CPQ5120E.MC in the *existing* Machine Class directory. For example:

```
CD D:\IBMLAN\RPL\MACHINES\BB10.US
MD CPQ5120E.MC
```

Once the prototype Machine Class is determined, copy its files, including the existing directory structure, into a new subdirectory for the new Machine Class. For example:

```
XCOPY D:\IBMLAN\RPL\MACHINES\BB10.US\ISAVGA.MC\*.*
D:\IBMLAN\RPL\MACHINES\BB10.US\CPQ5120E.MC /E /S /V
```

Next, you should determine if the new Machine Class requires a File Index Table (FIT). Remember that a FIT file is used to redirect file I/O by mapping WorkSpace On-Demand client file names to WorkSpace On-Demand server file names. As a general rule-of-thumb, hardware that requires access to

special files in dedicated directories will need a FIT file to map an access path to them. Some examples include:

- Video Drivers
- Sound Card Support
- Other, Specialized Devices

If you do need to provide a FIT file, you should copy it from the prototype Machine Class into the CPQ5120E.MC. For example:

```
COPY D:\IBMLAN\RPL\MACHINES\BB10.US\IBM350.MC\IBM350.FIT
D:\IBMLAN\RPL\MACHINES\BB10.US\CPQ5120E.MC\CPQ5120E.FIT
```

**Note**

On ISAVGA.MC and FIT, the generic Machine Class, ISAVGA.MC, does not provide a FIT file. If you are using ISAVGA.MC as your prototype Machine Class *and* you also need to provide a FIT table, you should examine the FIT tables in the other Machine Classes to determine which is the closest to meeting your needs. In this example, we used the FIT from IBM350.MC.

Now that you have a Machine Class's directory structure in place, the next step is to customize it to support the hardware that you require.

---

## 8.3 Creating Machine Classes: An Example

The following sections are examples of how to create Machine Classes to support specific hardware requirements for WorkSpace On-Demand clients. Once you've worked with these specific examples, you should have an understanding of the methodology used to create Machine Classes for the specific hardware environments that you need to support.

### 8.3.1 Creating a Machine Class for a Specific Video Adapter

This subsection describes how to create a Machine Class for a Cirrus Logic 5434 video adapter.

#### 8.3.1.1 Introductory Remarks - (Methodology)

For this exercise, we used a Compaq Prolinea 5120e as our target machine. Because this particular machine is shipped with a Cirrus Logic 5434 video adapter, we'll need to create a new Machine Class to support this machine as a WorkSpace On-Demand client. In order to simplify this exercise, we installed a network adapter that is supported by WorkSpace On-Demand, the IBM 16/4 Auto (ISA bus).

Next, we confirmed that this machine is a candidate for use as a WorkSpace On-Demand client by first verifying that it could be RIPL'ed from the WorkSpace On-Demand server by using the generic ISAVGA.MC Machine Class and by installing OS/2 Warp Version 4 (in our test, we installed from an OS/2 Warp Version 4 CD-ROM).

### 8.3.1.2 Creating an INI File for the New Machine Class

The first phase is to enable the new Machine Class to appear in the WorkSpace On-Demand Manager administration GUI as a new, selectable entry. The GUI information is provided by the CPQ5120E.INI file in the D:\IBMLAN\RPL\MACHINES\BB10.US\CPQ5120E.MC subdirectory. Before working with the file, we have to rename the copied ISAVGA.INI to CPQ5120E.INI.

Using an INI-file editor, you must make sure that all data entries for the *machine\_class* application are correct. You will find the following entries in the CPQ5120E.INI:

<b>BUS_TYPE</b>	Value is ISA, VLB, MCA or PCI
<b>DISKETTE_SUPPORT</b>	Value is YES or NO
<b>HARDFILE_TYPE</b>	Value is IDE or SCSI
<b>KEYBOARD_MODE</b>	Value is " "
<b>KEYBOARD_TYPE</b>	Value is 101 or appropriate
<b>MOUSE_TYPE</b>	Value is PS/2 MOUSE or SERIAL MOUSE
<b>NAME</b>	Value is CPQ5120E which will appear in the OS/2 Warp Server Administration GUI when you create a new WorkSpace On-Demand client
<b>PARALLEL_SUPPORT</b>	Value is YES or NO
<b>REMARK</b>	Value is "CPQ5120E, ISA, Cirrus Logic 5434" which is just a description field
<b>SERIAL_SUPPORT</b>	Value is YES or NO
<b>VIDEO_MONITOR</b>	Value is "PARAMETER_NOT_SUPPORTED"
<b>VIDEO_RESOLUTION</b>	Value is "PARAMETER_NOT_SUPPORTED"
<b>VIDEO_TYPE</b>	Value is "Cirrus Logic 5434"

### Selectable Monitor and Video Resolution

If you want to enable the OS/2 Warp Server Administration GUI to give you the option to change the Monitor and Video Resolution information, you must delete the two keywords, VIDEO\_MONITOR and VIDEO\_RESOLUTION. A correct implementation of this function requires a *VIDEO.CFG* and an *SVGADATA.PMI* file from your new machines' driver installation. This topic is covered later in this discussion.

With this data in place, your new Machine Class will appear in the OS/2 Warp Server Administration GUI. The next steps will define which files are necessary to support your CPQ5120E class.

#### 8.3.1.3 Determining the Differences

The next phase of this process is establishing a baseline for the configuration files. This is done in order to track which files are modified or added to the target machine's configuration when configuring support for the Cirrus Logic 5434 video adapter. Beginning with a machine that has OS/2 Warp Version 4 installed with VGA support, there are several steps to be performed in this phase:

1. Boot Drive Snapshot (Preinstallation)

After installing OS/2 Warp Version 4 with VGA support and verifying that the Compaq Presario 5120e will boot, a *snapshot* file is created that captures the name, size, and date of every file on the boot drive. For example:

```
ATTRIB -r -h * /S
DIR C:\*.* /S /O:GN > PRESVGA.DAT
```

The name, date and size of these captured files provides a basis for determining which files are copied and renamed as a consequence of installing the Cirrus 5434 SVGA support.

2. Stripping Archive Bits

The next step is to remove the archive bit from all files on the boot drive prior to installing the Cirrus 5434 SVGA support. For example,

```
ATTRIB -A C:\*.* /S
```

This step provides a method to verify which files are *touched* during the installation of Cirrus 5434 SVGA support.

3. Saving OS/2 System Files

In order to re-create the changes to the OS/2 system files, you need to create backup copies of the following files:

CONFIG.SYS  
AUTOEXEC.BAT  
OS2.INI  
OS2SYS.INI  
WIN.INI  
SYSTEM.INI

#### 4. Install Cirrus 5434 support

Using the Selective Install provided by OS/2 Warp Version 4's installation facility, install support for the Cirrus Logic 5434 video adapter.

##### Note

For the video resolution, if you want to provide the system administrator with maximum flexibility when configuring a WorkSpace On-Demand client, **do not change the video resolution after installing SVGA support**. Changing the video resolution creates an entry in the WorkSpace On-Demand client's OS2.INI file that fixes it to whatever choice is made after installation. By not changing the video resolution at this point, the system administrator keeps the ability to select the desired resolution from the drop-down list in the Create Client panel of the WorkSpace On-Demand Manager.

#### 5. Boot Drive Snapshot (Postinstallation)

Create a second *snapshot* file that captures the name, size and date of every file on the boot drive after installing SVGA support. For example:

```
ATTRIB -r -h * /S  
DIR C:\*.* /S /O:GN > POSTSVGA.DAT
```

#### 6. Capture All Changed OS/2 System Files

In order to re-create the changes to the OS/2 system files, you need to recapture the following changed files:

CONFIG.SYS  
AUTOEXEC.BAT  
OS2.INI  
OS2SYS.INI  
WIN.INI  
SYSTEM.INI

With these files in place, you can determine the changes made during the installation of SVGA support. When saving these files into the same subdirectory as the ones prior to the SVGA installation, make sure you give them unique names to easily differentiate them from the original files.

#### 7. Capture All Other Changed Files

The last step in this phase is to capture the names of all the files that are touched by the Cirrus Logic 5434 SVGA installation. This can be done by checking for those files whose archive bit has been changed (remember that we stripped all archive bits in a previous step). For example:

```
DIR C:\*.* /S /AA > ATTRIB.DAT
```

Now that we have a method of identifying which files are affected by installing the Cirrus Logic 5434 video adapter, we can proceed with creating a Machine Class to support this configuration on our target machine.

#### 8.3.1.4 Determining What to Change

The best way to judge what files need to be changed is to take a close look at the ATTRIB.DAT file, which provides a list of all the files touched by the installation process. We provide a sample list and our comments as an example so that you can determine which files apply to your specific circumstances.

The volume label in drive C is OS2.  
The Volume Serial Number is 6725:6415.

Directory of C:\

9-05-97	1:24p	3541	0	CONFIG.SYS
9-05-97	1:31p	312	0	WP ROOT. SF
4 file(s)		5992 bytes used		

**Remark:** While we don't care about WP ROOT. SF., We will examine CONFIG.SYS in detail and document the changes that are made to this file.

Figure 56 (Part 1 of 7). Files Changed by the Cirrus Logic Install Process

```

Directory of C:\DMISL\BIN

9-05-97  1:29p      74752          0  sldb.dmi
          1 file(s)      74752 bytes used

Remark: This is Systems Management Information that is
not needed to build a Machine Class.
Directory of C:\IBMCOM

9-05-97  1:30p      699          0  LANTRAN.LOG
          1 file(s)      699 bytes used

Remark: The LANTRAN.LOG file gets recreated every time LAN
services is started. Therefore the archive bit is set. This file
is not needed to create a Machine Class.
Directory of C:\IBMLAN\LOGS

9-05-97  1:31p      48          0  NET.AUD
          1 file(s)      48 bytes used

Remark: The NET.AUD file gets recreated every time LAN
services is started. Therefore the archive bit is set. This file
is not needed to create a Machine Class.

```

*Figure 56 (Part 2 of 7). Files Changed by the Cirrus Logic Install Process*



Directory of C:\OS2

9-05-97	1:24p	60481	0	INI.RC
9-05-97	2:00p	201884	0	OS2.INI
9-05-97	1:28p	69525	0	OS2SYS.INI
9-05-97	1:24p	22778	0	PMCONTRL.INF
9-05-97	1:26p	1140	0	private.dif
8-12-96	1:25a	10962	0	screen01.sys
8-12-96	1:25a	11436	0	screen02.sys
9-05-97	1:24p	43238	0	svgadata.pmi
9-05-97	1:24p	23	0	SVGATMP.BAT
9-05-97	1:28p	2764	0	video.cfg
9-05-97	1:26p	35	0	video.err
13 file(s)		424266 bytes used		

**Remarks:** This directory contains several files that are needed to create a Machine Class. They are the OS2.INI, OS2SYS.INI, PRIVATE.DIF, SVGADATA.PMI and VIDEO.CFG. files. These files are copied into the CPQ5120E.MC subdirectory as outlined in Figure 57 on page 168.

Directory of C:\OS2\BOOT

9-05-97	1:30p	2655	0	PREVIOUS.DAT
1 file(s)		2655 bytes used		

**Remarks:** This file is dependent on the detection level of the Hardware Manager Setting. This file gets recreated at every new boot time when the setting is not disabled in the Hardware manager. This file is not needed to create a Machine Class.

Figure 56 (Part 3 of 7). Files Changed by the Cirrus Logic Install Process

Directory of C:\OS2\DLL

8-12-96	2:02a	32750	49	BVHSVGA.DLL
8-12-96	2:07a	42387	49	bvhvga.dll
8-06-96	12:26p	416668	0	cirrus.dll
8-12-96	5:40a	44196	0	display.dll
9-05-97	1:31p	146	0	dock0.cfg
9-05-97	1:31p	46	0	dock1.cfg
9-05-97	1:31p	46	0	dock10.cfg
9-05-97	1:31p	46	0	dock11.cfg
9-05-97	1:31p	46	0	dock12.cfg
9-05-97	1:31p	46	0	dock13.cfg
9-05-97	1:31p	46	0	dock14.cfg
9-05-97	1:31p	46	0	dock15.cfg
9-05-97	1:31p	46	0	dock2.cfg
9-05-97	1:31p	46	0	dock3.cfg
9-05-97	1:31p	46	0	dock4.cfg
9-05-97	1:31p	46	0	dock5.cfg
9-05-97	1:31p	46	0	dock6.cfg
9-05-97	1:31p	46	0	dock7.cfg
9-05-97	1:31p	46	0	dock8.cfg
9-05-97	1:31p	46	0	dock9.cfg
8-09-96	4:20a	249590	333	dsPRES.dll
8-09-96	3:07a	82891	333	ibmdev32.dll
9-05-97	1:31p	753	0	SCENTER.CFG
8-12-96	2:03a	60455	49	videopmi.dll
24 file(s)		930526 bytes used		

**Remarks:** Several files from this directory are needed to create a Machine Class. They are the CIRRUS.DLL, BVHSVGA.DLL, DISPLAY.DLL and the VIDEOPMI.DLL files. Although the other files were touched, during the SVGA installation, examining the PRE and POST SVGA.DAT files shows that these files have the same size and time stamp before and after the video installation. Therefore, only the four files mentioned above are copied into a PRIVATE video subdirectory (see section 8.3.1.6, "Create a Video Subdirectory" on page 169, for details).

Figure 56 (Part 4 of 7). Files Changed by the Cirrus Logic Install Process

Directory of C:\OS2\INSTALL

9-05-97	1:23p	34	0	CD.LOG
9-05-97	1:24p	2835	0	DSPINSTL.LOG
9-05-97	1:23p	120	0	install.err
9-05-97	1:23p	228	0	install.hst
9-05-97	1:24p	18312	0	INSTALL.LOG
9-05-97	1:23p	302	0	OS2MM.LOG
9-05-97	1:25p	1206	0	REINSTALL.INI
9-05-97	1:24p	308	0	winfile.ins
9-05-97	1:23p	20	0	WINOS2.LOG
9-05-97	1:27p	41650	0	WPINSTAL.INI
10 file(s)		65015 bytes used		

**Remarks:** These files are not needed to create a Machine Class, they are simply additional sources of installation and change information.

Directory of C:\OS2\MDOS

8-29-96	10:38p	147963	49	vsvga.sys
8-29-96	10:39p	65386	49	vvga.sys
2 file(s)		213349 bytes used		

**Remarks:** These drivers are standard DOS drivers available in OS/2 Warp Version 4 as well as WorkSpace On-Demand. They are not needed to create a Machine Class.

Figure 56 (Part 5 of 7). Files Changed by the Cirrus Logic Install Process

Directory of C:\OS2\MDOS\WINOS2

9-05-97	9:48a	460	0	ATM.BAK
9-05-97	1:23p	460	0	ATM.INI
5-10-96	4:02p	4076	0	CONTROL.BAK
9-05-97	9:48a	202	0	PROGMAN.BAK
3-06-96	8:28p	44	0	STARTUP.BAK
9-05-97	9:53a	1687	0	SYSTEM.BAK
9-05-97	1:30p	1890	0	system.ini
9-05-97	9:53a	4137	0	WIN.BAK
9-05-97	1:30p	4138	0	win.ini
3-06-96	8:22p	13538	0	WOS2ACCE.BAK
3-06-96	8:22p	5091	0	WOS2MAIN.BAK
11 file(s)		35723 bytes used		

**Remarks:** BAK files are automatically created backup files and are not needed to create a Machine Class. Although the ATM.INI file was touched it, wasn't changed. However, we do need to take a closer look at the WIN.INI and SYSTEM.INI files, because they are needed to create a Machine Class.

Figure 56 (Part 6 of 7). Files Changed by the Cirrus Logic Install Process

```
Directory of C:\OS2\MDOS\WINOS2\SYSTEM
```

8-06-96	12:26p	116416	0	16ms640.drv
7-16-96	3:43p	133312	0	16m_640.drv
7-16-96	4:22p	121616	0	256s1280.drv
7-16-96	3:43p	151424	0	256_1280.drv
7-16-96	4:22p	120592	0	64ks1024.drv
7-16-96	3:43p	154000	0	64k_1024.drv
3-08-96	9:08p	10976	0	8514FIX.FON
3-08-96	9:08p	12272	0	85140EM.FON
3-08-96	9:08p	9264	0	8514SYS.FON
9-05-97	1:24p	25948	0	CONTROL.INF
3-08-96	9:08p	31712	0	COURF.FON
3-06-96	7:40p	34523	0	MIDIMAP.BAK
3-08-96	9:08p	81728	0	SERIFF.FON
3-08-96	9:09p	21504	0	SMALLF.FON
3-08-96	9:08p	89680	0	SSERIFF.FON
3-06-96	6:39p	81504	0	SYMBOLF.FON
16 file(s)		1196471 bytes used		

**Remarks:** These are SVGA Driver and Font files and they are common to all machines. Therefore, these files don't need to be copied into PRIVATE directories.

Figure 56 (Part 7 of 7). Files Changed by the Cirrus Logic Install Process

### 8.3.1.5 Machine Class Directory Structure

What follows is a diagram of the CPQ5120E subdirectory structure. This is where you place all of the *REQUIRED* files identified by examining the ATTRIB.DAT file in the previous steps.

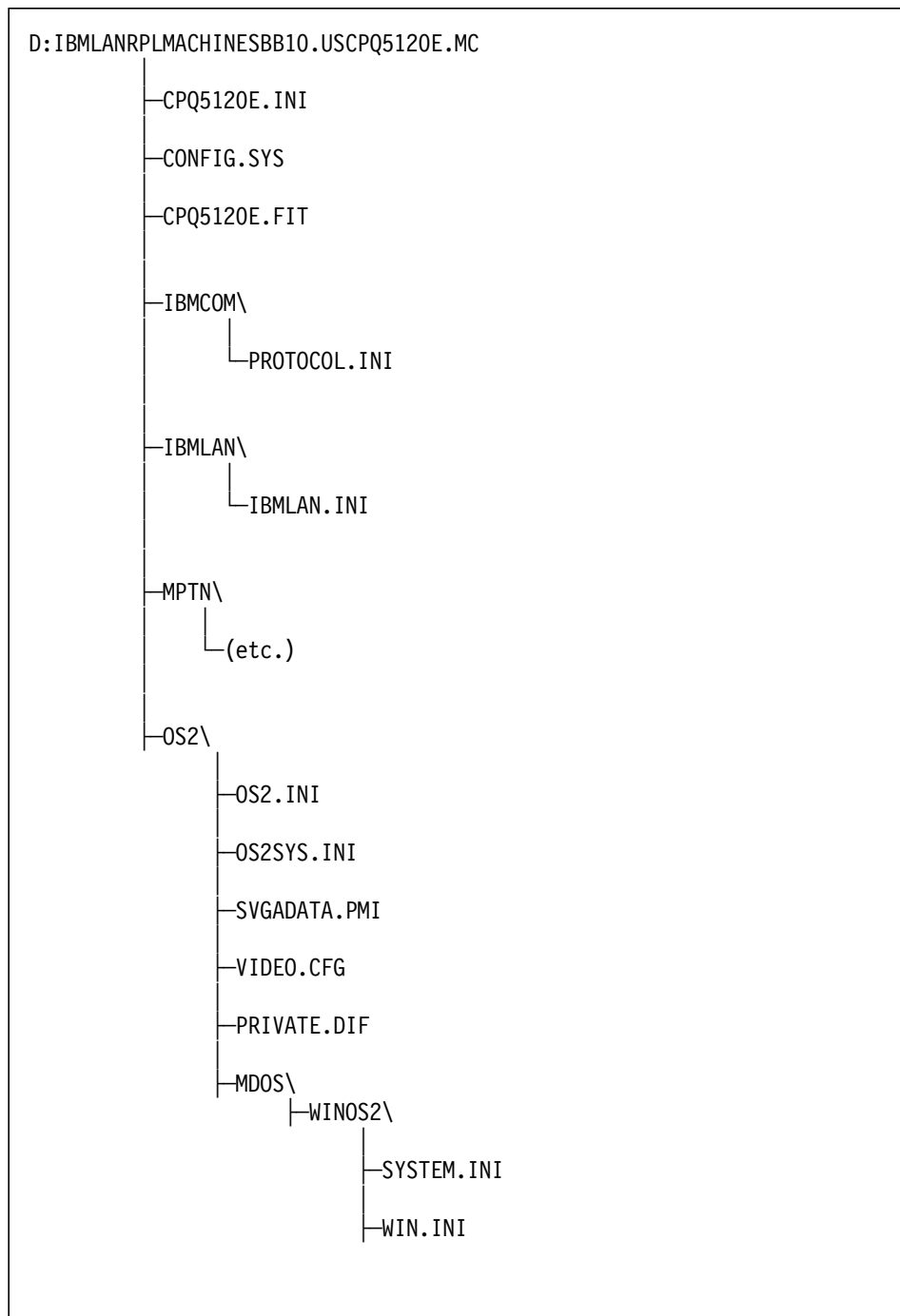


Figure 57. CPQ5120E Machine Class Subdirectory

### 8.3.1.6 Create a Video Subdirectory

As you might have suspected, not all of the marked files in ATTRIB.DAT are placed in the Machine Class subdirectory. The video driver files are placed under the WorkSpace On-Demand RIPL tree. For this, we create a new directory for the Cirrus Logic 5434 drivers:

```
CD D:\IBMLANRPLBB10\USOS2\VIDEO
MD CL5434
```

Next, copy the display driver files CIRRUS.DLL, DISPLAY.DLL, BVHSVG.A.DLL, and VIDEOPMI.DLL into this directory. We also renamed the CIRRUS.DLL to CL5434.DLL for better readability. This is especially desirable if you have to install several versions of display drivers in order to support multiple Machine Classes.

### 8.3.1.7 Create a FIT Extension

The FIT file, CPQ5120E.FIT, provides the system with a remapping facility pointing to the new location of the video drivers as shown in the following example:

```
; support for the CPQ5120E Machine Class

; video support for the Cirrus Logic 5434 driver

; writeable files

Z:\OS2\PRIVATE.*      \\BBSRV01\WRKFILES\DEFAULT\OS2
Z:\OS2\SVGADATA.*     \\BBSRV01\WRKFILES\DEFAULT\OS2
Z:\OS2\VIDEO.*        \\BBSRV01\WRKFILES\DEFAULT\OS2

; readonly files

Z:\OS2\DLL\BVHSVG.A.DLL  BB10.US\OS2\VIDEO\CL5434\BVHSVG.A.DLL
Z:\OS2\DLL\CIRRUS.DLL    BB10.US\OS2\VIDEO\CL5434\CL5434.DLL
Z:\OS2\DLL\DISPLAY.DLL  BB10.US\OS2\VIDEO\CL5434\DISPLAY.DLL
Z:\OS2\DLL\VIDEOPMI.DLL BB10.US\OS2\VIDEO\CL5434\VIDEOPMI.DLL
Z:\OS2\DLL\IBMGPMI.DLL  BB10.US\OS2\VIDEO\CL5434\IBMGPMI.DLL
```

Figure 58. CPQ5120E.FIT File

### 8.3.1.8 Create an Updated CONFIG.SYS

In order for the new video drivers to be loaded at startup time, CONFIG.SYS must be modified. To determine exactly which lines were changed due to the Cirrus installation, use a file comparison tool to show which lines are

different in both versions of the CONFIG.SYS files. Figure 59 on page 170 shows the changed lines in our example.

```
REM **** NCVIDEO BEGIN ****
DEVINFO=SCR,VGA,Z:\OS2\BOOT\VIOTBL.DCP
SET VIDEO_DEVICES=VIO_SVGA
SET VIO_SVGA=DEVICE(BVHVGA,BVHSVGA)
DEVICE=Z:\OS2\MDOS\VVGA.SYS
REM **** NCVIDEO END ****
```

Figure 59. CONFIG.SYS Changes

### 8.3.1.9 Create an Updated OS2.INI

Use an INI editor for a detailed look into the OS2.INI files. Because OS2.INI contains binary data, several REXX command files that manipulate the content of individual OS2.INI files are provided. The source for these REXX command files can be found in Appendix F, “Tools for Modifying and Managing CONFIG.SYS and INI Files in Machine Classes” on page 281. Use the following procedures to make and apply all the necessary changes to the OS2.INI:

1. First, use the INICREATEDELTA4.CMD file to create a DELTA.INI file.  
This command file requires as input two INI files. In this example we have called these OS2VGA.INI and OS2SVGA.INI. The OS2VGA.INI file is the copy of the OS2.INI file from the OS/2 Warp Version 4 system that supported the video adapter in VGA mode, and the OS2SVGA.INI file is the copy of the OS2.INI file from the same OS/2 Warp Version 4 system after the SVGA support for the video adapter being classed was added.  
  
The INICREATEDELTA4.CMD procedure looks for any new or changed application data, keywords or data entries in the OS2SVGA.INI and puts those into the DELTA.INI file.
2. Next, edit the DELTA.INI file manually with an INI file editor and insure that only necessary changes remain in the new CPQ5120E's OS2.INI. In this case, the following two application entries are required:
  - PM\_DISPLAYDRIVERS  
This particular application is required for all the Cirrus Logic device information. Also, make sure that there is no keyword, like DEFAULTSYSTEMRESOLUTION, in the file. If there is, delete this keyword. You should also manually add the keyword RESOLUTION\_CHANGED with a data entry of 1.
  - WIN\_RES\_...



There are several entries of this kind which are required by the WINOS2 environment in order to change to another video resolution.

3. Run the INIMERGE.CMD using the target's (CPQ5120E) OS2.INI and the DELTA.INI file names as input to the command. This command merges the delta into the new Machine Class INI file.

These steps produce a new OS2.INI file that is used in the new Machine Class subdirectory.

#### **8.3.1.10 Create an Updated OS2SYS.INI**

Always check for changes in OS2SYS.INI file as well. However, this specific installation didn't change any entries in the file; so the update procedures are unnecessary in this case.

#### **8.3.1.11 Create an Updated WIN.INI**

This file, unlike OS2.INI, is a pure ASCII file and does not contain binary data. Since it is an ASCII file, use a file comparison tool to determine what changes were made to WIN.INI. In this case, the following changes were discovered and must be added to the new, updated WIN.INI.

```
[Desktop]
IconSpacing=100
```

*Figure 60. WIN.INI Changes*

#### **8.3.1.12 Create an Updated SYSTEM.INI**

Use a file comparison tool again to determine what changes were made to the SYSTEM.INI file. In this case, the following changes were discovered and should be added to the updated SYSTEM.INI:

```

[boot]
display.drv=256_1280.drv
sdisplay.drv=256s1280.drv

[boot.description]
display.drv=Cirrus 54xx Accelerated 640x480x256
sdisplay.drv=Cirrus 54xx Accelerated 640x480x256

[256s1280.drv]
fontsize=96
dacdepth=6
Enable216=0
WidthxHeight=640x480

[c1vga]
videomode=0x5f
logo=0

```

Figure 61. *SYSTEM.INI* Changes

With these changes in place, the new CPQ5120E Machine Class can now be used to create new RIPL requesters using the standard administration tools as described in Chapter 6, “Basic Workspace On-Demand Administration” on page 95.

### 8.3.2 Creating a Machine Class with SCSI Support

The default Machine Classes have no support for SCSI controllers or SCSI devices. To enable SCSI support, the default classes have to be modified. Adding support for SCSI hard drives is reasonably straightforward. This section describes adding support for hard drives; other SCSI devices may require additional steps.

When adding SCSI support to a base OS/2 installation, the system copies the drivers from the installation source and adds the device statements for the drivers to the CONFIG.SYS. No changes are made to any of the other system files.

Therefore, to add SCSI support to a Machine Class, you need to do the following:

- Create the Machine Class subdirectory structure by copying it from an existing structure.
- Create an INI File for the new Machine Class.
- Determine the drivers copied and the CONFIG.SYS changes.

- Copy the drivers into the WorkSpace On-Demand directory structure.
- Modify the FIT file to include the SCSI drivers.
- Update the base CONFIG.SYS.

#### 8.3.2.1 Adding the FutureDomain TMC-850

As an example, the FutureDomain TMC-850 SCSI card and IBM350 will be used as a client machine. OS/2 Warp Server is installed on drive C:, and the directory containing the new Machine Class has already been created using the name C:\BMLANRPL\MACHINES\BB10.US\350WSCSI.MC. See 8.2.3, “Creating a New Machine Class’s Directory Structure” on page 156, for more information on creating the directory structure. The FIT file has been renamed to 350WSCSI.FIT and the Machine Class INI file renamed to 350WSCSI.INI. It has been verified that this Machine Class works correctly prior to the modifications.

##### Note

The client machine has PROTDISK.SYS in the CONFIG.SYS file. This driver does not allow local file access on the client machines. With this driver, the SCSI device will only be used for the swap file. If you want the clients to have access to the local SCSI drive, this driver must be removed from the CONFIG.SYS.

#### 8.3.2.2 Creating an INI File for the New Machine Class

The first phase is to enable the new Machine Class to appear in the WorkSpace On-Demand Manager administration GUI as a new, selectable entry. The GUI information is provided by the 350WSCSI.INI file in the D:\BMLAN\RPL\MACHINES\BB10.US\350WSCSI.MC subdirectory. Before working with the file, you need to rename the copied IBM350.INI to 350WSCSI.INI.

Using an INI-file editor, you have to make sure that all data entries for the *Machine\_Class* application are correct. This is same as described in 8.3.1.2, “Creating an INI File for the New Machine Class” on page 158. See this section for the details.

#### 8.3.2.3 Determining the Differences

To determine what changes you need to make, you can go through the same process as with applying video drivers. In fact, this is recommended as it will provide you with more conclusive evidence of what has changed. However, since SCSI drivers make minimal changes, you may find that you do not need to use that procedure. It is also possible that the machine’s

hard disk is attached to the SCSI controller, in which case it is difficult to view a CONFIG.SYS that does not contain SCSI support.

With SCSI drivers, there are two activities you need to perform. The first is to change the CONFIG.SYS file, and the second is to copy the device drivers.

If you are not sure which are the SCSI support statements, then you will need a copy of a CONFIG.SYS that does not have SCSI support. You can then compare this to a CONFIG.SYS that has SCSI support installed and extract the difference. Even if the CONFIG.SYS is not from the same machine type, you still should be able to tell which are SCSI statements and which belong to other hardware categories.

In most cases, adding SCSI support adds just two statements for drivers to the CONFIG.SYS (three for Micro Channel). The first driver is one that is specific to the SCSI adapter installed in the workstation, and the second is an OS/2 SCSI driver.

For example, if you loaded support for an Adaptec 2940 SCSI Adapter, you would see the following statements added in your CONFIG.SYS.

```
BASEDEV=OS2SCSI.DMD  
BASEDEV=AHA7870.ADD /A:0
```

The driver AHA7870.ADD is adapter specific, and the OS2SCSI.DMD is an OS/2 SCSI driver. SCSI drivers that are shipped with OS/2 Warp include:

- OS2SCSI.DMD
- IBM2SCSI.ADD
- OS2ASPI.DMD
- VASPI.SYS
- DELIVERY.SYS

These drivers are installed in the \OS2\BOOT subdirectory and are loaded by default in the CONFIG.SYS as BASEDEV= statements.

OS2ASPI.DMD, and VASPI.SYS are used only if ASPI support is required.

DELIVERY.SYS is a driver that is only needed if you have an IBM SCSI-2 FAST/WIDE adapter installed in your system. It is loaded as a BASEDEV= statement in your CONFIG.SYS.

For standard SCSI support, you require an adapter-specific driver with either OS2SCSI.DMD and/or IBM2SCSI.ADD. OS2SCSI.DMD is a device manager

which provides a hardware-independent interface to device drivers written to IBM's SCSI standard.

IBM2SCSI.ADD supports IBM PS/2 SCSI adapters for IBM Micro Channel SCSI controllers.

In the example for the Future Domain Adapter, the analysis of the OS/2 Warp Version 4 system showed that the following lines were added to the CONFIG.SYS and that the OS2SCSI.DMD and FD8XX.ADD device drivers were copied to the \OS2\BOOT directory.

```
BASEDEV=OS2SCSI.DMD
BASEDEV=FD8XX.ADD
```

#### 8.3.2.4 Copy the SCSI Device Drivers

The two SCSI device drivers that were installed in the C:\OS2BOOT directory of the OS/2 Warp Version 4 system need to be copied to the C:\IBMLANRPLBB10.US\OS2BOOT directory, where the WorkSpace On-Demand clients load the drivers from. These drivers are added to the CONFIG.SYS file with a BASEDEV statement. OS/2 searches the drivers that are listed with the BASEDEV statement in this directory.

```
COPY FD8XX.ADD C:\IBMLANRPLBB10.US\OS2BOOT
COPY OS2SCSI.DMD C:\IBMLAN\RPL\BB10.US\OS2\BOOT
```

#### 8.3.2.5 Create a FIT Extension

The FIT-file provides the system with a remapping facility pointing to the new location of the SCSI drivers. The FIT file is located in C:\IBMLANRPLMACHINESBB10.US\350WSCSI.MC directory. The following example shows the lines added to the FIT file:

```
; support for the IBM350 w/ Future Domain SCSI card Machine Class

Z:\OS2\BOOT\FD8XX.ADD      BB10.US\OS2\BOOT\FD8XX.ADD
Z:\OS2\BOOT\OS2SCSI.DMD    BB10.US\OS2\BOOT\OS2SCSI.DMD
```

Figure 62. 350WSCSI.FIT File

#### Note

Files in the OS2BOOT directory do not need a FIT mapping, they are handled via the default FIT. This FIT example is shown as an example of what might be required for other cases.

### 8.3.2.6 Create an Updated CONFIG.SYS

In order for the new SCSI drivers to be loaded at start up time, CONFIG.SYS must be modified. To determine exactly which lines were changed due to the SCSI installation, use a file comparison tool to show which lines are different in both versions of the CONFIG.SYS files. Figure 63 shows what was changed in the CONFIG.SYS for the FutureDomain TMC-850 SCSI adapter.

```
BASEDEV=OS2SCSI.DMD
BASEDEV=FD8XX.ADD
```

Figure 63. CONFIG.SYS Changes

For this example, all the above steps will complete the Machine Class for FutureDomain TMC-850. You can use this Machine Class in the GUI, and the client defined with this Machine Class can use the SCSI hard drive for the swap file.

### 8.3.2.7 SCSI Adapter Drivers

Table 11 shows some of the more common adapters and their driver names. If the adapter that you have is not listed, you can probably locate it from the OS/2 device driver repository on the Web or from the manufacturer.

Table 11 (Page 1 of 2). SCSI Adapters		
Manufacturer	Driver Name	Adapter
Adaptec	AHA152X.ADD	AHA-1510/1520/1522 ISA SCSI AIC-6350/6360
Adaptec	AHA154X.ADD	AHA-1540/1542 ISA SCSI
Adaptec	AHA164X.ADD	AHA-1640 Machine Class SCSI host adapter
Adaptec	AHA174X.ADD	AHA-1740/1742/1744 EISA SCSI host adapter
Adaptec	AIC7770.ADD	AHA-2840VL/2842VL ISA SCSI AHA-2740/2742 EISA SCSI AIC-7770
Adaptec	AIC7870.ADD	AHA-2940 PCI SCSI AHA-2940W AIC-7870
BusLogic	BTSCSI.ADD	BusMaster SCSI Host Adapter <ul style="list-style-type: none"><li>• BT-445S ISA VL SCSI</li><li>• BT-542B, BT-542S ISA SCSI</li><li>• BT-640A, BT646S Machine Class SCSI</li><li>• BT-742A, BT-747S EISA SCSI</li><li>• BT-946 PCI SCSI</li></ul>

Table 11 (Page 2 of 2). SCSI Adapters

Manufacturer	Driver Name	Adapter
DPT	DPTM20XX.ADD	PM2011/2012
Future Domain	FD8XX.ADD	845, 850 860, 875, 885
Future Domain	FD16-700.ADD	16xx, 1790, 1795 1800/18C30/18C50 3260/36C70 MCS600/700
Future Domain	FD7000EX.ADD	TMC-7000EX
IBM	FD850IBM.ADD	TMC 850IBM
IBM	FD16IBM.ADD	IBM 16-bit ISA SCSI Adapter
IBM	IBM2SCSI.ADD	IBM 16 or 32-bit Machine Class Adapter
IBM	DELIVERY.SYS	32-bit Fast and Wide SCSI Machine Class Adapters
MediaVision	TMV1SCSI.ADD	ProAudio Spectrum with Trantor SCSI

For further information on the SCSI drivers, see "Miscellaneous OS/2 Warp Device Drivers" in *The Guide to OS/2 Warp Device Drivers*, SG24-4627, as well as the online help.

### 8.3.3 Creating a Machine Class for a CD-ROM

This section describes how to create a Machine Class that enables support for a CD-ROM drive. The procedure you need to follow here is similar to the one used to enable video support. See 8.3.1, "Creating a Machine Class for a Specific Video Adapter" on page 157, for the details.

CD-ROM support is similar to SCSI support. When adding CD-ROM support to a base OS/2 installation, the system copies the drivers from the installation source and adds the device statements for the drivers to the CONFIG.SYS. No changes are made to any of the other system files.

Therefore, to add CD-ROM support to a Machine Class, you need to do the following:

- Create the Machine Class subdirectory structure by copying it from an existing structure.
- Create an INI File for the new Machine Class.
- Determine the drivers copied and the CONFIG.SYS changes.
- Copy the drivers into the WorkSpace On-Demand directory structure.

- Modify the FIT file to include the CD-ROM drivers.
- Update the base CONFIG.SYS.

### 8.3.3.1 Adding an IDE CD-ROM

For the purposes of this example, the ISAVGA Machine Class will be used as a base. In this example, OS/2 Warp Server is installed on drive C:, and the directory structure of the prototype Machine Class is already created using the name C:\BMLANRPL\MACHINES\BB10.US\ISACD.MC. See 8.2.3, “Creating a New Machine Class’s Directory Structure” on page 156, for details on creating the directory structure. The FIT file is renamed ISACD.FIT.

#### Note

- The PROTDISK.SYS driver in the CONFIG.SYS file does not allow access to any files on the client machine’s local hard drive, but it does allow you to read files from the CD-ROM.
- If your CD-ROM is an SCSI device, start by creating a SCSI-enabled Machine Class first, and then modify it for CD-ROM support. See 8.3.2, “Creating a Machine Class with SCSI Support” on page 172, for Machine Class for SCSI support.

### 8.3.3.2 Creating an INI File for the New Machine Class

The INI file is required to enable the new Machine Class to appear in the WorkSpace On-Demand Manager administration GUI as a selectable entry. The GUI information is provided by the ISACD.INI file in the D:\BMLAN\RPL\MACHINES\BB10.US\ISACD.MC subdirectory. Before working with the file, you need to rename the copied ISAVGA.INI to ISACD.INI.

Using an INI file editor, you have to make sure that all data entries for the *Machine\_Class* application are correct. See 8.3.1, “Creating a Machine Class for a Specific Video Adapter” on page 157, for further details.

### 8.3.3.3 Determining the Differences

As described in the video support section, you can find the difference by monitoring the files before and after the CD-ROM support installation on the client machine. Very often, a machine is installed using the CD-ROM drive. Installing the machine over the network will give you the opportunity to deselect CD-ROM Support and create a before picture. Then, using selective install, you can note the differences.



The first step in analyzing what has changed is to look at the files that were added or updated in the install process. ATTRIB.DAT shows a list of all the files touched by the installation process. In this example, for the IDE CD-ROM, the files that were changed are shown in Figure 64 on page 179.

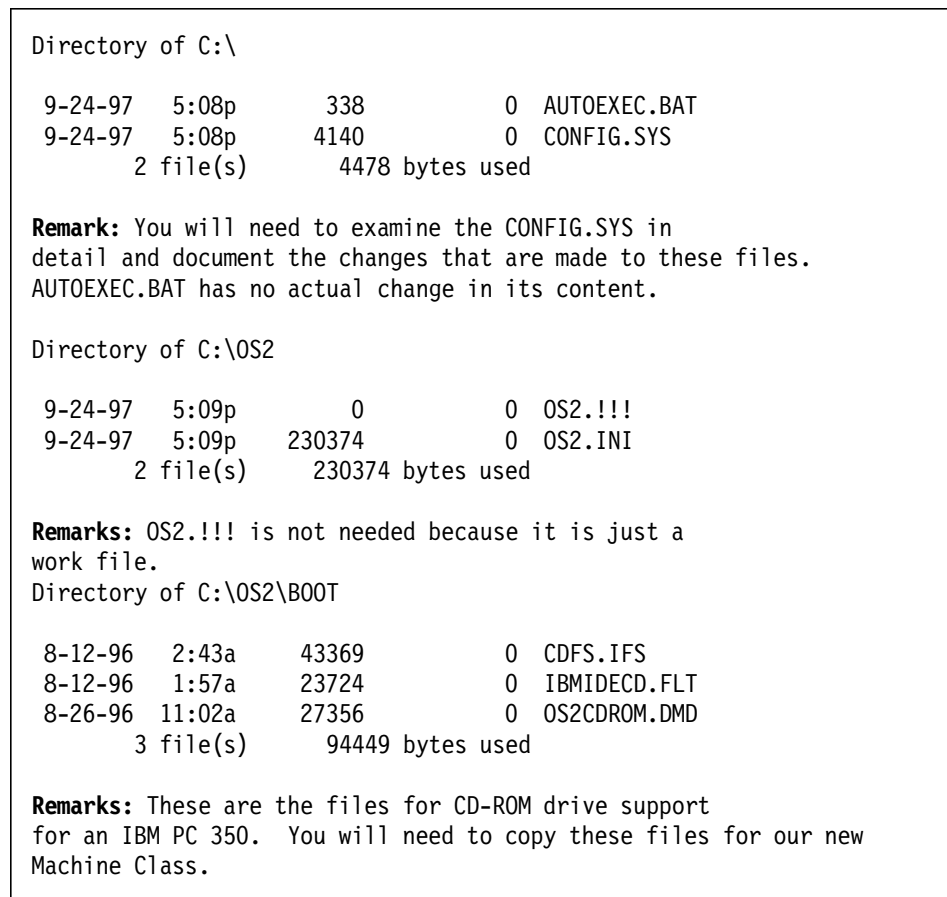


Figure 64 (Part 1 of 3). Files Changed by the CD-ROM Install Process

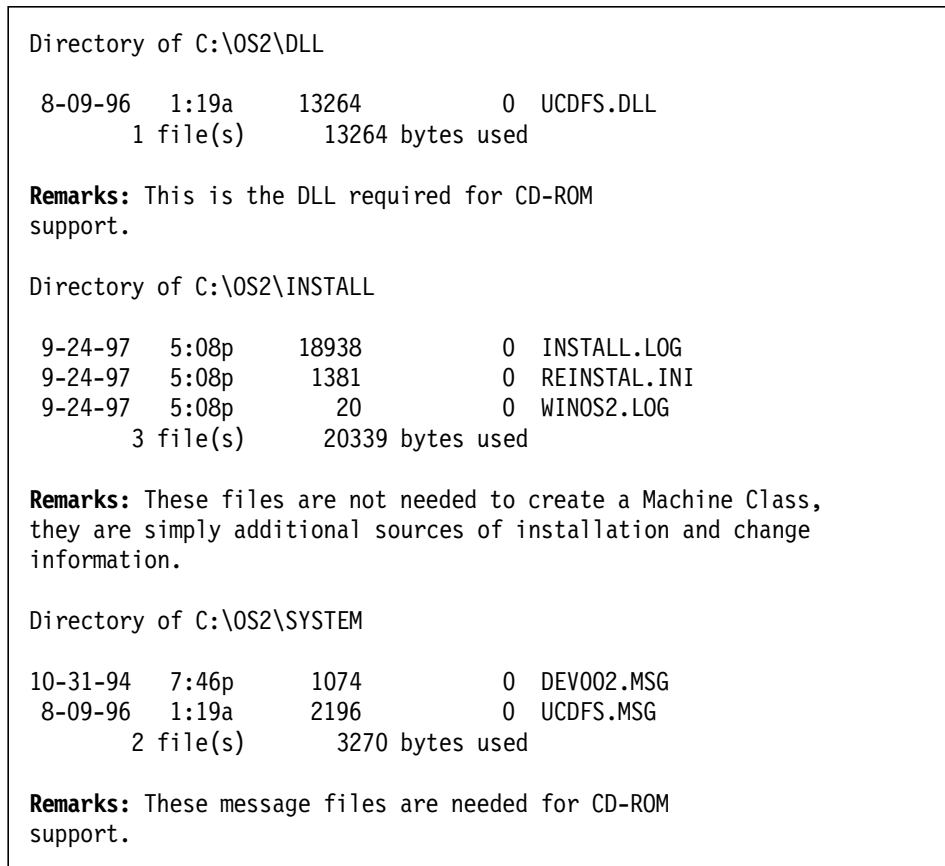


Figure 64 (Part 2 of 3). Files Changed by the CD-ROM Install Process

### 8.3.3.4 Copy the CD-ROM Support Files

The CD-ROM device drivers and other files copied by the CD-ROM installation is listed in the ATTRIB.DAT file. You need to copy these drivers to C:\IBMLANRPLBB10.USOS2BOOT directory, where the WorkSpace On-Demand clients loads the drivers from. Following are the copy statements that copies files needed for adding the CD-ROM for the ISAVGA example:

```
COPY CDFS.IFS C:\IBMLANRPLBB10.USOS2BOOT
COPY IBMIDCD.FLT C:\IBMLAN\RPL\BB10.US\OS2\BOOT
COPY OS2CDROM.DMD C:\IBMLAN\RPL\BB10.US\OS2\BOOT
COPY DEV002.MSG C:\IBMLAN\RPL\BB10.US\OS2\SYSTEM
COPY UCDFS.MSG C:\IBMLAN\RPL\BB10.US\OS2\SYSTEM
COPY UCDFS.DLL C:\IBMLAN\RPL\BB10.US\OS2\DLL
```

### 8.3.3.5 Create a FIT Extension

The FIT file provides the system with a remapping facility pointing to the location of the CD-ROM drivers. The FIT file is located in the C:\IBMLANRPLMACHINESBB10.USISACD.MC directory. Figure 65 shows the lines added to the FIT file:

```
; support for the ISA w/ CD-ROM drive Machine Class

Z:\OS2\BOOT\CDFS.IFS          BB10.US\OS2\BOOT\CDFS.IFS
Z:\OS2\BOOT\IBMIDEC.DFLT      BB10.US\OS2\BOOT\IBMIDEC.DFLT
Z:\OS2\BOOT\OS2CDROM.DMD      BB10.US\OS2\BOOT\OS2CDROM.DMD
Z:\OS2\DLL\UCDFS.DLL          BB10.US\OS2\DLL\UCDFS.DLL
Z:\OS2\SYSTEM\DEV002.MSG      BB10.US\OS2\SYSTEM\DEV002.MSG
Z:\OS2\SYSTEM\UCDFS.MSG       BB10.US\OS2\SYSTEM\UCDFS.MSG
Z:\OS2\MDOS\VCDROM.SYS        BB10.US\OS2\VCDROM\VCDROM.SYS
```

**Remarks:** VCDROM.SYS shown in the last line is the virtual DOS device driver. This is already installed with WorkSpace On-Demand as default, and it is needed when DOS support is required on the client.

Figure 65. ISACD.FIT File

#### Note

Files in the OS2BOOT, OS2DLL and OS2SYSTEM directory structures do not need a FIT mapping, they are handled via the default FIT. Only the last entry for the VCDROM.SYS is required because it is being remapped to a different directory path. All but the last entry in this FIT example are shown only as an example of what might be required for other cases.

### 8.3.3.6 Create an Updated CONFIG.SYS

In order for the new CD-ROM drivers to be loaded at startup time, CONFIG.SYS must be modified. Four device statements are needed in your CONFIG.SYS file. The first driver is dependent on the interface being used. These may already be in your CONFIG.SYS to support other devices such as hard drives. An example of these drivers are:

The first is specific to the bus type that to which the CD-ROM is attached:

BASEDEV=IBM1S506.ADD (for IDE based CD-ROM support)

or

BASEDEV=OS2SCSI.ADD (for SCSI based CD-ROM support)

The next three are specific to CD-ROM support:

```
IFS=z:\OS2\BOOT\CDFS.IFS /Q
DEVICE=z:\OS2\BOOT\OS2CDROM.DMD /Q
DEVICE=z:\OS2\MDOS\VCDROM.SYS
```

Along with the above four statements, you also need to have the specific driver for your CD-ROM loaded as a BASEDEV= statement. For example, if you have a Panasonic 562 CD-ROM, your statement would be:

```
BASEDEV=SBCD2.ADD /P:220
```

The CDFS.IFS driver is a file system driver used to load the CD-ROM file system or CDFS. It is normally located in the \OS2\BOOT subdirectory.

OS2CDROM.DMD is a device manager that provides support for CD-ROM drives, and VCDROM.SYS provides the virtual support.

Use a file comparison tool to determine which lines have been changed. Figure 66 shows the changes that were made in the CONFIG.SYS for the CD-ROM drive that is the example in this chapter.

```
DEVICE=C:\OS2\BOOT\OS2CDROM.DMD /Q
IFS=C:\OS2\BOOT\CDFS.IFS /Q
DEVICE=C:\OS2\MDOS\VCDROM.SYS
BASEDEV=IBMIDEC.D.FLT
```

*Figure 66. CONFIG.SYS Changes*

The IBMIDEC.D.FLT driver provides support for Enhanced IDE CD-ROM drives following the ATAPI Industry Specification 1.2. This includes the following CD-ROM drives:

- Sony CDU55E
- Philips LMSCM207
- Mitsumi FX001DE

Once you have completed the steps in this section, the CD-ROM should be accessible from the workstations defined to the ISACD Machine Class.

### 8.3.4 Mouse Considerations

This section discusses the mouse driver and the various configurations that are available. The default WorkSpace On-Demand Machine Classes all have PS/2-style mouse configurations. Today, there are hundreds of mouse devices available. Most of them are Microsoft or PS/2 compatible, and you can use the drivers as they are in the default Machine Classes.

However, if the Machine Class you are creating does not have a PS/2 or Microsoft compatible mouse attached, this section describes the mouse drivers and related files that you can use to provide the needed mouse support.

The default MOUSE.SYS driver tries to detect what kind of pointing device you have attached to your system. It checks the pointing devices in the following order:

1. Pointing Device Interface (PDI) or PS/2 Style port
2. COM1 port
3. COM2 port
4. ISA bus In-port
5. Bus card

MOUSE.SYS supports both relative and absolute pointing devices. A relative pointing device is one that causes the pointer to move across the screen relative to the motion of the device. An absolute pointing device sends information corresponding to a screen location where the pointer should appear (for example, a touch screen).

The files which OS/2 Warp uses for the mouse access are listed in Table 12.

<i>Table 12. Mouse Drivers</i>	
<b>Name</b>	<b>Description</b>
VMOUSE.SYS	Virtual mouse driver for DOS applications
POINTDD.SYS	Provides general mouse pointer draw support
PMDD.SYS	Provides mouse pointer draw support for OS/2 sessions
MOUSE.DRV	Mouse driver for Windows applications
MOUSE.SYS	Provides support for pointing devices
MOUCALLS.DLL	Dynamic link library for mouse calls

The following drivers are used in all the default Machine Classes:

```
OS2MDOSVMOUSE.SYS
\OS2\BOOT\POINTDD.SYS
\OS2\BOOT\MOUSE.SYS
\OS2\BOOT\PMDD.SYS
```

In addition to the drivers above, the MOUSE.DRV driver for Windows applications and MOUCALLS.DLL library are also installed into the default image.

If the pointing device was delivered with its own device driver or is not supported by the MOUSE.SYS driver, you can use the TYPE=xxxxx\$ statement to install the device. For example, some older Logitech devices are not detectable by MOUSE.SYS. However, all versions of OS/2 still support these devices using the PCLOGIC.SYS driver. An example of such a statement in a CONFIG.SYS file is shown below:

```
DEVICE=C:\OS2\PCLOGIC.SYS SERIAL=COM1
DEVICE=C:\OS2\MOUSE.SYS TYPE=PCLOGIC$
```

The TYPE=PCLOGIC\$ statement refers to a device-dependent driver called PCLOGIC.SYS. This driver has a separate parameter called SERIAL=COM1 that connects the mouse to COM1.

The PCLOGIC.SYS driver does not ship with WorkSpace On-Demand. Should you require this driver or any other driver that your mouse may require, you will need to copy it into the x:IBMLANRPLBB10.USOS2BOOT directory.

Alternatively, if you would like to keep the WorkSpace On-Demand image clean, you could create a directory for the driver and insert a FIT file entry for it. This is the approach used in the Machine Class example for the Cirrus video adapter, where drivers were installed into a *PRIVATE* video directory. An example is shown in Figure 67.

Z:OS2BOOTPCLOGIC.SYS BB10.USOS2MOUSEPCLOGIC.SYS

*Figure 67. Mouse Drivers. Mouse Driver FIT File Redirection*

The following pointing devices have been tested and verified to work. Included are the CONFIG.SYS statements for each entry.

<i>Table 13 (Page 1 of 2). Supported Pointing Devices</i>	
<b>Device Name</b>	<b>CONFIG.SYS Statements</b>
IBM PS/2 Mouse	DEVICE=C:\OS2\MOUSE.SYS
MS PS/2 Mouse	DEVICE=C:\OS2\MOUSE.SYS
MS Serial Mouse	DEVICE=C:\OS2\MOUSE.SYS
Kensington Expert PS/2 Mouse	DEVICE=C:\OS2\MOUSE.SYS
Logitech PS/2 Mouse	DEVICE=C:\OS2\MOUSE.SYS
MS Bus Mouse	DEVICE=C:\OS2\MOUSE.SYS
MS Inport Mouse	DEVICE=C:\OS2\MOUSE.SYS
Logitech Serial Mouse (Series M)	DEVICE=C:\OS2\MOUSE.SYS

Table 13 (Page 2 of 2). Supported Pointing Devices

Device Name	CONFIG.SYS Statements
Logitech Serial Mouse (Series C)	DEVICE=C:\OS2\PCLOGIC.SYS SERIAL=COM1 DEVICE=C:\OS2\MOUSE.SYS TYPE=PCLOGIC\$
Logitech Trackman serial mouse	DEVICE=C:\OS2\PCLOGIC.SYS SERIAL=COM2 DEVICE=C:\OS2\MOUSE.SYS TYPE=PCLOGIC\$
PC Mouse Systems serial mouse	DEVICE=C:\OS2\PCLOGIC.SYS SERIAL=COM2 DEVICE=C:\OS2\MOUSE.SYS TYPE=PCLOGIC\$
PC Mouse Systems bus mouse	DEVICE=C:\OS2\MSBUS01.SYS DEVICE=C:\OS2\MOUSE.SYS TYPE=MSBUS\$
Visi-On serial mouse	DEVICE=C:\OS2\PCLOGIC.SYS SERIAL=COM2 DEVICE=C:\OS2\MOUSE.SYS TYPE=PCLOGIC\$
<b>Note:</b> The mouse driver is smart enough to figure out if the mouse is in a mouse port or in one of the COM ports. Be sure that the MOUSE.SYS statement appears before DEVICE=C:\OS2\COM.SYS in your CONFIG.SYS file if the mouse is in a COM port.	

For further information on the mouse driver, see "Miscellaneous OS/2 Warp Device Drivers" in *The Guide to OS/2 Warp Device Drivers*, SG24-4627.

### 8.3.5 Keyboard Considerations

This section discusses the keyboard driver. The default WorkSpace On-Demand Machine Classes all have IBM Enhanced keyboards attached. Although keyboards are fairly standard, this section discusses the keyboard driver and related files needed if the Machine Class being created requires non-standard keyboard support.

The files WorkSpace On-Demand uses to access the keyboard are shown in Table 14.

Table 14 (Page 1 of 2). Keyboard Files

Name	Description
IBMKBD.SYS	This is the hardware-specific portion of the keyboard device driver. It provides support for the keyboards listed in 8.3.5.1, "Supported Keyboards" on page 186. It also gets the scan codes and passes them to the KBDBASE.SYS driver.
KBDBASE.SYS	The base OS/2 keyboard driver. It processes the scan codes sent from the IBMKBD.SYS driver and instructs IBMKBD.SYS if there are any updates.
VKBD.SYS	The DOS virtual keyboard driver is used by DOS application.
KEYBOARD.DRV	This is the Windows Keyboard driver. It is used by Windows applications.

Table 14 (Page 2 of 2). Keyboard Files

Name	Description
ANSI.SYS	Provides extended screen/keyboard support for DOS sessions.
KBDCALLS.DLL	DLL for keyboard calls.
BKSCALLS.DLL	Basic keyboard dynamic link library.

### 8.3.5.1 Supported Keyboards

The IBMKBD.SYS driver supports the following keyboards:

- IBM Enhanced 101/102-key keyboard
- IBM 3279 Enhanced 130-key keyboard
- Honeywell 101 WN
- Lexmark Enhanced 101-key keyboard
- Mitsumi 101/102-key keyboard Model-R , -S, -U, KPQ-E99ZC

#### Note

Any keyboard compatible with the IBM Enhanced 101 Key keyboard will be supported.

**The DEVINFO=KBD Statement:** The DEVINFO=KBD statement in the CONFIG.SYS file prepares a keyboard for system code-page switching. It specifies a keyboard layout and a file named KEYBOARD.DCP that contains a keyboard table for translating keystrokes into characters. The syntax of this statement is shown below:

DEVINFO=KBD,<layout>,<drive path filename>

<layout>	:Specifies the keyboard layout.
<drive path filename>	:Specifies the complete name of the file that contains the keyboard translation table (*.DCP).

An example of the use of the DEVINFO=KBD statement is shown below:

DEVINFO=KBD,US,C:\OS2\BOOT\KEYBOARD.DCP

This statement prepares a U.S keyboard layout, but it does not switch the keyboard yet. To switch the keyboard to another layout, use the KEYB command. A list of all available layouts is shown in Table 15 on page 187.



<i>Table 15. Keyboard Layouts</i>			
Parameter	Country Layout	Parameter	Country Layout
AR	Arabic	NL	Netherlands
BE	Belgium	NO	Norway
BR	Brazil	PL	Poland
CF	Canada (French)	PO	Portugal
CS	Czechoslovakia	SP	Spain
DK	Denmark	SV	Sweden
SU	Finland	SF	Switzerland (French)
FR	France	SG	Switzerland (German)
GR	Germany	TR	Turkey
HE	Hebrew	TW	China
HU	Hungary	UK	United Kingdom
IS	Island	US	United States
IT	Italy	YU	Yugoslavia
LA	Latin America		

Additionally, you can choose the subcountry code for those countries that have more than one keyboard layout. An example of such a configuration is shown below: `DEVINFO=KBD,FR120,C:\OS2\BOOT\KEYBOARD.DCP`

This prepares the enhanced French keyboard for code-page switching.

A list of countries with more than one subcountry code is shown in Table 16.

<i>Table 16. Subcountry Parameter</i>	
Codes	Country
243,245	Czechoslovakia
189,120	France
141,142	Italy
166,168	United Kingdom
<b>Note:</b> If you don't specify the subcountry code for these countries, WorkSpace On-Demand uses the first subcountry code listed.	

For further information on the Keyboard driver, see "Miscellaneous OS/2 Warp Device Drivers" in *The Guide to OS/2 Warp Device Drivers*, SG24-4627, as well as the online help.

### 8.3.6 Serial Port Drivers

Serial ports are used for as an interface to a number of devices. The most common device is usually the printer. All the default Machine Classes provide the standard serial port drivers. The standard classes assume that the hardware is enabled. Most PC hardware allows you to enable or disable serial ports. Before attempting to add or modify support, ensure that the hardware is enabled.

The name of the serial port driver used by WorkSpace On-Demand is COM.SYS. It supports up to four serial ports called COM1, COM2, COM3, and COM4.

The files used by WorkSpace On-Demand to access the serial ports are shown in Table 17.

Table 17. Files Used to Access the Serial Port	
File name	Description
COM.SYS	This is the primary serial communication driver. It should be located in CONFIG.SYS after any other serial communication drivers, such as MOUSE.SYS.
VCOM.SYS	This driver is used for DOS applications. It should be denoted in CONFIG.SYS after COM.SYS.
COMM.DRV	This driver is used for WIN-OS/2 sessions. It is not loaded in the CONFIG.SYS file.

The default Machine Classes have the following statements in the default CONFIG.SYS file:

```
DEVICE=Z:\OS2\BOOT\COM.SYS
DEVICE=Z:\OS2\MDOS\VCOM.SYS
```

Where Z: is defined as the boot drive

For ISA/EISA machines, only COM1 and COM2 are available by default. If you want to use COM3 and COM4 as well, you have to specify additional parameters. The parameters for COM.SYS and their descriptions are shown below:

COM.SYS (<n>,<addr>,<IRQ>,<S>) [(<n>,<addr>,<IRQ>,<S>)] ...

<n>        COM port number (1, 2, 3, or 4)  
<addr>    COM port address in hex number (3e8, 2e8, 3320 or others)  
<IRQ>     IRQ level in decimal number (from 1 to 15)  
<S>       Unexpected interrupt handling value (optional).  
          You can set it to:  
          - D de-install COM.SYS if more than 1000 interrupt occur.  
          - I ignore unexpected interrupts

An example of a definition for COM3 is shown below:

```
DEVICE=C:\OS2\COM.SYS (3,3E8,10,D)
```

This defines COM3 at I/O port address 03E8h / IRQ 5. The driver unloads if more than 1000 interrupts occur.

It is not possible in ISA/EISA computers to share the IRQ level, and therefore all IRQ levels defined for COM ports must be unique.

### 8.3.6.1 Configuring the Serial Port

There are no PM utilities to configure the serial port as there is with OS/2. The command line MODE command sets the mode for serial ports. It is valid in both OS/2 and DOS sessions. Before using the MODE command, be sure that the serial port device driver COM.SYS is installed.

The syntax of the MODE command is shown below:

```
MODE COMx,<baud-rate>,<parity>,<databits>,<stopbits>,<p>
```

x            Port number between 1 and 4  
<baud-rate> Baud rate between 110 and 57600 baud  
<parity>    N=None, O=Odd, E=Even  
<databits>   Databits between 5 and 8  
<stopbits>   Stopbits, values are 1, 1.5, 2.  
<p>         Specifies a timeout of 30 seconds for DOS sessions

The P parameter can only be used in DOS sessions. All the parameters are positional. This means, if you want to omit a parameter but include a parameter that follows it, you have to enter a comma for the parameter that doesn't change.

Example:

```
MODE COM1,,,,P
```

For further information about using the MODE command, please refer to the online help.

### 8.3.6.2 IRQ Settings

If you have a ISA machine, each device or port needs a unique interrupt and I/O port address assignment. Unlike the parallel ports, serial ports can be configured to a different interrupt or I/O address by using additional parameters for the COM.SYS statement in CONFIG.SYS. The Standard I/O port addresses and IRQ assignments are shown in Table 18.

Table 18. Standard I/O Port Addresses and IRQ Assignments			
Bus type	COM port	I/O port	IRQ number
General	COM1	3F8h	IRQ4
General	COM2	2F8h	IRQ3
ISA bus	COM3	3E8h, 3E0h, 338h	IRQ4
ISA bus	COM4	2E8h, 2E0h, 238h	IRQ3
MCA bus	COM3	3220h	IRQ3
MCA bus	COM4	3228h	IRQ3

Standard definitions for COM3 and COM4 have never been defined. Instead, a convention places the port addresses for COM3 and COM4 at 03E8h and 02E8h, respectively. This is a generally accepted convention but not a standard. Users who would like to use these ports have to check the IRQ and I/O port address settings for conflicts. After you have established unique IRQ and port address settings, you can configure COM3 and COM4 with the COM.SYS statement in CONFIG.SYS.

For further information on the Serial Port driver, see "Miscellaneous OS/2 Warp Device Drivers" in *The Guide to OS/2 Warp Device Drivers*, SG24-4627, as well as the online help.

### 8.3.6.3 Setting IRQ Levels on ISA Systems

When setting IRQs, you should generally choose the lines shown as open in Table 19. These are the lines most likely to be available without conflicts with other adapters.

Table 19 (Page 1 of 2). Standard Definitions of IRQs	
IRQ	Standard usage
0	System timer
1	Keyboard
2	Secondary programmable interrupt controller
3	COM2 (serial communication port 2)

<i>Table 19 (Page 2 of 2). Standard Definitions of IRQs</i>	
<b>IRQ</b>	<b>Standard usage</b>
4	COM1 (serial communication port 1)
5	LPT2 (parallel port 2)
6	Floppy disk drive
7	LPT1 (parallel port 1)
8	Real time clock
9	Open
10	Open
11	Open
12	PS/2 mouse (if present)
13	Math coprocessor
14	Hard drive
15	Open

If the system only has one parallel port, IRQ 5 would also be available for other use (for example COM3 or COM4 support).



---

## **Chapter 9. Administering WorkSpace On-Demand in the Enterprise**

This chapter describes the management and setup of WorkSpace On-Demand servers and clients in a large enterprise environment with many locations spread over a large geographic area. While you can manage a large enterprise environment using the GUI interfaces provided with WorkSpace On-Demand, you will probably find that using the command line interface and other available management tools is easier and more productive. This chapter describes some ways in which you can use the command line interface and command scripts to automate some administration tasks.

Large enterprise environments often give rise to unique administration issues. These include such items as software distribution, planning the integration of new clients into existing LANs, problems arising from the different hardware configurations and the system software needed to run applications in this environment. This chapter describes some of those issues and offers some potential solutions.

---

### **9.1 WorkSpace On-Demand in an Enterprise Environment: An Overview**

The typical WorkSpace On-Demand installation in an enterprise environment will have a large number of client workstations and servers that must be managed. These clients and servers may be spread over many locations in a large geographic area. The management of this environment must be carefully planned and structured, so that the setup and configuration of every PC can be managed as easily and effectively as possible. Figure 68 on page 194 shows a typical enterprise environment.

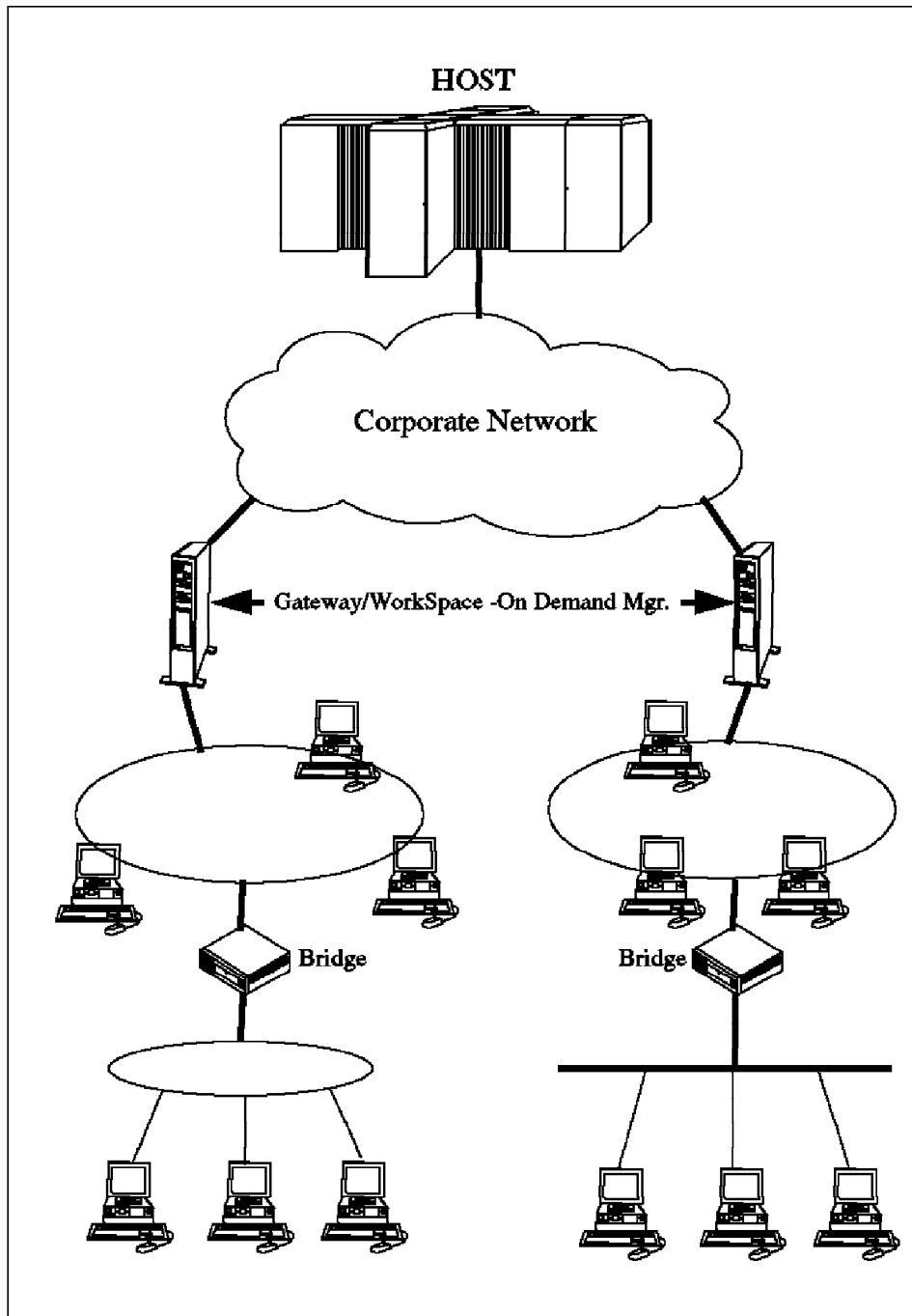


Figure 68. Structure of an Enterprise Environment



An enterprise environment may be complex and can contain many different systems connected by different types of networks. Multiple LANs may be connected to each other and to one or more host systems through bridges or routers, and different segments within the corporate network may differ in their topology, network protocols and bandwidth. Within a single LAN segment, there may be gateways for host communications, servers for LAN-based file and print services, WorkSpace On-Demand management servers, and many clients.

To implement WorkSpace On-Demand in an enterprise environment, you should consider the following items:

1. The hardware configurations to be supported for both the clients and servers
2. The system software ("middleware") products required to run enterprise applications on the WorkSpace On-Demand clients
3. The infrastructure of your enterprise-connected LANs
4. The management of the servers
5. The management of the network topology
6. Migration to WorkSpace On-Demand and coexistence with your existing LAN-based software platform(s)

The remaining sections of this chapter discuss each of these topics in more depth.

---

## 9.2 Hardware Configurations and Resulting Machine Classes

You should give careful consideration to the PC architecture and the different hardware components that you plan to support with WorkSpace On-Demand. The more different machine types you use, the more Machine Classes you will need to define. With every Machine Class you define, you will increase the effort to implement WorkSpace On-Demand since you must test each Machine Class with all the applications you want to run in this environment.

You can reduce the work of implementing, testing and supporting your WorkSpace On-Demand environment by limiting the combinations of critical hardware components, and therefore the number of Machine Classes, to a minimum. See Chapter 8, "WorkSpace On-Demand Machine Classes" on page 149, for details on how to create and implement Machine Classes.

When planning to deploy a large number of WorkSpace On-Demand clients, you should consider the following:

- Try to avoid a wide range of hardware configurations in order to minimize the implementation, testing and administration effort and the associated costs.
- Select a single hardware platform to support. After the hardware decision has been made, buy the number of machines needed for the planned installation cycle and store these machines during the roll-out process. If cost or other considerations prohibit a single purchase, try to buy machines in a small number of large lots rather than buying them in small groups over time. This will reduce the likelihood of small but significant differences in hardware configuration.
- For extremely large projects, deploying WorkSpace On-Demand across the entire enterprise may take several years and may make it impossible to base the entire roll-out on a single hardware platform. In such cases, you should plan the deployment in a series of cycles, year by year, and expect to use a new hardware platform in each cycle, since hardware manufacturers are likely to update their product lines during the period of the roll-out. You will need to implement and test WorkSpace On-Demand on each new hardware configuration and its resulting Machine Class.
- As there will probably be new system software releases during this time, implementing WorkSpace On-Demand will require a combination of hardware and software testing. Not only must you take care of the installation of new machines but also plan for migration of existing machines, already deployed, to the new software release.
- Install a test environment at your central site that reflects your enterprise topology. In this test environment, attempt to have examples of every supported hardware configuration so you can test your current WorkSpace On-Demand configuration on all supported hardware.
- Be aware of the ongoing support for a Machine Class once it has been rolled out into the enterprise. You will need to support the hardware with every new release of the WorkSpace On-Demand product and system software components running under WorkSpace On-Demand.

---

### 9.3 System Software Necessary in the WorkSpace On-Demand Environment

To run your applications in the enterprise environment, it may be necessary to install additional system software ("middleware"). Typical functions that require middleware are:

- Host transaction processing through LU0
- 3270 applications
- APPC communications

- LAN-Server File and Print
- Hardware/software inventory

Table 20 shows the dependencies of functionality and middleware for the list given above:

<i>Table 20. System Software Necessary to Provide Special Functions</i>	
<b>Application Function</b>	<b>Middleware Required</b>
HOST transaction processing through LU0	IBM OS/2 Access Feature
3270 applications	IBM Personal Communication/3270 for OS/2
APPC communications	IBM OS/2 Access Feature
LAN-Server File and Print	IBM OS/2 Warp Server
Hardware/Software inventory	IBM NetFinity Server/Client for OS/2

In a traditional client-server environment, the necessary software for an application function must be installed on every client workstation in the network. In a large enterprise, this is usually done using software distribution tools such as IBM NetView Distribution Manager/MVS and NetView DM/2. Figure 69 on page 198, shows the application structure for today's client/server environment.

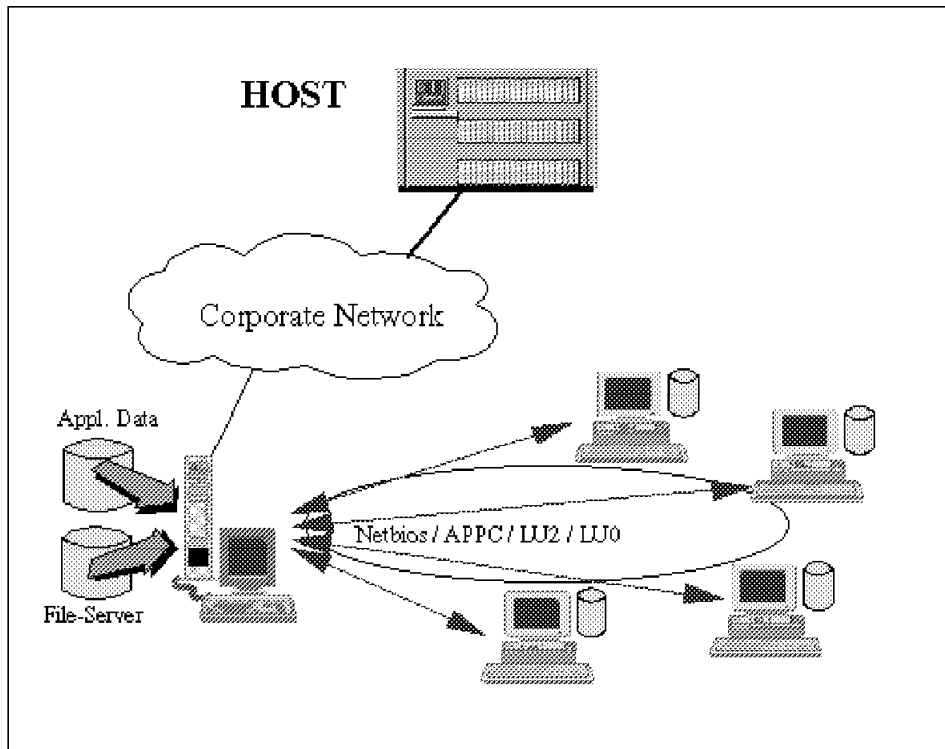


Figure 69. Application Structure. A Typical Client/Server Environment without Workspace On-Demand.

Obviously, every PC in such an environment must be installed and configured individually to avoid address and naming conflicts. This configuration is typically done by a response file-driven installation of the middleware on each PC in the enterprise.

Now, what is the difference when we run an application in a Workspace On-Demand environment? The application architecture does not change; so the communications, transactions, LAN access and 3270 emulation must still work on the Workspace On-Demand client. However, the client now obtains all of its software from the server including operating system, communications and other subsystems. No software is installed on the client; so all configuration must be carried out on the server. Figure 70 on page 199 shows the system architecture of a typical Workspace On-Demand environment.

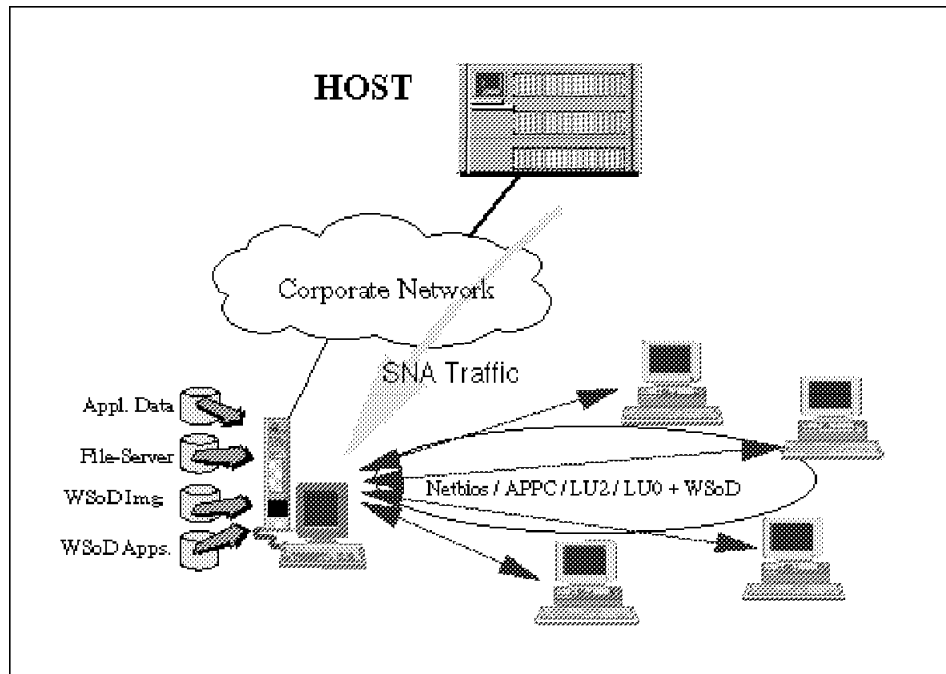


Figure 70. System Structure for an Application in a WorkSpace On-Demand Environment

With the software being completely stored on the WorkSpace On-Demand server, all clients get their operating system and applications from the same software image. In order to allow unique configuration information for each client, configuration files are stored in the client's working directory under the IBMLANRPL tree on the server.

The following sections of this chapter provide a more detailed look at the configuration and installation of specific middleware products on the WorkSpace On-Demand server for the functions listed in Table 20 on page 197.

### 9.3.1 IBM OS/2 Access Feature in the WorkSpace On-Demand Environment

IBM's OS/2 Access Feature is necessary for base SNA communication functions such as LUA sessions and APPC communications. In order to make these functions available to WorkSpace On-Demand clients, you can follow the steps of this section to install the product on the WorkSpace On-Demand server and create configurations for client workstations.

1. Install OS/2 Access Feature on an OS/2 Warp 4 client, which we will call the *reference client*, along with the complete set of required functions for your environment.

2. Create a directory, CMLIB, under the IBMLANRPLBB10.US directory.
3. Create a directory, EPW, under the IBMLANRPLBB10.US directory to provide FFST/2 support. Create subdirectories under EPW for HELP, DLL and SYSTEM.
4. Copy all the files from the reference client's program directory, CMLIB, and its subdirectory structure to the IBMLANRPLBB10.USCMLIB directory on the WorkSpace On-Demand server.
5. Copy selected files for the FFST/2 from the client's OS/2 subdirectory to the IBMLANRPLBB10.USEPW directory on the WorkSpace On-Demand server. Figure 71 shows the files that should be copied to the IBMLANRPLBB10.USEPW directory and to the subdirectories HELP, DLL and SYSTEM.

Directory of \ibmlan\rp1\bb10.us\epw

EPW.EXE  
EPW.INI  
EPW.MSG  
EPWCONS.EXE  
EPWDD.SYS  
EPWDDR3.EXE  
EPWDF.EXE  
EPWDFOLD.EXE  
EPWFOLD.ICO  
EPWH.MSG  
EPWICON.EXE  
EPWMP.EXE  
EPWMUX.EXE  
EPWPCT.EXE  
EPWPSI.EXE  
EPWRCV.EXE  
EPWROUT.EXE  
LOG.SYS

*Figure 71 (Part 1 of 2). FFST/2 Files Necessary for OS/2 Access Feature*

```
Directory of \ibm\lan\rpl\bb10.us\epw\DLL
EPWCUA.DLL
EPWELG01.DLL
EPWINIT.DLL
EPWNL001.DLL
EPWPRO.DLL
EPWPROB.DLL
EPWPROB2.DLL
EPWPSI16.DLL
EPWPSI32.DLL
EPWSVC16.DLL
EPWSVC32.DLL
Directory of \ibm\lan\rpl\bb10.us\epw\HELP
EPWNL001.HLP
Directory of \ibm\lan\rpl\bb10.us\epw\SYSTEM
LOGDAEM.EXE
```

Figure 71 (Part 2 of 2). FFST/2 Files Necessary for OS/2 Access Feature

6. Create a directory, CMLIB, under the RPLRPLUSER<Client\_Name> directory.
7. Create a directory, EPW, under the RPLUSER<Client\_Name> directory (necessary for FFST/2 logging support).
8. Create an individual OS/2 Access Feature configuration for the client and save it to the directory RPLRPLUSER<Client\_Name>\CMLIB. The name of the configuration file must match the name for the default configuration in the client's CMLIB directory (for instance, WSODOAF.CFG)

**Note**

The creation of the client's OS/2 Access Feature configuration can be done with response files. Using response files allows you to create multiple, different client configurations using one generation process on the WorkSpace On-Demand server. Figure 121 on page 298 in Appendix G, "Samples for Managing WorkSpace On-Demand in the Enterprise" on page 297, shows an example of an OS/2 Access Feature response file used to create a configuration for a WorkSpace On-Demand client.

9. Change the CONFIG.SYS file for the WorkSpace On-Demand client in \IBMLAN\RPL\MACHINES\<client\_name> to support the OS/2 Access Feature. The changes are shown in bold in Figure 72 on page 202.

```
LIBPATH=.;Z:\;Z:\OS2\DLL;Z:\java11\dlI;Z:\MPTN\DLL;Z:\IBMCOM\DLL;
Z:\IBMI18N\DLL;Z:\IBMLAN\NETLIB;Z:\MUGLIB\DLL;Z:\netscape;
Z:\javaos2\dlI;Z:\tcpip\dlI;Z:\tcpip\pcomos2;Z:\os2\mdos;
Z:\mmos2\dlI;Z:\CMLIB\DLL;Z:\EPW\DLL;
SET
PATH=Z:\OS2;Z:\java11\bin;Z:\OS2\SYSTEM;Z:\MPTN\BIN;Z:\IBMCOM;
Z:\IBMLAN\NETPROG;Z:\MUGLIB;Z:\OS2\INSTALL;Z:\;Z:\netscape;
Z:\javaos2\bin;Z:\tcpip\bin;Z:\tcpip\pcomos2;Z:\os2\mdos;
Z:\mmos2;Z:\os2\mdos\winos2;z:\CMLIB;
SET
DPATH=Z:\OS2;Z:\OS2\SYSTEM;Z:\MPTN;Z:\IBMCOM;Z:\IBMLAN;
Z:\IBMLAN\NETPROG;Z:\OS2\INSTALL;Z:\;Z:\tcpip\pcomos2;
Z:\os2\mdos;Z:\mmos2;Z:\mmos2\install;Z:\os2\mdos\winos2;
Z:\CMLIB;Z:\CMLIB\EN_US;

SET BOOKSHELF=Z:\OS2\BOOK;Z:\ibmlan\netprog;Z:\tcpip\help;
Z:\CMLIB\US_US\BOOK;

REM OS/2 Access Feature entries

RUN=z:\epw\SYSTEM\LOGDAEM.EXE DEVICE=z:\epw\LOG.SYS
RUN=z:\epw\EPWDDR3.EXE DEVICE=z:\epw\EPWDD.SYS
DEVICE=z:\cmlib\ACSLANDD.SYS
DEVICE=z:\cmlib\CMKFMDE.SYS
DEVICE=z:\cmlib\ACSHPRDD.SYS
RUN=z:\epw\EPWROUT.EXE 1
RUN=z:\epw\EPW.EXE
SET CPMATH=z:\cmlib
```

Figure 72. CONFIG.SYS Changes for OS/2 Access Feature

10. Edit the client's FIT file IBMLANRPLFITScient\_name.FIT to support OS/2 Access Feature and FFST/2, as shown in Figure 73 on page 203.



```

;OS/2 Access Feature support
z:\cmlib                                BB10.US\CMLIB
z:\cmlib\cm.ini                         \\WSODSRV\WRKFILES\ISACD\cmlib\cm.ini
z:\cmlib\*.cfg                         \\WSODSRV\WRKFILES\ISACD\cmlib
z:\cmlib\*.ndf                         \\WSODSRV\WRKFILES\ISACD\cmlib
z:\cmlib\*.sec                         \\WSODSRV\WRKFILES\ISACD\cmlib
z:\cmlib\*.rsp                         \\WSODSRV\WRKFILES\ISACD\cmlib
z:\cmlib\*.log                         \\WSODSRV\WRKFILES\ISACD\cmlib
z:\cmlib\*.cf2                         \\WSODSRV\WRKFILES\ISACD\cmlib

;FFST\2 Support
z:\EPW                                BB10.US\EPW
z:\epw\*.ini                         \\WSODSRV\WRKFILES\ISACD\epw
z:\os2\system\epw\*.dat               \\WSODSRV\WRKFILES\ISACD\epw
z:\os2\system\*.dmp                   \\WSODSRV\WRKFILES\ISACD\epw
z:\os2\system\*.dat                   \\WSODSRV\WRKFILES\ISACD\epw

```

Figure 73. Sample for Additions in Client's FIT File for OS/2 Access Feature Support

In an enterprise environment, it is likely that you will want to provide access to the OS/2 Access Feature for all the clients in a particular LAN. To avoid editing every client's control files, you can apply the necessary changes to the files listed below:

- Apply the changes shown in Figure 72 on page 202 to the CONFIG.SYS file in the IBMLANRPLMACHINESBB10.US<machine\_class> directory.
- Make the FIT file changes shown in Figure 73 to the default user FIT file DFBB10US.FIT.
- Use the CMSETUP program with the /R parameter to create the necessary response file to drive the configuration process.
- Write additional REXX batches to create the necessary directories, CMLIB and EPW, in the RPLRPLUSER<client\_name> directory and to copy the OS/2 Access Feature configuration files, CM.INI, WSODOAF.CFG, WSODOAF.NDF, WSODOAF.SEC, WSODOAF.RSP, and WSODOAF.CF2, to this directory.

### 9.3.2 IBM Personal Communication/3270 for OS/2 in the WorkSpace On-Demand Environment

IBM Personal Communication/3270 for OS/2 is used to gain access to host-based 3270 applications. The PCOMM Lite version of this product is included in the WorkSpace On-Demand product, but lacks certain functions present in the full version. This section covers setting up the full version of Personal Communication/3270 for OS/2 in a WorkSpace On-Demand environment and shows the interoperability with the OS/2 Access Feature

described in 9.3.1, “IBM OS/2 Access Feature in the WorkSpace On-Demand Environment” on page 199. Figure 122 on page 301 in Appendix G, “Samples for Managing WorkSpace On-Demand in the Enterprise” on page 297, shows the Personal Communication/3270 for OS/2 configuration file used to run Personal Communication/3270 for OS/2 over an OS/2 Access Feature configuration.

The steps required to run Personal Communication/3270 for OS/2 in a WorkSpace On-Demand environment are:

1. Install Personal Communication/3270 for OS/2 on your server. You can do this by choosing the **Installation on Server** radio button in the Personal Communication/3270 for OS/2 installation panel. Alternatively, you can install the product on an OS/2 Warp 4 reference client, then copy the files to the WorkSpace On-Demand server as described in the following steps.
2. Create a directory, PCOMOS2, under the IBMLANRPLBB10.US directory.
3. Create a directory, PCOMOS2, under the IBMLANRPLUSER<Client\_Name> directory for the configuration files.
4. Copy the files from the reference client’s C:PCOMOS2 directory, including the subdirectories, to the directory IBMLANRPLBB10.USPCOMOS2 on the WorkSpace On-Demand server.
5. Create a configuration file for PCOMM/3270 for every WorkSpace On-Demand client and copy this file to the IBMLANRPLUSER<Client\_Name>PCOMOS2 directory.

**Note**

If your environment allows, you can create a unique configuration file, PCOS2.WS, that is identical for every client. In this case, you will only need to copy this file to each client’s working directory IBMLANRPLUSER<Client\_Name>PCOMOS2.

6. Create an icon on for Personal Communication/3270 for OS/2 on the WorkSpace On-Demand client’s desktop. You can do this in one of several different ways:
  - You can use a REXX script like that shown in Figure 74 on page 205 that makes a SysCreateObject call for the PCSWS.EXE program, and run this script once on every machine.

```

/* REXX Program to create PCOMM OS/2 Icon on Desktop */

call RxFuncAdd 'SysLoadFuncs', 'RexxUtil', 'SysLoadFuncs'
call SysLoadFuncs

exe = "EXENAME=z:\PCOMOS2\PCSW.S.EXE "
parm = "Parameters=z:\PCOMOS2\PC3270.WS"
type = "ProgType=PM"
applstring = exe || ';' || parm || ';' || type || ' '
SysCreateObject("WPPProgram", "PCOMM_OS/2", "<WP_DESKTOP>", ,
               applstring, reply);

```

Figure 74. Sample REXX Script to Create the PCOMM/3270 ICON on a Client's Desktop

- You can use a .RC file as shown in Figure 75 to modify the default OS2.INI file that is used for every WorkSpace On-Demand client, using the MAKEINI utility. You can enter MAKEINI from the command line. Change to the OS/2 directory of the Machine Class:

```
CD IBMLANRPLMACHINESBB10.US<machine_class>OS2
```

You only need to use the command MAKEINI OS2.INI PCOS2.RC to store the information in the OS2.INI for the Machine Class. Every new WorkSpace On-Demand client based on this Machine Class will then get the icon on its desktop.

```

/* File used by MAKEINI.EXE to create OS2.INI */

STRINGTABLE BEGIN
"""
/* Register Thin Object classes */
"PM_InstallClass" "ThinObject" "tobject"
"PM_InstallClass" "ThinDesktop" "tdesk"
"PM_InstallClass" "ThinProgram" "tpgm"
"PM_InstallObject" "PCOS2;WPPProgram;<WP_DESKTOP>;UPDATE"
                "EXENAME=z:\PCOMOS2\PCSW.S.EXE;
                PARAMETERS=z:\PCOMOS2\PC3270.WS;
                ICONPOS=10 80;
                OBJECTID=<PCOM_OS2>
END
/* The parameters for the PM_InstallObject line normally are in one line */
/* We spread the parameters over several lines to make it better readable. */

```

Figure 75. Sample RC File to Modify the Model OS2.INI

7. Edit the client's FIT file so that it points to the correct PCOMOS2 directories as shown in Figure 76 on page 206.

```
;Personal Communication/3270 for OS/2/3270 Support
Z:\PCOMOS2          BB10.US\PCOMOS2
Z:\PCOMOS2\*.WS     \\WSODSRV\WRKFILES\ISACD\PCOMOS2
```

Figure 76. FIT File Extensions for Personal Communication/3270 for OS/2/3270 Support

8. Edit the CONFIG.SYS file for the WorkSpace On-Demand client. Add the Z:\PCOMOS2 directory to the LIBPATH, PATH and DPATH statements.

```
LIBPATH=.;Z:\;Z:\OS2\DLL;Z:\java11\dll;Z:\MPTN\DLL;Z:\IBMCOM\DLL
;Z:\IBMI18N\DLL;Z:\IBMLAN\NETLIB;Z:\MUGLIB\DLL;Z:\netscape;
Z:\javaos2\dll;Z:\tcpip\dll;Z:\os2\mdos;Z:\mmos2\dll;Z:\PCOMOS2
SET
PATH=Z:\OS2;Z:\java11\bin;Z:\OS2\SYSTEM;Z:\MPTN\BIN;
Z:\IBMCOM;Z:\IBMLAN\NETPROG;Z:\MUGLIB;Z:\OS2\INSTALL;
Z:\;Z:\netscape;Z:\javaos2\bin;Z:\tcpip\bin;Z:\os2\mdos;
Z:\mmos2;Z:\os2\mdos\winos2;Z:\PCOMOS2
SET
DPATH=Z:\OS2;Z:\OS2\SYSTEM;Z:\MPTN;Z:\IBMCOM;Z:\IBMLAN;
Z:\IBMLAN\NETPROG;Z:\OS2\INSTALL;Z:\;Z:\os2\mdos;Z:\mmos2;
Z:\mmos2\install;Z:\os2\mdos\winos2;Z:\PCOMOS2
```

Figure 77. CONFIG.SYS Changes for Personal Communication/3270 for OS/2/3270 Support

After you complete these steps, you should be able to access Personal Communication/3270 for OS/2 from your WorkSpace On-Demand clients.

### 9.3.3 IBM TME 10 NetFinity in the WorkSpace On-Demand Environment

IBM's TME 10 NetFinity Server and Client are used to manage hardware and software inventory, measure performance and manage critical resources. In our example, we set up TME 10 NetFinity to obtain information about the hardware configuration of a WorkSpace On-Demand client. To make the NetFinity Client software usable by all WorkSpace On-Demand clients, we followed the following steps:

1. Install TME 10 NetFinity on the server using the TME 10 NetFinity Server product shipped with OS/2 Warp Server. In the installation panel, select the **TME 10 NetFinity Server - Client Installation** from the list of installable products. Install the product to the IBMLANRPLBB10.USNETFIN.40 directory to be sure that every WorkSpace On-Demand client can access the software.

#### Note

In an enterprise environment, the software will typically be installed unattended by using response files.

2. Create an icon on the WorkSpace On-Demand client's desktop. As shown in 9.3.2, "IBM Personal Communication/3270 for OS/2 in the WorkSpace On-Demand Environment" on page 203, you can do this in two different ways:
  - Use a REXX script like that shown in Figure 78, which makes a SysCreateObject call for the SFING30.EXE program, and run this script once on every machine

```
/* REXX Program to create TME 10 NetFinity System Information
   Tool Icon on Desktop */

call RxFuncAdd 'SysLoadFuncs', 'RexxUtil', 'SysLoadFuncs'
call SysLoadFuncs
applname="TME10 NetFinity System Information Tool"
exe = "EXENAME=z:\NetFin.40\BIN\SINFG30.EXE "
parm = ""
type = "ProgType=PM"
applstring = exe || ';' || parm || ';' || type || ' '
SysCreateObject("WPProgram", applname, "< WP_DESKTOP>", ,
               applstring, reply);
```

Figure 78. Sample REXX Script to Create a TME 10 NetFinity Icon on a Client's Desktop

- You can use a .RC file as shown in Figure 79 on page 208 to modify the default OS2.INI file used for every WorkSpace On-Demand client, using the MAKEINI utility. You can enter MAKEINI from the command line. Change to the OS/2 directory of the machine class:

```
CD IBMLANRPLMACHINESBB10.US<machine_class>OS2
```

You only need to use the command MAKEINI OS2.INI NETFIN.RC to store the information in the OS2.INI for the Machine Class. Every new WorkSpace On-Demand client based on this Machine Class will have the TME 10 NetFinity System Information Tool icon on its desktop.

```

/* File used by MAKEINI.EXE to create OS2.INI */

STRINGTABLE BEGIN
    ""
    /* Register Thin Object classes */
    "PM_InstallClass" "ThinObject" "tobject"
    "PM_InstallClass" "ThinDesktop" "tdesk"
    "PM_InstallClass" "ThinProgram" "tpgm"
    "PM_InstallObject" "TME10 NetFinity System Information Tool;
        WPPProgram;<WP_DESKTOP>;UPDATE"
        "EXENAME=z:\NetFin.40\BIN\SINFG30.EXE;
        ICONPOS=40 80 ;
        OBJECTID=<NetFinity>"
END

/* The parameters for the PM_InstallObject line normally are in one line */
/* We spread the parameters over several lines to make it better readable. */

```

Figure 79. Sample RC File to Modify the Model OS2.INI

When you complete these steps, every WorkSpace On-Demand client will have access to the TME 10 NetFinity System Information Tool and can gather hardware information for that client. There are more functions available with TME 10 NetFinity, but the goal here is to provide some basic guidelines on how to implement the product in a WorkSpace On-Demand environment. You can take the example shown above and extrapolate these guidelines to other components of TME 10 NetFinity.

### 9.3.4 General Guidelines for Implementing Middleware

In the previous sections, we have discussed specific guidelines on implementing middleware in a WorkSpace On-Demand environment. This section covers a general approach to enabling an enterprise application in a WorkSpace On-Demand environment. The steps may vary for different middleware, but in general, this approach can be followed for most middleware products:

1. Install the necessary middleware on an OS/2 Warp 4 reference client. Try to install all the files necessary for the different configurations of this software (for example, OS/2 Access Feature normally installs only the files necessary to support a specified configuration).
2. Test the functions necessary in your environment to make sure that everything is properly set up.
3. Check whether the component has dependent subsystems that are installed in a separate directory (for example, FFST/2).

4. Create a product directory under IBMLANRPLBB10.US on the WorkSpace On-Demand server.
5. Create a product directory in the working directory of the client IBMLANRPLUSER<client\_name> for the configuration files to which the client must have write access.
6. Copy the product files from the reference client's product directory and its subdirectories to the product directory IBMLANRPLBB10.US<product> on the WorkSpace On-Demand server.
7. Create client-specific configuration files if necessary, and copy these configuration files to the client's working directory IBMLANRPLUSER<client\_name><product>.
8. Edit the CONFIG.SYS files of the Machine Classes that are to run the software (for example, IBMLANRPLMACHINESBB10.USISAVGA.MCCONFIG.SYS) to add the required statements.
9. Add the necessary lines to the default FIT file DFBB10US.FIT to support the software on every client.
10. If the software is to be started from an icon on the desktop, create the icon by modifying the OS2.INI in the IBMLANRPLMACHINESBB10.US<machine\_class>OS2 directory. This is necessary for every Machine Class that runs this software. As described in 9.3.2, "IBM Personal Communication/3270 for OS/2 in the WorkSpace On-Demand Environment" on page 203, you can do this by using the MAKEINI utility with a product-specific .RC file as shown in Figure 80 on page 210.

```

/* File used by MAKEINI.EXE to create OS2.INI */

STRINGTABLE BEGIN
    ""
    /* Register Thin Object classes */
    "PM_InstallClass" "ThinObject" "tobject"
    "PM_InstallClass" "ThinDesktop" "tdesk"
    "PM_InstallClass" "ThinProgram" "tpgm"
    "PM_InstallObject" "Sample Program
        ;WPPProgram;<WP_DESKTOP>;UPDATE"
        "EXENAME=Z:\SAMPLE\NEWPM.EXE;
        PARAMETERS=Z:\SAMPLE\SAMPLE.CFG;
        ICONPOS=10 80;
        OBJECTID=<SAMPLE_PGM>"

END

/* The parameters for the PM_InstallObject line normally are in one line */
/* We spread the parameters over several lines to make it better readable. */

```

Figure 80. Sample RC File to Modify an OS2.INI File

Complete these steps once to create a model WorkSpace On-Demand server configuration. Zip the files and directories created/modified to a single file to make the configuration reproducible. You can then transfer this ZIP file to the desired servers and install the configuration using a REXX script after the WorkSpace On-Demand product installation is finished.

To support many clients, create the client-specific configuration files using response files. In some cases, it might be necessary to write additional programs if the default configuration program for the software does not allow you to create configuration files for other machines. You can generate the response files at your central site if you have a database that contains an entry for every PC in your network to generate the client-specific entries in this response file.

If you do not have such a database, you may have an administrator at each location where you plan to use this environment. This administrator can create the configuration files for the clients and copy them to the corresponding directories after the clients are defined to the WorkSpace On-Demand server.

These guidelines will give you a high degree of flexibility to support various middleware and will decrease the administration effort you need to spend on configuring and managing the configurations for every machine in your WorkSpace On-Demand environment.



## 9.4 Infrastructure Considerations

When planning to run WorkSpace On-Demand in an enterprise environment, you should have a detailed understanding of the network and hardware infrastructure in your enterprise. The existing infrastructure may significantly affect the implementation of WorkSpace On-Demand in your environment.

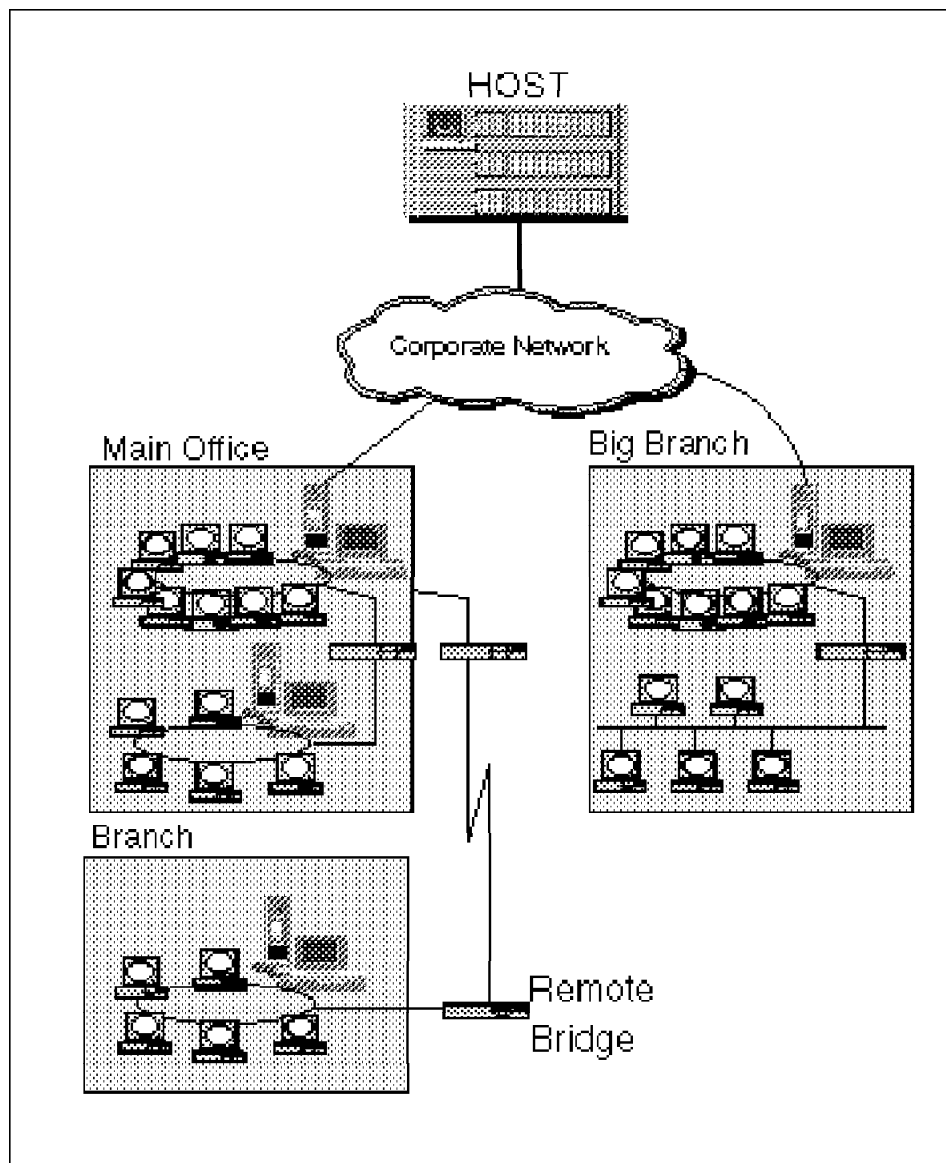


Figure 81. Infrastructure of Enterprise Networks

The typical enterprise environment will have a heterogeneous structure comprised of a central site and many other locations spread over a large geographic area, as shown in Figure 81. These locations can be structured into main offices and branches connected to the main office by remote bridges or routers. In large locations, there may be more than one physical LAN, more than one physical network technology, different types of LAN connection boxes such as bridges and routers, and many clients connected to the network.

With such a heterogeneous infrastructure, there are several implementation considerations you may want to look at before implementing WorkSpace On-Demand. This section provides you with several infrastructure topics to consider to prepare you for the use of WorkSpace On-Demand in your enterprise environment:

1. How many clients are to run WorkSpace On-Demand at each location?

This is a very important question, because the number of connected clients to one WorkSpace On-Demand server is limited by a number of hardware and software considerations such as

- What kind of hard drives are used in the server (RAID or SCSI)?
- How many network adapters does the server have?
- How much RAM does the server have?
- Which version of OS/2 Warp Server is used (Entry, Advanced or SMP)?
- What other server services are running on this server (DB2, CM/2, ...)?

The combination of these parameters determines how many clients can practically be connected to one server. This limit is most often defined in terms of acceptable performance for the end-user, rather than a physical limit imposed by the network or system architectures. Since this evaluation is subject to each enterprise's interpretation of acceptable response time, expense and throughput, you will need to evaluate how the combinations of the parameters listed above work best in your WorkSpace On-Demand environment. See Appendix A, "Tuning WorkSpace On-Demand" on page 253, for general performance information.

Detailed information regarding performance and number of connected clients can be found on the Web at <http://www.software.ibm.com/workspace>. On this Web site, you will find a reference to the performance measurements that were made in the IBM performance labs.

Depending on the number of clients you want to connect to every WorkSpace On-Demand server, it might be necessary to have more than one WorkSpace On-Demand server in some locations. For these locations, you have to consider which clients should boot from which server.

If you have more than one server, it makes sense to define backups in case a server fails. These backup definition can be made by defining a WorkSpace On-Demand client to multiple servers. If a client definition is enabled on both servers, the clients will boot from whichever server replies first to its RPL FIND request. See 4.8, “Capacity Planning and High-Availability for WorkSpace On-Demand servers” on page 68, for a more detailed description to this topic.

2. How many LAN segments are connected to one logical LAN?

This question is of importance if you have a logical LAN consisting of many separate physical LAN segments. Figure 82 shows an example of a logical LAN with many LAN segments connected by bridges.

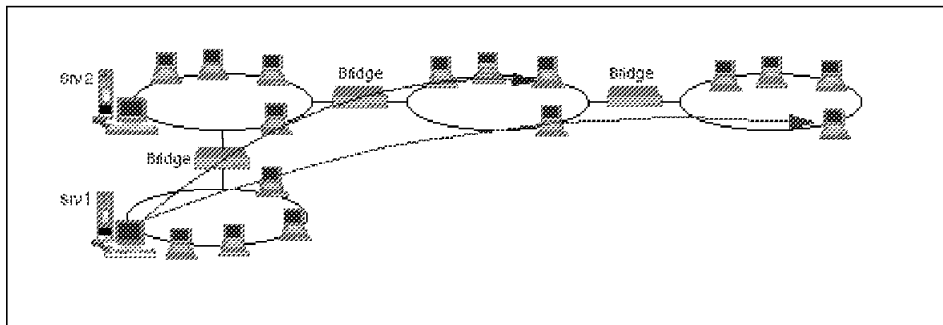


Figure 82. LAN Segments Connected by Bridges

Note that the WorkSpace On-Demand servers for this network are located on one segment, while the clients reside on other segments. Such a network can cause problems when there is heavy network traffic and the bridges become congested, since a congested bridge may discard a frame instead of retransmitting it, with unpredictable results for the boot sequence of WorkSpace On-Demand.

Wherever possible, avoid having bridges between your WorkSpace On-Demand server and its clients. If this is impossible, try to minimize the number of bridges (hop count) between the server and the client. For example, you can reduce the hop count by designing the network structure as shown in Figure 83 on page 214.

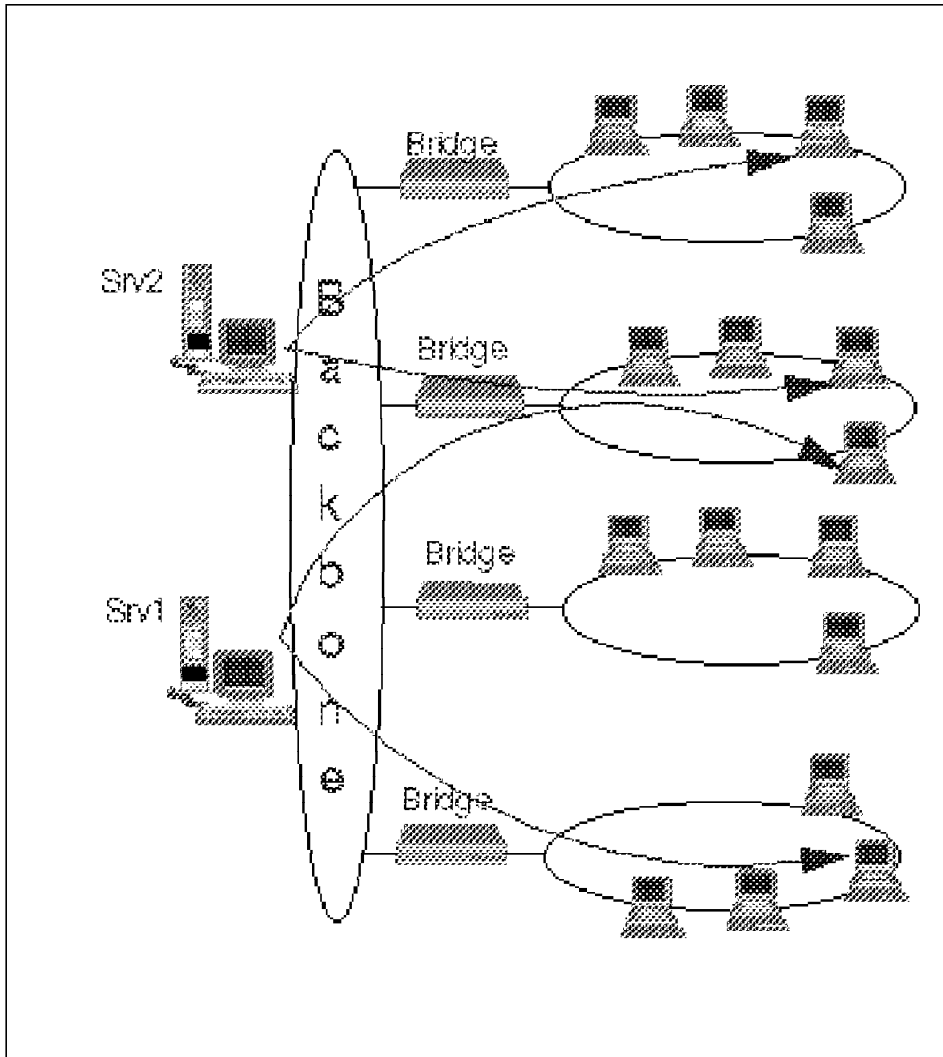


Figure 83. LAN Segments Connected by a Backbone

In the network shown above, the maximum hop count is two, with a maximum of one bridge between the client and its WorkSpace On-Demand server. This reduces the likelihood of lost frames and failed boot attempts.

### 3. Are the LAN segments connected by routers?

WorkSpace On-Demand uses the NetBIOS protocol to boot its clients. This protocol is not routable. If you have LAN segments connected by routers, you must have a WorkSpace On-Demand server in every LAN segment, regardless of the number of connected clients.

4. Do the network topologies differ between LAN segments?

If you have different physical LAN segments (such as token-ring and Ethernet) connected, there must be a WorkSpace On-Demand server in each LAN segment because the server must have the same type of network adapter as the client.

5. What is the bandwidth between the different LAN segments?

If you have LAN segments in separate locations connected by remote bridges over a leased line with a low bandwidth, you should place a WorkSpace On-Demand server in every location since the bridge/line capacity is unlikely to provide acceptable performance. If you have a line-speed comparable to a LAN transmission rate, you may try to connect a WorkSpace On-Demand client to a server across the remote bridge, but it is still advisable to place a server in each location. You can manage the different servers from a central site using the WorkSpace On-Demand administration tool.

6. What kind of client hardware is already installed in your enterprise?

If you want to migrate existing installations to WorkSpace On-Demand, you should create a hardware inventory of the existing client hardware. At the very least, this inventory should contain the different hardware configurations to examine which Machine Classes you will need to create and support.

7. What is the existing and expected dataflow in your environment?

The existing dataflow on your network will increase with the use of WorkSpace On-Demand. If you already have a high network load, you may need to consider the implications of the additional network traffic that WorkSpace On-Demand will create, particularly when booting client workstations.

During the boot process of a WorkSpace On-Demand client, at least 8 MB of data is transferred across the network. If there is need for additional middleware to be loaded during client startup, there will be an additional transfer requirement for each client.

If much of your existing network traffic is based on applications running on the machines that are to be migrated to WorkSpace On-Demand, this may not be a problem. The network load imposed by WorkSpace On-Demand will decrease significantly after the boot process of the WorkSpace On-Demand clients is completed, and the application traffic can continue more or less as usual. However, if you plan to bring new clients with new WorkSpace On-Demand applications into an existing network with a heavy traffic load, it may be necessary to restructure your network to avoid congestion and bad performance.

#### 8. What communication protocols are supported?

WorkSpace On-Demand requires the NetBIOS protocol to be loaded to the client's memory. If you are not already using NetBIOS, this will add an additional protocol in your network. To minimize NetBIOS traffic, you can configure your WorkSpace On-Demand clients to use NetBIOS only to load WorkSpace On-Demand and to use your existing protocols for application traffic.

If you set filters to avoid broadcast transmissions across bridges, you will need to disable these filters if you want to locate clients on a different segment from their WorkSpace On-Demand servers. The RPL FIND requests used by WorkSpace On-Demand will not work across bridges with enabled broadcast filters. If you want to leave the filters active in your bridges, you must connect a WorkSpace On-Demand server to every LAN segment.

The items mentioned in this section are general examples of the factors you may need to consider when implementing WorkSpace On-Demand in your enterprise. Since every enterprise environment is different, there may be topics that have not been included in this list that you will need to consider as well.

---

### 9.5 The Management of WorkSpace On-Demand Management Servers

In an enterprise network, it is likely that there will be many locations where WorkSpace On-Demand must be installed and where WorkSpace On-Demand servers must be managed. One of the most important questions may be the automation of server management tasks such as:

- Software installation on the server
- Adding new clients to the server
- Deleting clients from the server
- Modifying clients on the server
- Replication of Machine Classes
- Modification of Machine Classes
- Defining public applications
- Assigning public applications to users

Automating these functions will help to minimize the effort (and cost) of administering the WorkSpace On-Demand environment and may also result in a more consistent and stable implementation. By using automated processes:

- The possibility for user errors is decreased to a minimum.
- The administration of the WorkSpace On-Demand environment becomes easier.

- It is possible to implement a centralized WorkSpace On-Demand administration.

### 9.5.1 Unattended Installation of WorkSpace On-Demand

WorkSpace On-Demand is fully CID-enabled, and is therefore prepared for software distribution in a CID environment, as is typically used in a large enterprise. This means that you can install the WorkSpace On-Demand product in an unattended mode, using response files to configure the system.

The parameters reflecting your individual environment are set in two response files. The first of these is the WSODCID.RSP file, which is used to configure WorkSpace On-Demand with respect to the target directory, upgrading the command line interface, the Machine Classes to be installed, and the supported client software such as TCP/IP, Netscape Navigator or PCOMM. Figure 84 shows the WSODCID.RSP used in our test environment. You can modify this response file to fit your environment.

```
*****
*                               WORKSPACE ON-DEMAND MANAGER INSTALL                               *
*                               S E C T I O N                                                    *
* *****                                                                *

* ----- *
* Install Target LAN Drive of Server *
* *
*   ie. drive for target \ibmlan *
* ----- *
NCInstallAll.LANDrive=E:

* ----- *
* Command Line Interface and API Support Upgrade *
* *
*   Valid Parms: *
*     0=None (default) *
*     1=Command Line Interface and API Support Upgrade *
* ----- *
NCU_BASE_Manager.Selection=1
```

Figure 84 (Part 1 of 7). WorkSpace On-Demand WSODCID.RSP Response File

```

* ----- *
* Command Line Interface and API Support Upgrade for Merlin *
* *
* Valid Parms: *
* 0=None (default) *
* 1=Command Line Interface and API Support Upgrade for Merlin *
* ----- *
NCU_BASE_Merlin.Selection=0
* ----- *
* Command Line Interface and API Support Upgrade for Server *
* *
* Valid Parms: *
* 0=None (default) *
* 1=Command Line Interface and API Support Upgrade for Server *
* ----- *
NCU_BASE_Server.Selection=1
* ----- *
* Command Line Interface and API Support Upgrade for Server RIPL *
* directory structure *
* *
* Valid Parms: *
* 0=None (default) *
* 1=Command Line Interface and API Support Upgrade for Server *
* RIPL directory structure *
* ----- *
NCU_BASE_RIPL_SERVER.Selection=1

```

Figure 84 (Part 2 of 7). WorkSpace On-Demand WSODCID.RSP Response File



```

*-----*
* The following are the only 4 available combinations for Command Line *
* Interface and API Upgrade Selections: *
* *
* NCU_BASE_Manager.Selection=0 *
* NCU_BASE_Merlin.Selection=0 / ---- Merlin or WARP Server *
* NCU_BASE_Server.Selection=0 \ ---- Do not upgrade CLI and API *
* NCU_BASE_RIPL_SERVER.Selection=0 *
* *
* NCU_BASE_Manager.Selection=1 *
* NCU_BASE_Merlin.Selection=1 / ---- Merlin: *
* NCU_BASE_Server.Selection=0 \ ---- Upgrade CLI and API *
* NCU_BASE_RIPL_SERVER.Selection=0 *
* *
* *
* NCU_BASE_Manager.Selection=1 RIPL or NO RIPL Server: *
* NCU_BASE_Merlin.Selection=0 Upd. CLI and API on srv's *
* NCU_BASE_Server.Selection=1 / ---- dir. struct. only. No upd. *
* NCU_BASE_RIPL_SERVER.Selection=0 \ ---- CLI and API RIPL directory *
* if server is a RIPL server *
* *
* NCU_BASE_Manager.Selection=1 RIPL Server only: *
* NCU_BASE_Merlin.Selection=0 / ---- Upd. CLI and API on srv's *
* NCU_BASE_Server.Selection=1 \ ---- IBMLAN and RIPL directory *
* NCU_BASE_RIPL_SERVER.Selection=1 structure *
*-----*

*-----*
* RIPL Services Upgrade *
* *
* Valid Parms: *
* 0=None *
* 1=RIPL Services Upgrade *
*-----*
NCU_RIPL.Selection=1

* ***** *
* WORKSPACE ON-DEMAND CLIENT INSTALL *
* S E C T I O N *
* ***** *

*-----*
* Install Target Drive *
* *
* Drive \ibmlan\rp1 support *
* ie. c: *
*-----*
NCInstallAll.TargetDrive=e:

```

Figure 84 (Part 3 of 7). WorkSpace On-Demand WSODCID.RSP Response File

```

* ----- *
* Install Target Directory *
* *
*   Directory of \ibmlan\rpl support *
*   ie. \ibmlan\rpl\bb10.US *
* ----- *
NCOInstallAll.TargetDir=\ibmlan\rpl\BB10.US

* ----- *
* Install Type *
* *
*   Valid Parms: *
*   None=No required files, use for Selective Install of NC Components *
*   All=NC Install of Required Files (default) *
* ----- *
NCO_RequiredFiles.Selection=All
* ***** *
*           H A R D W A R E   S U P P O R T *
* ***** *
* ----- *
* Predefined Machine Class *
* *
*   Valid Parms: *
*   None=Do not install any Predefined Machine Classes (default) *
*   (VGA system only) *
*   All=Install all Predefined Machine Classes *
* ----- *
NCO_MachineClass.Selection=All

* ----- *
* Printer Support *
* *
*   Valid Parms: *
*   None=None Do not install any print drivers on Server *
*   All=All Install all printer drivers on Server (default) *
* ----- *
NCO_PrinterDrivers.Selection=All
* ***** *
*           S Y S T E M   S O F T W A R E   C O M P O N E N T S *
* ***** *

```

Figure 84 (Part 4 of 7). WorkSpace On-Demand WSODCID.RSP Response File

```

* ----- *
* Java 1.1.1 Virtual Machine *
* *
* Valid Parms: *
* 0=None (default) *
* 1=Java Support *
* ----- *
NCO_JavaVM.Selection=1
* ----- *
* Unicode Font *
* *
* Valid Parms: *
* 0=None (default) *
* 1=Unicode Font *
* ----- *
NCO_UnicodeFont.Selection=1
* ----- *
* OS/2 REXX Support *
* *
* Valid Parms: *
* 0=None (default) *
* 1=Rexx Support *
* ----- *
NCO_RexxBASE.Selection=1
* ----- *
* DOS Support *
* *
* Valid Parms: *
* 0=None (default) *
* 1=DOS Support *
* ----- *
NCO_DOSBASE.Selection=1
* ----- *
* WIN-OS2 Support *
* *
* Valid Parms: *
* 0=None (default) *
* 1=WIN-OS2 Support *
* ----- *
NCO_WINOS2BASE.Selection=1

```

Figure 84 (Part 5 of 7). WorkSpace On-Demand WSODCID.RSP Response File

```

* ----- *
* Multi Media Support *
* *
* Valid Parms: *
* 0=None (default) *
* 1=Multi Media Support *
* ----- *
NCO_MMBase.Selection=1

* ***** *
* NETWORK APPLICATIONS *
* ***** *

* ----- *
* TCPIP Base Applications *
* *
* Valid Parms: *
* None=None (default) *
* All=All (ftp, telnet, ftp daemon, telnet daemon) *
* ----- *
NCO_TcpipApps.Selection=All

* ----- *
* TCPIP DOS Box Support *
* *
* Valid Parms: *
* 0=None (default) *
* 1=All Tcip DOS Box Support *
* ----- *
NCO_TcpipDOS.Selection=1

* ----- *
* TCPIP Internet Dialer Support *
* *
* Valid Parms: *
* 0=None (default) *
* 1=All Tcip Internet Dialer Support *
* ----- *
NCO_TcpipInet.Selection=1

* ----- *
* Personal Communications - PCOMM *
* *
* Valid Parms: *
* 0=None *
* 1=PCOMM *
* ----- *
NCO_TcpipPcom.Selection=1

```

Figure 84 (Part 6 of 7). Workspace On-Demand WSODCID.RSP Response File

```

* ----- *
* Netscape Navigator Support *
* *
*   Valid Parms: *
*       0=None *
*       1=Netscape Navigator *
* ----- *
NCO_Netscape.Selection=1

```

Figure 84 (Part 7 of 7). WorkSpace On-Demand WSODCID.RSP Response File

#### Note

Do not try to install WorkSpace On-Demand into a directory other than \IBMLAN\RPL\BB10.US. The installation will work correctly, but WorkSpace On-Demand will not work as this path is hard-coded in the product.

Since WorkSpace On-Demand uses Feature Install for the installation, the default name of the second response file is NCINSTAL.RSP. This file drives the WorkSpace On-Demand installation process for the OS/2 Warp Server environment. It contains not only information about the LAN server environment but also the complete list of files belonging to WorkSpace On-Demand.

The NCINSTAL file contains a variable section at the start, which must be modified to the target Warp Server environment. Since the file also includes the complete file list of WorkSpace On-Demand, the size of this file is quite large (approximately 1.7 MB). Depending on the CID environment, it may be necessary to distribute this response file for CID installations across the wide area network.

For a low-speed network, you can reduce the transmission time by separating the variable section from the fixed part of the NCINSTAL.RSP file. The unchanged section can be delivered with the WorkSpace On-Demand image on a transportable media such as CD-ROM.

The variable part can be modified at the central site and then transmitted across the corporate network to the desired locations. You can then merge the two files using a REXX script to create the complete NCINSTAL.RSP file used to install WorkSpace On-Demand.

In our test installation, we created the variable section as a separate file named NCINVAR.RSP. The changes we made to this file are shown in bold in Figure 85 on page 224.

```

TopObjectID=NCInstallAll
NCInstallAll=(
    SubfeatureID=Noinst
    SubfeatureID=NCU_Client
    SubfeatureID=NCU_Server
    ObjectTitle[0]=WorkSpace On-Demand Install
    Mode=User
    PackageTitle=Integrated Install
    CompanyName=IBM
    VersionNumber=1.0
    VersionDate=7/16/97
    Selection=1
    FilesDone=2625
    TotalMediaUsed=1
    Variable=(
        Name=Ln
        Description=Language Code
        Value=EN
        TouchFlags=1
    )
    Variable=(
        Name=Country
        Description=Country Code
        Value=us
        TouchFlags=1
    )
    Variable=(
        Name=OSDir
        Description=Client Operating System Image Dir
        Value=BB10.{Country}
        TouchFlags=1
    )
    Variable=(
        Name=TargetDrive
        Description=Target Drive
        Value={RIPLDrive}
        TouchFlags=1
    )
    Variable=(
        Name=TargetDir
        Description=Target Dir
        Value=\ibm\lan\rp1\{OSDir}
        TouchFlags=1
    )
)

```

Figure 85 (Part 1 of 5). NCINVAR.RSP - The Variable Part of NCINSTALL.RSP

```

Variable=(
    Name=UninstallDir
    Description=Uninstall Dir
    Value=\ibm\lan\uninstal
    TouchFlags=1
)
Variable=(
    Name=TargetPath
    Description=Target Path
    Value={TargetDrive}{TargetDir}
    TouchFlags=1
)
Variable=(
    Name=UninstallPath
    Description=Uninstall Path
    Value={BootDrive}{UninstallDir}
    TouchFlags=1
)

Variable=(
    Name=SourceDrive
    Description=Source Drive
    Value=d:
)

Variable=(
    Name=SourceDir
    Description=Source Dir
    Value=\en\bb030.nc
)

Variable=(
    Name=SourcePath
    Description=Source Path
    Value=d:\EN\bb030.nc\install
)

Variable=(
    Name=ClientBootDrive
    Description=Client Boot Drive used for config.sys
    Value=Z:
)

```

Figure 85 (Part 2 of 5). NCINVAR.RSP - The Variable Part of NCINSTAL.RSP

```

Variable=(
    Name=InstallType
    Description=Install Type
    Value=RIPL
)

Variable=(
    Name=LANDrive
    Description=LAN Drive
    Value=E:
    TouchFlags=1
)

Variable=(
    Name=RIPLDrive
    Description=RIPL Drive
    Value={LANDrive}
    TouchFlags=1
)

Variable=(
    Name=ServerInstalled
    Description=Server Installed On Target Machine
    Value=0
)

Variable=(
    Name=GUI_Selectable
    Description=LAN Administration GUI Upgrade is selectable
    Value=0
)

Variable=(
    Name=RIPL_Selectable
    Description=RIPL Upgrade is selectable
    Value=0
)

Variable=(
    Name=TargetMCDir
    Description=Target Machine Class Dir
    Value=\\ibmlan\\rp1\\machines\\{OSDir}
    TouchFlags=1
)

```

Figure 85 (Part 3 of 5). NCINVAR.RSP - The Variable Part of NCINSTALL.RSP



```

Variable=(
    Name=TargetMCPPath
    Description=Target Machine Class Path
    Value={TargetDrive}{TargetMCDir}
    TouchFlags=1
)

Variable=(
    Name=MCRoot
    Description=Machine Class Root - used by ncrplcfg.exe
    Value=\
)

Variable=(
    Name=MCPPath
    Description=Machine Class Path
    Value={TargetDrive}{MCRoot}
)

Variable=(
    Name=CfgFile
    Description=Config.* File Name
    Value=config.nc
    TouchFlags=1
)

Variable=(
    Name=UninstallShadowFolder
    Description=Target for Shadow of Uninstall Object
    Value=<WP_NETWORK>
)

Dependency=(
    Setting=ServerInstalled
    Value=1
    ActionVariable=UninstallShadowFolder
    ActionValue=<LS_FOLDER>
    Condition=4
    ResolveTime=2
    Action=5
)

Dependency=(
    Setting=ServerInstalled
    Value=0
    ActionVariable=UninstallShadowFolder
    ActionValue=<WP_NETWORK>
    Condition=4
    ResolveTime=2
    Action=5
)

```

Figure 85 (Part 4 of 5). NCINVAR.RSP - The Variable Part of NCINSTAL.RSP

```

ObjectCreation=(
  ClassName=WPPProgram
  ObjectTitle[0]=Remove WorkSpace On-Demand
  SetupString=EXENAME={UninstallPath}\uninstal.exe^;
  MINIMIZED=YES^;
  OBJECTID=<WPOD_UNINSTALL>^;
  ICONFILE={UninstallPath}\uninstal.ico
  Location=<WP_ASDUNINSTFOLDER>
)
ObjectCreation=(
  ClassName=WPSshadow
  ObjectTitle[0]=Remove WorkSpace On-Demand
  SetupString=SHADOWID=<WPOD_UNINSTALL>^;
  OBJECTID=<WPOD_UNINSTALL_SHADOW>
  Location={UninstallShadowFolder}
)
MediaSet=(
  SetName=Media Set
  Media=(
    Type=2
    Capacity=400
    Description=Image on Drive D:
    MediaPath=d:\EN\bb030.nc\install
  )
)

```

Figure 85 (Part 5 of 5). NCINVAR.RSP - The Variable Part of NCINSTALL.RSP

We created a REXX script to automatically install WorkSpace On-Demand. This script checks the parameter list, merges the variable and fixed parts for the Feature Install to create one NCINSTALL.RSP file, then calls the WorkSpace On-Demand INSTALL.EXE program. Figure 123 on page 305 shows the WorkSpace On-Demand installation REXX script. In this instance, we started the installation from a command line, rather than using a Software Distribution tool such as NetView Distribution Manager/2.

The WSODINST.CMD program was called with following parameters:

<b>/B:e:</b>	The boot drive of the OS/2 Warp Server
<b>/L1:d:\wsod\wsodinst.I1</b>	The Log file
<b>/L2:d:\wsod\wsodinst.I2</b>	The Error log
<b>/S:d:\en\bb030.nc</b>	The source path
<b>/NN:e:\netscape</b>	The path for Netscape Navigator
<b>/R2:d:\wsod\wsodcid.rsp</b>	The WorkSpace On-Demand response file
<b>/RF:d:\wsod\ncinfix.rsp</b>	The fix part for Feature Install

<b>/RV:d:\wsod\ncinvar.rsp</b>	The variable part for Feature Install
<b>/X</b>	The indicator for CID installation

After the installation is finished, the machine must be rebooted again. In a CID environment, this will happen automatically because the installation program returns a code of xFE00, indicating a successful program termination with a required reboot. After this reboot, the GETRPL utility must be run to complete the WorkSpace On-Demand environment. To run this utility unattended, you may call the program with the /L parameter for the log file.

GETRPL /L:d:\WSOD\GETRPL.LOG /O:BB10.US.

You must be logged on as an administrator to run this command. See Chapter 5, "Installing WorkSpace On-Demand" on page 71, for a detailed description of the WorkSpace On-Demand installation.

**Note**

If you need more information about Feature Install, see the following Web page: [http://service.boulder.ibm.com/asd-bin/doc/en\\_us/home.htm](http://service.boulder.ibm.com/asd-bin/doc/en_us/home.htm).

## 9.5.2 Managing the WorkSpace On-Demand Server Using the Command Line Interface

You can manage a WorkSpace On-Demand server using the GUI interface or alternatively by using the new command line interface. In an enterprise environment, the command line interface can be more powerful and productive because the managed machine may be at a different location with a low-bandwidth connection. In this case, the command line interface is very useful to:

- Manage remote WorkSpace On-Demand management servers
- Automate the remote management of WorkSpace On-Demand management servers

### 9.5.2.1 Creating, Deleting, Modifying, and Querying Clients Using the CLI

The NET RIPLMACH command allows you to create new clients. You can use response files to create multiple new WorkSpace On-Demand clients. Figure 86 on page 230 shows the command syntax of the NET RIPLMACH command.

```
NET RIPLMACH /ADD /RS(P):response_file_pathname
              [/SE(RVER):ripl_server_name]

/ADD
    Specifies the add operation. This is a required
    keyword to add or create the machine.

/SE(RVER):
    Specifies the server name of the RIPL server
    where the RIPL requester definition is created.
    If you are on the server, this is an optional
    parameter and the default is the local machine.
    If you are on the administrator client, this is
    a required parameter.

/RS(P):
    Specifies the fully qualified path to a response file.
    The path can be a UNC name.
```

*Figure 86. Syntax for the NET RIPLMACH Command*

Using response files, you can create many WorkSpace On-Demand client definitions on a WorkSpace On-Demand server with a single invocation of the NET RIPLMACH command. You can also use the NET RIPLMACH command to modify client configurations by a response file. The syntax of the response file is the same as for the creation of new clients. The example shown in Figure 87 on page 231 was used to create ten new clients using the NET RIPLMACH command.

**Important !**

Each line in the response file must end with a valid character and not with spaces. The processing of the response file will fail if there are extra spaces at the end of any line. The NET RIPLMACH command will fail with a message such as:

WSCLT4: NET3952: You entered a value that is not valid for the /OS option.

### Invalid Parameter

In this example we did not use the /RES parameter but other testing has shown that only lower case x's are valid as separators for video resolution. For example, to express an 800 by 600 by 1024 screen at 75 Hertz refresh rate the correct parameter input would be:

```
/RES:800x600x1024@75Hz
```

Using an upper case X will result in an error message indicating that NET RIPLMACH has an invalid parameter.

```
[WSODCLT1]
```

```
/os:bb10.us  
/cl:isavga  
/mac:10005a123451  
/rem:"This is WorkSpace client 1"  
/ada:"ibm t-r shared ram family (UP/SMP, IBMTOK.OS2)"  
/dhcp:y  
/tcpn:rplclient1  
/tcpd:austin.ibm.com
```

```
[WSODCLT2]
```

```
/os:bb10.us  
/cl:isavga  
/mac:10005a123452  
/rem:"This is WorkSpace client 2"  
/ada:"ibm t-r shared ram family (UP/SMP, IBMTOK.OS2)"  
/dhcp:y  
/tcpn:rplclient2  
/tcpd:austin.ibm.com
```

Figure 87 (Part 1 of 4). Example for a Response File to Create New WorkSpace On-Demand Clients

```

[WSODCLT3]

/os:bb10.us
/cl:isavga
/mac:10005a123453
/rem:"This is WorkSpace client 3"
/ada:"ibm t-r shared ram family (UP/SMP, IBMTOK.OS2)"
/dhcp:y
/tcpn:rplclient3
/tcpd:austin.ibm.com

[WSODCLT4]

/os:bb10.us
/cl:isavga
/mac:10005a123454
/rem:"This is WorkSpace client 4"
/ada:"ibm t-r shared ram family (UP/SMP, IBMTOK.OS2)"
/dhcp:y
/tcpn:rplclient4
/tcpd:austin.ibm.com

[WSODCLT5]

/os:bb10.us
/cl:isavga
/mac:10005a123455
/rem:"This is WorkSpace client 5"
/ada:"ibm t-r shared ram family (UP/SMP, IBMTOK.OS2)"

[WSODCLT6]

/os:bb10.us
/cl:isavga
/mac:10005a123456
/rem:"This is WorkSpace client 6"
/ada:"ibm t-r shared ram family (UP/SMP, IBMTOK.OS2)"

```

*Figure 87 (Part 2 of 4). Example for a Response File to Create New WorkSpace On-Demand Clients*

```

[WSODCLT7]

/os:bb10.us
/cl:isavga
/mac:10005a123457
/rem:"This is WorkSpace client 7"
/ada:"ibm t-r shared ram family (UP/SMP, IBMTOK.OS2)"
/dhcp:y
/tcpn:rplclient7
/tcpd:austin.ibm.com

[WSODCLT8]

/os:bb10.us
/cl:isavga
/mac:10005a123458
/rem:"This is WorkSpace client 8"
/ada:"ibm t-r shared ram family (UP/SMP, IBMTOK.OS2)"
/ri:9.53.123.4
/ni:9.53.123.5
/dh:n
/ip:9.53.123.77
/subm:255.255.255.254
/tcpdn:austin.ibm.com
/tcpn:wsodclt8

[WSODCLT9]

/os:bb10.us
/cl:isavga
/mac:10005a123459
/rem:"This is WorkSpace client 9"
/ada:"ibm t-r shared ram family (UP/SMP, IBMTOK.OS2)"
/dhcp:y
/tcpn:rplclient9
/tcpd:austin.ibm.com

```

*Figure 87 (Part 3 of 4). Example for a Response File to Create New WorkSpace On-Demand Clients*

```

[WSODCLT10]

/os:bb10.us
/cl:isavga
/mac:10005a12345A
/rem:"This is WorkSpace client10 "
/ada:"ibm t-r shared ram family (UP/SMP, IBMTOK.OS2)"
/dhcp:y
/tcpn:rplclient10
/tcpd:austin.ibm.com

```

Figure 87 (Part 4 of 4). Example for a Response File to Create New WorkSpace On-Demand Clients

#### Note

The response file must be coded with the fully qualified path name (UNC names are allowed). If you do not provide a fully qualified path name, the NET RIPLMACH command will not accept the file with the /RS parameter.

If you want to delete WorkSpace On-Demand clients from a server, you can use the NET RIPLMACH command with the /DELETE parameter. The syntax of the NET RIPLMACH command with the /DELETE parameter is shown in Figure 88.

```

NET RIPLMACH  client_name /DELETE
               [/SE(RVER):rpl_server_name]

client_name
    Specifies the name of the WorkSpace On-Demand Client to be deleted.

/DELETE
    Specifies the delete Option

/SE(RVER):
    Specifies the server name of the RIPL server
    where the RIPL requester definition is created.
    If you are on the server, this is an optional
    parameter and the default is the local machine.
    If you are on the administrator client, this is
    a required parameter.

Example: NET RIPLMACH wsodclt10 /DELETE

```

Figure 88. Syntax for the NET RIPLMACH Command



If you want to delete multiple clients, you can use a REXX script as shown in Figure 89 on page 235 to automate this operation with a response file.

```
/* Sample REXX script to delete a number of clients
   given by a file with the client names */

parse upper arg dellist
call stream dellist, 'c', 'open read'
if result <> "READY:" then
do
    say "Error opening file"
    exit x2d('0812',4)
end
clt.0=0
i=0
do while lines(dellist)
    i=i+1
    clt.i = linein(dellist)
    clt.0 = i
end
call stream dellist, 'c', 'close'
i=1
do clt.0
    "net riplmach /delete "clt.i
    i=i+1
end
```

*Figure 89. Sample REXX Command to Delete Clients Using a Response File*

A sample input file for this script is given in Figure 90.

```
WSODCLT1
WSODCLT2
WSODCLT3
WSODCLT4
WSODCLT5
WSODCLT6
WSODCLT7
WSODCLT8
WSODCLT9
WSODCLT10
```

*Figure 90. Sample Response File to Delete Clients*

The client definitions made using the NET RIPLMACH command becomes active as soon as the command completes. There is no need to stop and restart the RPL service to activate the changes. This makes the command line interface very powerful and gives you a very high degree of flexibility in managing your WorkSpace On-Demand topology remotely.

You can also use the NET RIPLMACH command to gather information about a defined WorkSpace On-Demand Client. Using NET RIPLMACH without any parameters shows you a list of defined WorkSpace On-Demand clients with their associated operating system and status (enabled or disabled).

You can issue the NET RIPLMACH command with a specific client name that you wish to query, and the command will provide output as shown in Figure 91.

```
RIPL requester name      WSODCLT1
Remark                  This is WorkSpace On-Demand client 1
Requester definition enabled Yes
Network adapter address  10005A123451
Network adapter type     IBM T-R Shared RAM Family (UP/SMP,
                          IBMTOK.OS2)
Operating system version BB10.US
RIPL Machine Class name  ISAVGA
Boot drive              Z
Location of SWAPPER.DAT  Local
Video monitor            Default (Not applicable)
Video resolution         Default (Not applicable)
DHCP-enabled             Yes
DDNS-enabled             No
Machine IP address       None
Router IP address        None
Nameserver IP address    None
Subnet mask              None
IP hostname              None
IP domain name           None

Local printers attached to this requester:
  None

Network printers available to this requester:
  None

The command completed successfully.
```

*Figure 91. NET RIPLMACH - Sample Output for a Specific Client*

This command is useful if you want to query the WorkSpace On-Demand topology of a server remotely. You can periodically run a program as

shown in Figure 92 on page 237 to gather the information for the currently defined Workspace On-Demand clients attached to a particular server.

```
/* Sample REXX script to gather the client
   information of a server */

cltlist='clients.lst'
cltdata='clients.dat'

'del 'cltlist
'del 'cltdata

clt.0=0
i=0
do while queued() > 0
    pull entry
end /* do */

/* Query clients */

call stream cltlist, 'c', 'open write'
if result <> 'READY:' then
do
    say "Error opening file : "cltlist
    exit x2d('0812',4)
end /* do */

'net riplmach | rxqueue' /* Querying info to pipe */
```

*Figure 92 (Part 1 of 2). Sample Program to Gather Client Information*

```

do while queued() > 0
  pull entry
  i=i+1
  clt.i=entry
  say "Clt."i" = "clt.i
  clt.i = translate(clt.i)
  if pos('ENABLED',clt.i) = 0 & pos('DISABLED',clt.i) = 0 then
    do
      say "No valid client line : "clt.i
      i=i-1
    end
  else
    do
      clt.i=word(clt.i,1)      /* reduce line to the client's name */
    end
  end /* do */

  clt.0=i
  i=1
  do clt.0
    say 'Querying information for client : 'clt.i
    'net riplmach 'clt.i ' 1>>'cltdata
    i=i+1
  end
end

```

Figure 92 (Part 2 of 2). Sample Program to Gather Client Information

### 9.5.2.2 Automated WorkSpace On-Demand Client Definition

In a large enterprise network, you may need to manage thousands of clients spread over several hundred locations. To create the client definitions in the WorkSpace On-Demand servers, you must know the burned in network adapter addresses (universally administered addresses) of the clients in order to define them. Since your servers may reside in different locations, it can require a considerable effort to query the adapter information, create the new client definitions and apply the correct client name to a new piece of hardware.

In order to define a new client on a WorkSpace On-Demand server, you need at least the following information:

1. The name of the new client
2. The type of network adapter
3. The network adapter's universally administered address
4. The Machine Class

The name of a new client is typically predefined in an enterprise environment. This predefinition can be done at the central site where the information for all clients of the enterprise network is stored. The information about the type of network adapter can be predefined, too, since the clients will be ordered from the manufacturer in the configuration that you support.

In this section, we describe a process to automatically detect new clients and dynamically add these clients to the server's configuration. The process flow is displayed in Figure 93 on page 240.

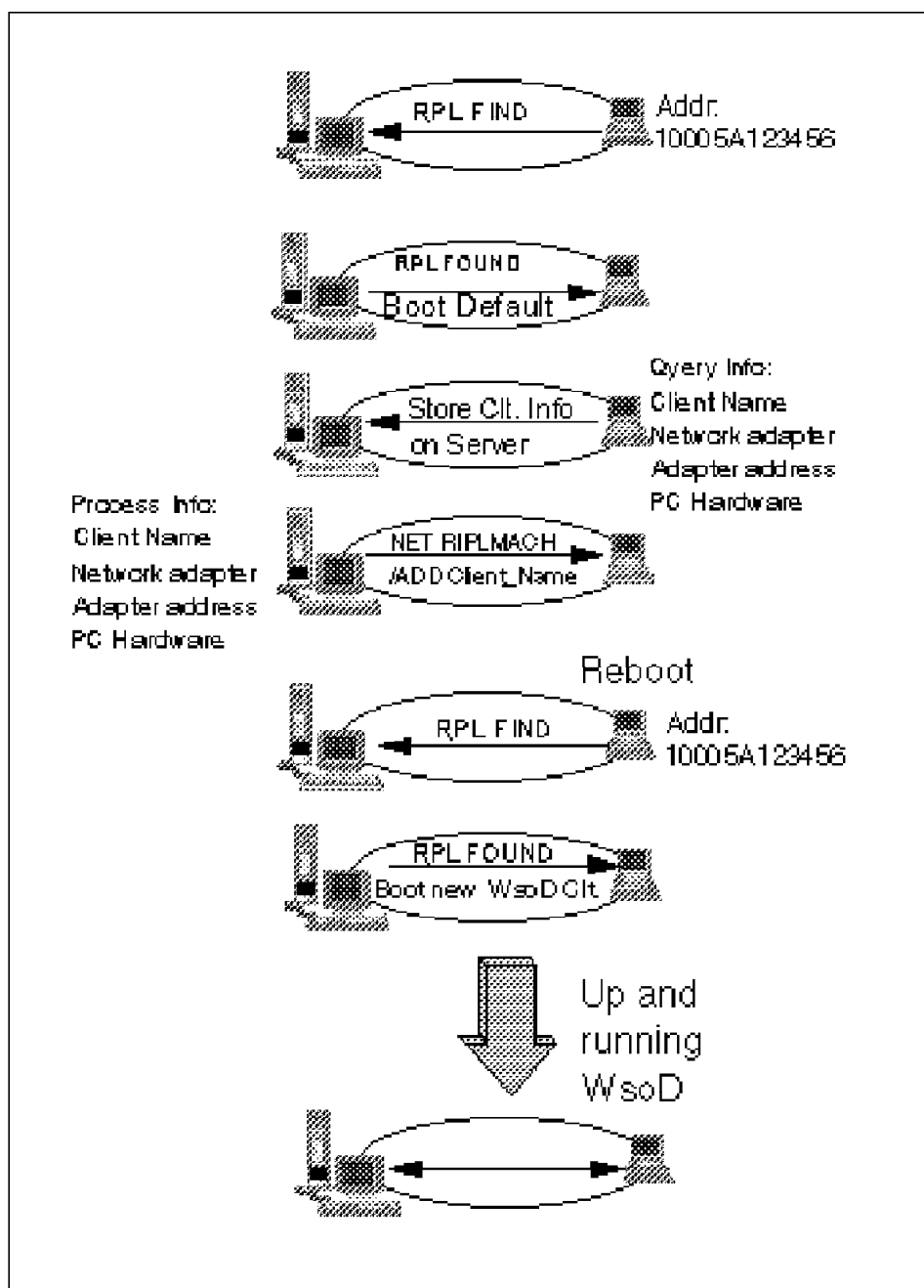


Figure 93. Automated Client Detection and Definition

To implement this process, follow these steps:

1. Install TME 10 NetFinity on the server as described in 9.3.3, “IBM TME 10 NetFinity in the WorkSpace On-Demand Environment” on page 206.

Using the TME 10 NetFinity System Information Tool enables the hardware detection for a new client.

2. Create a directory on the server with write access for the new clients to store their data.

In our testing, we created a directory D:NEWCLTS to store the configuration data. We created a network alias for this directory named NEWCLTS:

```
NET SHARE NEWCLTS=D:NEWCLTS.
```

This alias is necessary because we want to refer to this directory using a FIT file. An access control profile was created for this share to provide the RPLGROUP user group with read, write, create, execute, and delete access to this directory:

```
NET ACCESS NEWCLTS /ADD RPLGROUP:RWXCD.
```

This enables every new client to write its configuration data to this directory.

3. Write a program to query the client’s hardware configuration.

We created a REXX script named QUERYCFG.CMD, as shown in Figure 124 on page 313, to obtain the necessary hardware information. The script requests the station ID, and the user can type in the correct client name. We assume that in an enterprise environment, the names for the machines will be generated at a central site and that the person setting up the hardware will be provided with written information about the name of each machine.

Typically, every machine will have a sticker with its machine ID, to simplify the identification of the machines if they are switched off. The TME 10 NetFinity System Information Tool is called through the SINFG30 program. This program has command line options to store the data to a given file name. It also can store the data to a DB2/2 table, but we chose a simple ASCII file in our example.

We appended the contents of the LANTRAN.LOG file to the SINFG30 output file to complete the hardware information because the TME 10 NetFinity System Information Tool does not query the network adapter address. The hardware configuration file is stored as CLIENT\_NAME.data to that directory on the server where the NEWCLTS share is assigned to.

When the hardware information is stored, the client then waits for the completion of a server process that creates its correct WorkSpace On-Demand Client definition and then shuts down the system.

You can add more function to a batch process like this. One possible extension would be to test for the presence of a local disk drive and possibly format this drive if no valid file system is found. For this example, however, we wanted to give an example for the general technique used; so we made it as simple as possible.

4. Create a model client definition for a new client.

You can do this by defining a client to WorkSpace On-Demand using an adapter address that contains wild cards. For example, we defined an entry with the adapter address 00062???????. Each wildcard ? position is not tested when the server detects a RPL FIND frame. Hence, a match will be found for any network adapter that has an address beginning with 00062.

If you have many different network adapter types, you can create a model client definition for each adapter type that you use. Since the UAAs are vendor- and type-specific in the first 6 numbers, you can create model definitions with adapter addresses that have at least the positions 7 - 12 replaced with the ? wildcard specification.

In our testing, we created client definitions for the PCI Auto Wake on LAN and the MADGE Token-Ring adapters. The resulting RPL.MAP file is shown in Figure 94.

```
; workstation records
1000FFFFFFFF DEFALT40 FITS\DEFALT40 DELLSRV Z    , R_240_OTK
00062??????? MODEL FITS\MODEL DELLSRV Z    , R_BB10_US_OTKTRP
0000F??????? MDGMODEL FITS\MDGMODEL DELLSRV Z    , R_BB10_US_OTKMDGSMT
```

Figure 94. RPL.MAP file with Model Client Definitions

Every new PC with an adapter address matching the specified positions from the model definition will start a RIPL process and boot a system as configured for the model.

**Note**

We did not test a large number of adapter types; so, while it might appear that different versions of a similar adapter (for example, PCI and ISA versions) have the same adapter address in the first six positions, they may in fact use different drivers. You should test this approach for the types of adapters you intend to use in your enterprise.



5. Change the CONFIG.SYS of the model clients.

In the IBMLANRPLMACHINESMODEL directory, change the CONFIG.SYS and replace the RUNWORKPLACE statement with a command line (CMD.EXE) as shown in Figure 95. The change of the RUNWORKPLACE statement forces the new client to process the QUERYCFG.CMD instead of PMLOGON.EXE to get the hardware configuration.

```
SET RUNWORKPLACE=Z:\OS2\CMD.EXE /k z:\util\querycfg.cmd

DEVICE=Z:\NETFIN.40\BIN\SYSINFO.SYS
```

Figure 95. Model Client CONFIG.SYS Changes

The SYSINFO.SYS driver must be added to this CONFIG.SYS to support the TME 10 System Information Tool that gathers the hardware information.

6. Change the FIT files for the model clients.

Make changes as shown in Figure 96 to the client's FIT file to gain access to the NetFinity directory, to the UTIL directory where the QUERYCFG.CMD file is stored, and to the target directory on the server where the output files are stored.

```
; NetFinity and Utility support for automated detection of new clients
z:\NETFIN.40      BB10.US\NETFIN.40
z:\UTIL           BB10.US\UTIL
z:\*.data         \\DELLSRV\NEWCLTS
z:\*.ok           \\DELLSRV\NEWCLTS
```

Figure 96. FIT File Changes for a Model Client

**May Also Run With OS/2 Warp Definition**

A model client definition can possibly be made for an OS/2 Warp client with the same functionality, though this has not been tested. The REXX scripts shown here are specific to WorkSpace On-Demand.

7. Write a program to automatically generate the new WorkSpace On-Demand clients. We created a REXX program that automatically adds newly detected clients to the WorkSpace On-Demand server. The program has the following parts:

- A main process that runs every 60 seconds

- The procedure ScanClt:

This procedure scans the client\_name.data file and searches for an entry with the universally administered address. The value for the network adapter address is stored in a global variable called address.

- The procedure QueryAdp:

This procedure fills the variable adptype with the name of the network adapter type based on the list in the AdpList STEM variable.

- The procedure LogMsg:

This procedure is used to log the program execution.

- The procedure CreateClt:

This procedure is used to create the new clients. It scans through the RPL.MAP file and checks for a workstation record with the adapter address of the new client.

Any client that does not have an entry specified in the RPL.MAP file, but for which the adapter address fits the global mask defined for the model, will be booted like the model, and the program will create an entry in the RPL.MAP file. This entry is initially disabled. This line is deleted from the RPL.MAP file because otherwise the NET RIPLMACH command will fail with an error saying that there is already an entry with a matching adapter address in the RPL.MAP file.

The name for the model is the second word in this line, and based on this model name, the correct adapter type is found using the procedure QueryAdp. In a production environment, you probably will not define the supported adapters in this way, but will do something more like that shown in Figure 97 on page 245.

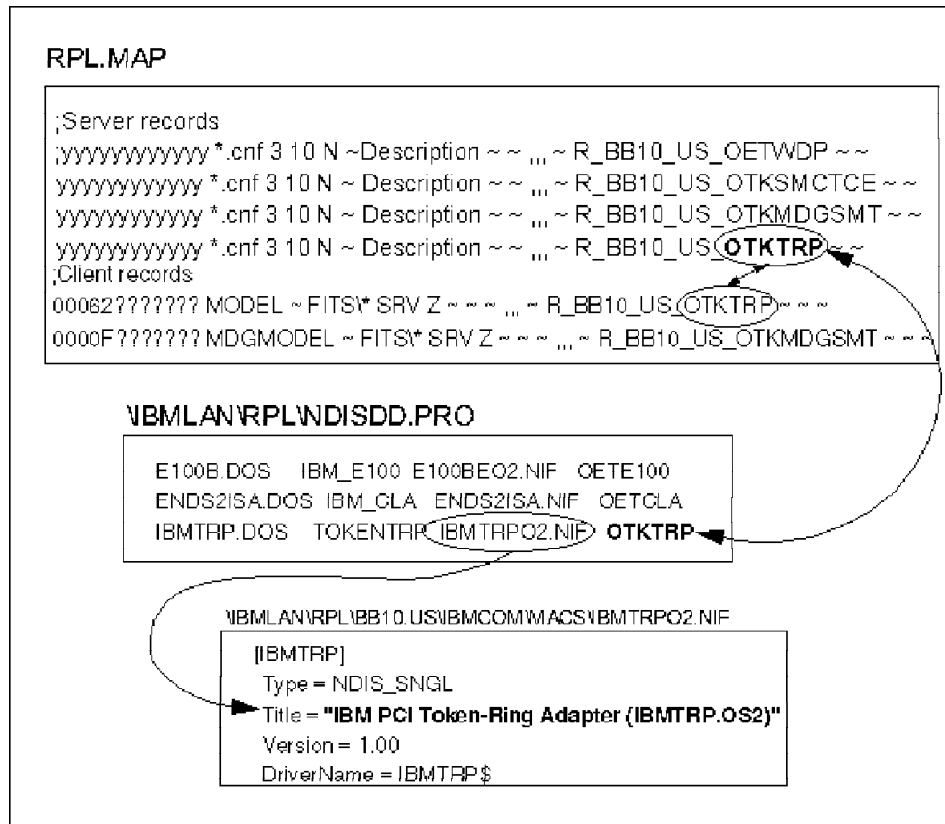


Figure 97. Dynamic Detection of Network Adapter Name

If you scan the client line with the model definition, you find the pointer to the server record. Using the information in the activated server records of the RPL.MAP file, the corresponding entries in the NDISDD.PRO and finally the text from the .NIF file, you can obtain the names for the supported network adapters.

This is the safest way to have the correct values for the /ADA parameter of the NET RIPLMACH command because you can scan through the files during the execution of the program. The values must not be hardcoded in the program. They will be set dynamically.

The parameters for the NET RIPLMACH command are set to the following values:

- The Machine Class is set to ISAVGA
- The operating system to BB10.US
- The MAC address to the value of the variable address
- The client name to the value of the variable cltname

- The remark to *WorkSpace On-Demand client* + the client name
- The adapter type to the value of the variable *adptype*
- The stdout and stderr output is redirected to append to the Log file

The NET RIPLMACH command is then invoked with these parameters.

After finishing the CreateClt procedure, the main program renames the *client\_name.data* file into *client\_name.ok*. The existence of this file is the signal for the QUERYCFG.CMD program running on the client that the client definition is done on the server. The client can be rebooted as soon as this file exists, and when the reboot is complete, the client will have a fully functional WorkSpace On-Demand configuration. You can find the complete listing of the program in G.5, “ADDNEWCL.CMD: Automatted WorkSpace On-Demand Client Definition” on page 314

The following figures show the change in the RPL.MAP file during the process of automated client definition.

#### 1. The RPL.MAP file with only the model clients defined

```
; workstation records
1000FFFFFFFF DEFALT40 FITS\DEFALT40 DELLSRV Z   ,, R_240_OTK
00062??????? MODEL FITS\MODEL DELLSRV Z   ,, R_BB10_US_OTKTRP
0000F??????? MDGMODEL FITS\MDGMODEL DELLSRV Z   ,, R_BB10_US_OTKMDGSMT
```

Figure 98. RPL.MAP File with Model Client Definitions

#### 2. The RPL.MAP file when new clients are booted for the first time

```
; workstation records
1000FFFFFFFF DEFALT40 FITS\DEFALT40 DELLSRV Z   ,, R_240_OTK
00062??????? MODEL FITS\MODEL DELLSRV Z   ,, R_BB10_US_OTKTRP
0000F??????? MDGMODEL FITS\MDGMODEL DELLSRV Z   ,, R_BB10_US_OTKMDGSMT
0000F62A5B29 MDGMODEL FITS\MDGMODEL DELLSRV Z   ,, D_BB10_US_OTKMDGSMT
000629844293 MODEL FITS\MODEL DELLSRV Z   ,, D_BB10_US_OTKTRP
```

Figure 99. RPL.MAP File with Model Client Definitions

#### 3. The RPL.MAP file when the server process has defined a new client

```
; workstation records
1000FFFFFFFF DEFALT40 FITS\DEFALT40 DELLSRV Z   ,, R_240_OTK
00062??????? MODEL FITS\MODEL DELLSRV Z   ,, R_BB10_US_OTKTRP
0000F??????? MDGMODEL FITS\MDGMODEL DELLSRV Z   ,, R_BB10_US_OTKMDGSMT
0000F62A5B29 MDGMODEL FITS\MDGMODEL DELLSRV Z   ,, D_BB10_US_OTKMDGSMT
000629844293 ITSOLAB2 FITS\ITSOLAB2 DELLSRV Z   ,, R_BB10_US_OTKTRP
```

Figure 100. RPL.MAP File with Model Client Definitions

As you can see, one client is already installed, and another one of the type MDGMODEL is not yet processed. This entry is still disabled by the D instead of the R at the beginning of the workstation type field of that line.

#### **Avoid Duplicate NetBIOS Names**

Every new client that is not yet defined will be booted with a client name as defined in the model configuration. We have not tested an environment with multiple machines that had the same network adapter installed, but it is likely that only one model client at a time can become active and run through this process. As soon as the client is rebooted, the next machine can be processed. You may extend the procedures in your environment to resolve this conflict.

The process and procedures described here are only a basic set of possible functions. The intention is to give an easy example of how to add new clients to an existing WorkSpace On-Demand environment. By using standard tools such as the TME 10 NetFinity System Information Tool, you can customize the WorkSpace On-Demand environment to meet your enterprise's requirements.

TME 10 NetFinity provides information on the hardware configuration of your client machines. If you use IBM hardware, the exact model description is shown in the output data file. In our test, we used an IBM Personal Computer 350, Machine Type/Model 6586/5SH. If we look at the output of the TME 10 NetFinity System Information Tool, we find a section as shown in Figure 101 on page 248.

```

***** Model and Processor Information *****
Model Name           : IBM Personal Computer 350
Machine Type/Model   : 6586/5SH
Processor            : Pentium( tm )
Math Coprocessor     : Pentium( tm )
Number of Processors : 01
Processor Speed      : 100 MHz
System Board Speed   : 50 MHz
Internal Processor Cache : Enabled
Model                : FC
Submodel             : 01
BIOS Revision        : 00
BIOS ROM Date        : 10-26-1995
Build ID             : LPKT53A
Dedicated IRQ Levels : 0 1 2 8 12 13 14 15
Shared IRQ Levels    : 11

```

*Figure 101. TME 10 NetFinity System Information Tool Output*

This shows that the tool gets the correct Type/Model ID for IBM PC hardware. If you create Machine Classes as described in Chapter 8, “WorkSpace On-Demand Machine Classes” on page 149, for every Type/Model combination that you plan to support, it is very easy to create WorkSpace On-Demand client definitions that exactly match the hardware you are using. These client definitions will then be as precise as possible and not generalized to the ISAVGA Machine Class as in our example.

The TME 10 NetFinity System Information Tool also shows the memory size installed in the client. With this in mind, another useful parameter may be to customize the swap-file location (local or remote) depending on the memory size of the client. This parameter can be set in the parameter list of the NET RIPLMACH command.

For non-IBM hardware, you should build the Machine Classes corresponding to the video adapters installed because the video adapter is the most complex configuration item for a Machine Class. Figure 102 on page 249 shows an extract of the output from TME 10 for the video configuration of a non-IBM PC.

```
***** VGA-compatible controller *****  
Manufacturer           : CIRRUS LOGIC  
Display controller     : VGA-compatible controller  
Class Code             : 0003  
PCI Bus Number 0      : Device Number 02  
Vendor ID              : 1013  
Device ID              : 00A8  
Revision ID            : 00FC
```

Figure 102. TME 10 NetFinity System Information Tool Video Adapter Detection

The specific video adapter is shown in the lines Class Code, Vendor ID and Device ID. If you create and name your Machine Classes based on the information that the System Information Tool provides, you can create Machine Class definitions for non-IBM hardware almost as easily as for IBM hardware.

The following sections provide some examples of possible extensions to the sample procedures. You may find more information in the output file that is useful to fit the Machine Class and client definition to your environment.

### 9.5.2.3 Using the Command Line to Query Machine Classes

In an enterprise environment, it may be useful to check the WorkSpace On-Demand management servers from time by time to determine the Machine Classes defined at each server. This may be necessary to ensure that your server's definitions are still consistent with the actual supported hardware. If any Machine Class that is defined at one location is not defined on another server or even worse, has a different content, you can use the NET RIPLMCLASS command to gather the information.

The NET RIPLMCLASS command gives you either an overview of the installed Machine Classes or a detailed view of the configuration for a specific Machine Class. For the correct syntax of the command, see the online WorkSpace On-Demand *Administrator's Guide*. This command cannot be used to modify a Machine Class. To obtain a list of the installed Machine Classes on your server, you can use the following command:

```
NET RIPLMCLASS /OS:BB10.US
```

This command will display a list of installed Machine Classes on your server. If you want to obtain the information for a different server in your network, you must specify the /SE: option with the server name. For example, NET RIPLMCLASS /OS:BB10.US /SE:MYSRV shows all defined Machine Classes for the server MYSRV. Figure 103 on page 250 shows the output of NET RIPLMCLASS /OS:BB10.US.

```
Machine classes defined on WSODTEST for operating system version bb10.us
ibm300gl          ibm350          ibm730
ibm750           isavga          pcivga

The command completed successfully.
```

Figure 103. Output of NET RIPLMCLASS Command

If you use the NET RIPLMCLASS with the Machine Class name specified, the command will provide output as shown in Figure 104.

```
Machine class name      ibm350
Remark                 IBM PC350 ·6586-57H“, PCI/ISA, S3 Trio64/64V+
Bus type               ISA
Diskette drives supported Yes
Parallel devices supported Yes
Serial devices supported Yes
Hard disk type         IDE
Keyboard type          101
Keyboard mode          3
Video type             S3 Trio64/64V+ ·32 Bit“ Version 2.81.10
Video monitor
Video resolution
Mouse type supported   PS/2 Mouse
CD-ROM device support
Multimedia device support
SCSI adapter support
Network adapter type
Local printers attached to this requester:

Network printers available to this requester:

The command completed successfully.
```

Figure 104. Output of NET RIPLMCLASS IBM350 /OS:BB10.US

You can redirect the output of this command to an ASCII file and collect the files for all your installed WorkSpace On-Demand servers to ensure that the definitions are correct and consistent. If you have the need to modify your Machine Classes remotely, you can create utilities to modify the .INI files for this Machine Class. Normally, you would correct the files for a Machine Class at your central site and deliver the corrected files to your remote branches either across the corporate network or on removable media such as CD-ROM or diskette.

#### 9.5.2.4 Using the Command Line Interface to Create and Change Public Applications

How you administer your applications and users depends on your enterprise's administrative structure and the knowledge of the people in your remote locations. In your environment, many users may need to use the same set of public applications; so it may make sense to define



applications and grant them to users through an automated process. Alternatively, the administrators (if any) in your remote locations may not be familiar with such tasks.

The command line interface has the same capabilities as the GUI, but has the advantage that it can be run remotely in a full screen session; or you can run it unattended using response files. Therefore the command line interface gives you the chance to support the remote administrators or to perform all administration from a central site.

The command line interface is enhanced to support the new WorkSpace On-Demand application parameters as well as the standard OS/2 and DOS public applications. You can use the command line interface to create WorkSpace On-Demand public applications, to modify the application specific parameters, to assign the applications to users, or to make the application parameters user-specific.

Another big advantage in using the command line interface is the standardization of your application definitions. By using default scripts to define your applications, you avoid having different application definitions across your different locations. In large environments where you may have many locations and many users, the use of standardized batch processes or REXX scripts may be the only way to provide a consistent environment.

The use of response files to create applications, set the application parameters to the default values and assign applications to your users is a powerful way to automate your WorkSpace On-Demand administration and apply the necessary standards. Figure 126 on page 322 in Appendix G, "Samples for Managing WorkSpace On-Demand in the Enterprise" on page 297, shows a REXX script that creates WorkSpace On-Demand applications and sets the application parameters using the command line interface.

The response file shown in Figure 127 on page 329 was used as a response file to create two WorkSpace On-Demand applications and assign these applications to users:

- An application that only invokes a command line
- An application for PM Camera

The application parameters are not provided; so some applications such as PM Camera will need more application parameters. The response file is included to show the base capabilities of the command line interface in conjunction with the use of response files.



---

## Appendix A. Tuning WorkSpace On-Demand

This Appendix provides information about WorkSpace On-Demand performance. The information in this appendix is a summary of performance information that can be found on WorkSpace On-Demand Web page (URL <http://www.software.ibm.com/os/warp/workspace/>). You should refer to this Web page for more details and up-to-date performance data.

---

### A.1 Performance Considerations

The following items were found to have an impact on overall system throughput while doing performance testings on WorkSpace On-Demand. These items are presented in the recommended order for investigating these enhancements.

#### Note

The results you experience will be highly-dependent upon the number of clients on your network, the number and type of applications the clients use, and the interactions demands those client applications place upon the server.

1. Use OS/2 Warp Server Advanced HPFS386 file system.

In a WorkSpace On-Demand environment, the file access for the client is performed at the server. The performance of the file server will effect both the client IPL time as well as the application execution. The application execution can be effected by the server access to both executable and data files. Thus, the faster the file server performs, the better the client performance. OS/2 Warp Server Advanced, with the HPFS386 file system, provides the fastest performance as a file server and should be considered where the number of clients and application activity exceed the capabilities of OS/2 Warp Server Entry to provide acceptable performance.

2. Use a large HPFS386 (disk) cache

As mentioned above, the performance as the file server is very important. Disk cache size is the major tuning parameter to maximize the file server performance. To maximize the cache hit rate, adequate memory is required.

3. Use a server with a fast processor

If you have a lot of clients per WorkSpace On-Demand server and a large HPFS386 disk cache on the server, a fast processor performance can enhance the performance. In an environment where the server has enough memory for cache, the server can process the client request for IPL, application execution, and data by reading from/writing to the cache.

An SMP system can provide additional performance for client/server applications where the server portion for the application is also running on the WorkSpace On-Demand server.

4. Use a fast LAN (16 Mbit token-ring or 100 Mbit Ethernet)

As was noted in item 1 above with IPL files, application execution and data files, there can be substantial data traffic on the LAN. As both server and client CPU, memory and hard drive performance continue to improve, it is possible that the LAN speed can become a performance bottleneck.

5. Use multiple network adapter cards on the server.

You can have multiple LAN adapter on the server to balance the load. If you have too many clients connected to the same LAN and the LAN utilization becomes too high and the traffic is constrained, you need to separate your LAN into multiple segments.

6. Configure the clients to swap to local hard disk.

Configuring the clients to swap to local hard disk removes additional network traffic and server load to process for swapping. This requires that each client have a local hard disk for swapping.

7. Adequate memory in the client to minimize swapping.

As with stand-alone systems swapping can effect the performance of any application on the machine. If you are swapping to a local hard drive then performance impact is the same as you would find when adding memory to a stand alone system.

For a client that uses the remote swap files to the server, adding more memory will reduce the swapping activity as well as lower the LAN traffic. Thus, it will improve the total system performance.

---

## A.2 Java Runtime Performance Considerations

The Java runtime is optimized to run in video modes that support 256 or more colors. You will see increased load time for Java applications when running in VGA (16-color) mode.

---

## Appendix B. WorkSpace On-Demand Hints and Tips

This Appendix provides hints and tips on the WorkSpace On-Demand product that were learned by the residents and others who worked with the product to produce this redbook.

---

### B.1 OS/2 Warp Server SYS0071 Errors When Attempting to Start an Application

In cases where OS/2 Warp Server (not OS/2 Warp Server Advanced or OS/2 Warp Server Advanced - SMP) is operating under heavily loaded conditions, occasional instances of application initialization failure have been observed. The symptom of this failure is that the application fails to initialize and returns a SYS0071 message indicating that the server has run out of resources.

LAN FixPak IP8506 addresses this condition. LAN FixPak IP8506 is automatically installed when the WorkSpace On-Demand Manager is installed following the installation of OS/2 Warp Server. When IP8506 is installed, the server heuristics section of the IBMLAN.INI file is modified to include an additional heuristic, heuristic bit #21. This heuristic extends the timeout value of the scavenger thread. If you observe SYS0071 failures, *do not modify this value until you have discussed the problem with the support center.*

---

### B.2 WorkSpace On-Demand client Network Adapter Configuration Failures

If you encounter problems when booting a WorkSpace On-Demand client, the following information is useful for problem determination. Generally speaking, the initialization of a WorkSpace On-Demand client occurs in six phases. In this section, we discuss the RPL failures you may encounter during each of these initialization phases and the most likely cause(s) of these failures.

In a successful RPL sequence, as soon as the RPL feature gains control and initializes the adapter, information is shown on the display of the WorkSpace On-Demand client. This information takes the form of a two-letter message followed by a number (See Appendix E, "RIPL Workstation Error/Status Messages" on page 277, for a discussion of these messages). Messages appear on the screen as additional information becomes available. The state of the WorkSpace On-Demand client's display screen is useful as a guide to the initialization phase(s).

### B.2.1 Phase 0: Message Numbers Missing from Screen

If the adapter fails to initialize, messages and their accompanying numbers fail to appear on the screen. In cases like this, you should check for the following:

1. Hardware failure
2. Cable disconnected at the adapter/machine interface
3. Machine/BIOS not configured to boot from network
4. Adapter not configured/capable for RPL

### B.2.2 Phase 1: RQ Counter Increments, with No Other Result

This counter increments each time a RPL FIND frame is broadcast by the requesting WorkSpace On-Demand client. This condition indicates that the WorkSpace On-Demand server is not responding. Check for the following:

1. Insure that REMOTEBOOT services are started on the WorkSpace On-Demand server.

#### Note

If you are unable to start REMOTEBOOT services and you receive the following messages, insure that you have configured and installed support for IEEE 802.2 in the WorkSpace On-Demand server's MPTS.

```
C:\>net start remoteboot
The REMOTEBOOT service is starting..
The REMOTEBOOT service could not be started.
NET3064: There is a problem with the file
(unknown): NET3088: View your
error log (type NET ERROR) for details..
```

For more information, type HELP NET3064.

Program	Message	Time
REMOTEBOOT	3299	09-11-97 02:55pm
RPLNET2.DLL: DIR.OPEN.ADAPTER failed.		
The adapter's system process is not installed (DLC error 56H)		
REMOTEBOOT	3256	09-11-97 02:55pm
NET3256: An unrecoverable error occurred in the dynamic link library of the service.		

2. Verify that the WorkSpace On-Demand client's MAC address matches the workstation record entry in the RPL.MAP file.

3. Check for a *downstream* cabling problem on the LAN.
4. Verify that your requestor has not been disabled at the WorkSpace On-Demand server.

### B.2.3 Phase 2: Failure During DOS Driver Initialization

At this point, failure to load and initialize the DOS NDIS drivers can result in a machine hang (usually with the protocol manager version on the screen) or with a message to the effect that the correct driver could not be found.

Check for the following:

1. Error(s) in the Boot Block Definition (.CNF) file
2. Error(s) in the DOS PROTOCOL.INI file
3. Bug or Restriction in the DOS NDIS driver

#### HINT

This stage of the RPL process uses the .CNF file to load specific files and drivers into the requesting machine's memory instead of the actual DOS operating system. As a debugging tool, you can create a real DOS IPL image diskette to test for errors in the Boot Block Definition (.CNF) file. See the *Network Administrator Reference Volume 3 for OS/2 LAN Server* for information on creating a DOS IPL image diskette.

### B.2.4 Phase 3: 3174 Message - Can't Find OS2LDR.MSG

An error at this stage of the RPL process indicates that the SMB trail that locates and opens the OS/2 loader message file has failed. Check for the following:

1. Access Control Error(s)

Remember that the WorkSpace On-Demand client's *COMPUTERNAME* is used as the *USERNAME* and that the user is a member of the group *RPLUSER*.

2. Duplicate Machine Names (COMPUTERNAME)

Add Name Query Failure - this is a NetBIOS protocol error.

3. Error(s) in the File Index Table (FIT)

If you change a server's name, either manually or by using the CHGSRVR utility, you must update the corresponding share name in the FIT.

#### **HINT**

Access to a protocol analyzer such as an IBM DatagLANce or Network General Sniffer simplifies diagnosing problems encountered in this phase of the RPL process. The SMB Tool (SMBTOOL.EXE) provided with Warp Server can also be used to determine which files cannot be located on the server.

### **B.2.5 Phase 4: Various Early Boot Failures**

These types of failures are characterized by the appearance of the OS/2 block in the upper-left corner of the WorkSpace On-Demand client's display, after which the machine either hangs or posts a message to the effect that a specific file cannot be found (even though it does exist). In both cases, using the <ALT> + F2 key sequence to display the CONFIG.SYS processing sequence helps to locate the point of failure. Use the following procedures when dealing with these types of problems:

#### **1. Insufficient Memory**

In cases where the machine hangs, the most likely cause is insufficient memory when loading BASEDEVs. At this stage of the RPL process, the mini-file-system driver, MFSD20.SYS, is still active and in control. MFSD20.SYS can only address 640 KB of RAM. Remember that MFSD20.SYS is loaded by the Boot Block Definition or .CNF file. Keep in mind also that BASEDEVs are processed before any other device driver statements in CONFIG.SYS. There are two ways to address insufficient memory problems:

- If RAM shadowing is enabled, use the SETUP utility to modify the machine's BIOS and disable it.
- Check with the IBM Support Center to see if a kernel fix is available to load a smaller kernel.

#### **2. File Not Found**

Use the same methodology as in B.2.4, "Phase 3: 3174 Message - Can't Find OS2LDR.MSG" on page 257, to determine why a specific file cannot be found on the server.

### **B.2.6 Phase 5: Failure of the OS/2 Redirector (NETWKSTA.200)**

As before, diagnosing this problem is made much easier by utilizing the <ALT> + F2 key sequence during the boot process to display the order of CONFIG.SYS processing. Possible reasons for NETWKSTA.200 failing to load or initialize include:

#### **1. OS/2 Bind Failure**



Check the OS/2 PROTOCOL.INI for errors

2. Incorrect or Corrupted MAC Drivers

Check LANTRAN.LOG for error messages

3. CONFIG.SYS Driver Ordering

Check CONFIG.SYS and insure that all network drivers are loaded *PRIOR* to loading NETWKSTA.200.

---

### B.3 Access Control Issues

Remember that with Warp Server (386HPFS) access control permissions are part of the file structure itself and are inherited when you create new files or subdirectories *AS LONG AS YOU ARE LOGGED ON AS AN ADMINISTRATOR*. With OS/2 Warp Server Entry, access control permissions *MUST BE REAPPLIED* whenever the file system is modified.

---

### B.4 File Index Table (FIT) Considerations

Keep in mind that there are two types of FIT files: a *STATIC* FIT required by the workstation or client and a *DYNAMIC* FIT or USERFIT extension that is applied when a user launches an application. This discussion is limited to the static, or client FIT.

The static FIT is created when you create a client. The static FIT is created by combining the default FIT, Z:\IBMLAN\RPL\FITS\DFBB10US.FIT and the FIT extension, Z:\IBMLAN\RPL\MACHINES\BB10.US\NCNEW.MC\NCNEW.FIT. The static FIT and MFSD20.SYS are stored together, along with other device drivers, in a 64 KB segment. Currently, MFSD20.SYS is about 40 KB in size. This limits the static FIT to approximately 14 KB in size.

#### HINT

If you absolutely have to have a FIT larger than 14 KB (See the performance implications of a large FIT file in the next paragraph), you can modify the Boot Block Definition file (.CNF). By moving the RPL boot record **AFTER** the mini-file-system driver in the .CNF file, you can support a FIT of up to approximately 21 KB in size.

There is a performance implication associated with the size of the static FIT file. Whenever a reference to a file on the WorkSpace On-Demand client's boot drive is resolved, the *ENTIRE* FIT file is scanned from beginning to end. This is a table scan which does not use any kind of sophisticated search algorithm like a binary search tree, nor does it terminate after finding an

initial match ... the table is always scanned to completion. Obviously, keeping the FIT table as small as possible improves performance.

### B.4.1 WorkSpace On-Demand FIT Files

File Index Tables are used to map application file references to the physical location of the file. There are three types of FIT's:

<b>Machine Class</b>	Typically contains hardware-specific file mapping references (for example, SVGA video support).
<b>Default</b>	Contains default mappings used in the creation of each client machine_name FIT file.
<b>User Specific</b>	Contains mappings specific to the user.

During user logon or refresh the FIT is dynamically created/changed. This table is composed of the Machine Class, default and any user specific FIT files which are dynamically combined.

### B.4.2 Using RDRDEBUG to View the Client FIT

Shipped on the WorkSpace On-Demand CD-ROM is a program that will allow you to inspect the FIT that is being used by a WorkSpace On-Demand client. This program is not installed in the install process so you will need to install it manually before using it.

The program RDRDEBUG.EXE is included in the SMBTOOL.ZIP file located in the SERVICE\BMLANSRV\IP08506 directory on the WorkSpace On-Demand CD-ROM. You need to copy the SMBTOOL.ZIP file from the CD-ROM to a temporary file on your server. Unpack this zipped file and copy the file RDRDEBUG.EXE to a location where it can be accessed by the WorkSpace On-Demand client. For example, it could be placed into the IBMLANRPLBB10.USOS2 directory.

To execute the program you will need to set up an OS/2 command prompt (CMD.EXE) as an application on the WorkSpace On-Demand client and give the user access to the command prompt application. Start the OS/2 command prompt session on the WorkSpace On-Demand client and issue:

```
RDRDEBUG /F | MORE
```

The **/F** directs the RDRDEBUG program to display the information from the currently active FIT to the OS/2 window on the WorkSpace On-Demand client. The **| MORE** allows you to view one page at a time the many pages of FIT information that will be displayed.

The information from the FIT will be displayed in both character and ASCII format.

---

## B.5 Changing System File Considerations

If you have some change in a system file such as CONFIG.SYS, PROTOCOL.INI, and so forth for a specific client, you have to be careful when you change the client definition using WorkSpace On-Demand administration GUI. System files located under the d:\IBMLAN\RPL\MACHINES\<machine name>\ subdirectory are replaced with the default system files of the Machine Class.

For example, if you have a client definition named PALE40 which has a SCSI device. The CONFIG.SYS file in C:\IBMLAN\RPL\MACHINES\PALE40 is changed to include the SCSI device driver. Then when you change the printer setting of PALE40 using the GUI, and save the new setting, the modified CONFIG.SYS is replaced with the default CONFIG.SYS.

If you need to change a specific client definition using the GUI and the CONFIG.SYS file for the client is modified, you have to back up the CONFIG.SYS file and restore the CONFIG.SYS after you change the client definition using the GUI. Or you have to edit and modify the CONFIG.SYS again after you change the client definition using the GUI.

---

## B.6 LAN Adapter Address Consideration

In the first phase of client IPL, WorkSpace On-Demand uses Universal Administered Address (UAA) as the source address on FIND frame to the server. This is not configurable, but you can configure each client to use Local Administered Address (LAA) after WorkSpace On-Demand client MPTS starts. This is done by manually by editing each client's PROTOCOL.INI file, which is located in d:\IBMLAN\RPL\MACHINES\<machine\_ name>\IBMCOM\ subdirectory.

```
[IBMTOK_nif]

DriverName = IBMTOK$
NETADDRESS = "400052003101"
MAXTRANSMITS = 6
RECVBUFS = 2
RECVBUFSIZE = 256
XMITBUFS = 2
```

*Figure 105. Network Adapter Address in the PROTOCOL.INI File*

---

**Note**

If your LAN adapter's NIF section does not have the NETADDRESS option, you can define the adapter address with the NETADDRESS option in the [netbeui\_nif] and [landd\_nif] sections.

---

## B.7 Undocumented Internal Public Application Parameters

There are some application parameters that are internal to WorkSpace On-Demand that have not been documented in the WorkSpace On-Demand Administrator's Guide. Two of these internal parameters are used by the applications that are described below. They are:

- WSOD\_LAUNCH\_MINIMIZED** This parameter is used by the Notes and the Access Feature applications to suppress the brief display of a CMD.EXE window during application startup.
- WSOD\_LAUNCH\_NODROP** This parameter is used by the Notes application to suppress dropping of network resources.

The reason for using WSOD\_LAUNCH\_NODROP for NOTES and other application are:

- When NOTES.EXE is launched by the WorkSpace On-Demand application launcher (APPSTART.EXE) the launcher sets up the environment, including all network resources specified by the application parameters defined for Notes.
- NOTES.EXE does some of its own environment setup, then starts ILNOTES.EXE and NOTES.EXE ends.
- Under normal circumstances, APPSTART.EXE detects that NOTES.EXE has ended and so it drops all network resources. However NOTES.EXE started ILNOTES.EXE, which continues to run and requires the same network resources assigned to NOTES.EXE.
- Under these circumstances, the use of the WSOD\_LAUNCH\_NODROP parameter prevents APPSTART.EXE from dropping the network resources assigned to NOTES.EXE. This means that these network resources are not dropped automatically even when ILNOTES.EXE ends. They will only be dropped when the system shuts down, or by some other means such as the user or a program.
- This parameter may be required by other applications that behave in the same way as NOTES.EXE and ILNOTES.EXE.

---

## B.8 Caution Required When Naming Public Applications

Caution is required when an administrator names a public application during creation of the public application. This is because the name given to a public application may cause a conflict with the application's .INI file (if any). The circumstances that can cause this conflict are as follows:

- When a public application is defined and assigned to a user, a file with the <public\_application\_name>.ini is created on the Domain Controller in the \BMLAN\DCDB\USERS\<user\_name>\ directory. If for example the administrator creates a public application NETSCAPE then the file \BMLAN\DCDB\USERS\<user\_name>\NETSCAPE.INI will be created on the Domain Controller when the NETSCAPE application is added to (or set up for) the user.
- The default user FIT file shipped with WorkSpace On-Demand that is stored on the Domain Controller in \BMLAN\DCDB\BB10USDU.FIT contains entries that direct the Netscape application to write all its user level files into \BMLAN\DCDB\USERS\<user\_name>. One of these entries is NETSCAPE.INI, thereby overwriting the WorkSpace On-Demand file of the same name.
- The README.APP file on the WorkSpace On-Demand CD contains the following:

### 6.0 SETTING UP NETSCAPE NAVIGATOR

---

To set up Netscape Navigator Version 2.02 for OS/2 as a public application, type the following from an OS/2 Window session:

```
NET APP NETSCAPE /APPDIR:Z:\NETSCAPE /COMMAND:NETSCAPE.EXE  
/REMARK:"Netscape Navigator" /TYPE:WSOD /INTERFACE:PM /ADD
```

*Figure 106. README.APP Set Up Information from WorkSpace On-Demand CD-ROM*

- These instructions will cause the file name conflict because the application's name is NETSCAPE.
- In order to avoid this conflict, make sure that the name the you assign to the public applications are not the same name as the application's .INI file. For example, call the application NETSCAP instead of NETSCAPE. Calling the application NETSCAP will not impact the user, because the application's name displayed on the requester is the Application Description (/Remark in the Command Line invocation).

---

## B.9 Public Application Command Parameter 119 Character Limitation

When defining an WorkSpace On-Demand application, there is a limitation of 119 characters for the Command parameter. This parameter defines both the program to be executed and the parameter(s) to be passed to the command at execution and together this string is limited to 119 characters.

---

## B.10 Adding Application Parameters to a Public Application

Application specific parameters added using the graphical user interface or the command line must be limited to environment variables and not for any DEVICE or RUN statements that may be required for an application. DEVICE and RUN statements required by an application must be added to the client's CONFIG.SYS file.

---

## B.11 Disabling of an IBM WorkSpace On-Demand Requester

The Administrator is able to disable a RIPL requester that has previously been enabled. If this requester is powered on and started, it will be necessary not only to shutdown the requester but also to power off the machine, before it can be disabled via either the GUI or the command NET RIPLMACH.

---

## B.12 Upgrade to TCP/IP for OS/2 Version 4.1

The level of TCP/IP support included in WorkSpace On-Demand is 4.0. During the product development cycle, TCP/IP 4.1 for OS/2 became generally available. You may wish to read up on the latest details of TCP/IP for OS/2 Version 4.1 on the Web at:

[http://antero.boulder.ibm.com/asd-bin/doc/en\\_us/tcpip/f-prod.htm](http://antero.boulder.ibm.com/asd-bin/doc/en_us/tcpip/f-prod.htm)

Following are some of the highlights of TCP/IP for OS/2 4.1 that benefit from WorkSpace On-Demand.

- 32-Bit, BSD 4.4-Compliant TCP/IP Stack
- Dynamic IP Client Enhancements
  - Detecting a laptop that was relocated while in hibernation mode
  - Releasing the DHCP address when shutting down (optional)
  - Reusing old lease information if a DHCP server cannot be found on the network (optional)
- Virtual Private Network and IP Security
- RSVP Daemon and API

- Application Enhancements:
  - LPR, LPRMON, LPQ, LPRM: The print server name and print queue name can now default to names received from a DHCP server.
  - LPR: Has a -cp command line option to convert text files using the specified code page before sending the files to an LPD server.
  - LPRPORTD Performance Enhancements
- Year-2000 Support

To be able to use TCP/IP for OS/2 4.1 on a WorkSpace On-Demand client, you need to copy the files from TCP/IP for OS/2 directory to WorkSpace On-Demand directory manually. The overall of the procedure is as follows.

1. Install TCP/IP for OS/2 4.1 on a OS/2 Warp Version 4 machine.
2. Copy TCP/IP for OS/2 files to WorkSpace On-Demand directory.

**Note**

This information is based on working with TCP/IP for OS/2 4.1 and early drivers for WorkSpace On-Demand. Most of the new functions were not tested, and this process is not verified by the development. To test the installation, we used PING and tested the following functions:

- DHCP address allocation to WorkSpace On-Demand client
- Dynamic name registration to DDNS
- Reusing old lease information when DHCP server cannot be found
- Static IP configuration

### **B.12.1 Install TCP/IP for OS/2 4.1**

First you need to install TCP/IP for OS/2 4.1 so that we can copy the files to WorkSpace On-Demand directory. It is not necessary to install TCP/IP for OS/2 4.1 to the WorkSpace On-Demand server. You can install TCP/IP for OS/2 4.1 to another machine where OS/2 Warp Version 4 is installed. The prerequisite for TCP/IP for OS/2 4.1 is Java for OS/2 1.1.1 and Netscape for OS/2 2.02.

#### Note

It is recommended that you install TCP/IP for OS/2 4.1 onto a machine that has not had TCP/IP for OS/2 previously installed on it. This is because in the next step you may copy some older version TCP/IP configuration information from the machine on which you are reinstalling to the WorkSpace On-Demand directory structure.

When TCP/IP for OS/2 4.1 is installed, a new version of MPTS is also installed. As a result, the following directories are upgraded.

- d:IBMCOM
- d:MPTN
- d:TCPIP

(d is the drive letter where the each component is installed.)

### B.12.2 Copy the TCP/IP files to WorkSpace On-Demand Directory

Copy the TCP/IP for OS/2 files into appropriate WorkSpace On-Demand directory. You can issue the following XCOPY commands to copy these files:

- xcopy d:\IBMCOM d:\IBMLAN\RPL\BB10.US\IBMCOM /s /e
- xcopy d:\MPTN d:\IBMLAN\RPL\BB10.US\MPTN /s /e
- xcopy d:\TCPIP d:\IBMLAN\RPL\BB10.US\TCPIP /s /e
- xcopy d:\IBMCOM d:\IBMLAN\RPL\IBMCOM /s /e
- xcopy d:\MPTN d:\IBMLAN\RPL\MPTN /s /e
- copy d:\MPTN\BIN\MPTSTART.COM d:\IBMLAN\RPLUSER\BB10:US\MPTN\BIN\MPTSTART.COM
- copy d:\MPTN\ETC\\*. \* d:\IBMLAN\RPLUSER\BB10:US\MPTN\ETC\\*. \*
- copy d:\TCPIP\BIN\TCPSTART.COM d:\IBMLAN\RPLUSER\BB10:US\TCPIP\BIN\TCPSTART.COM

(d is the drive letter where the each component is installed.)

Following the above instruction the client should work with TCP/IP for OS/2 4.1.

#### Note

For our test environment, to test DDNS client support, we used STARTUP.COM to launch the DDNS configuration program. See B.13, "Consideration for TCP/IP Configuration Using DDNS" on page 267 for the details.



---

## B.13 Consideration for TCP/IP Configuration Using DDNS

DDNS client support for TCP/IP for OS/2 Warp is performed by DDNSCFG.EXE. In a regular TCP/IP for OS/2 machine, this program is launched from the STARTUP folder, but there is no STARTUP folder in WorkSpace On-Demand client. In our testing, we found that DDNSCFG.EXE does not come up when WorkSpace On-Demand client is started even if you configured the client to use DDNS through the administrator's GUI. Following are two solutions that we found to start DDNSCFG.EXE automatically at startup of the WorkSpace On-Demand client.

- Using STARTUP.CMD

You can use STARTUP.CMD to start the DDNS client support for TCP/IP for WorkSpace On-Demand as follows: For this example, the client name is BB350, and the FIT file name is BB350.FIT.

1. Create STARTUP.CMD.

Locate STARTUP.CMD in d:\BMLANRPLMACHINES<machine\_name> directory. Add the following statement to the STARTUP.CMD file.

```
start Z:MPTNBINDDNSCFG.EXE
```

### Note

If this is the first entry in the STARTUP.CMD file, you need to add an exit command to the file as well.

2. Edit the FIT file.

The FIT file is located in d:\BMLANRPLFITS directory, and it is named <machine\_name>.FIT file. Add a statement for STARTUP.CMD as shown in Figure 107.

```
AUSSRVPLFILES
; The first line of this file MUST be UNC name

; Per-workstation read-only configuration files.
Z:\CONFIG.SYS      MACHINES\BB350\CONFIG.SYS
Z:\IBMLAN\IBMLAN.INI  MACHINES\BB350\IBMLAN\IBMLAN.INI
Z:\STARTUP.CMD      MACHINES\BB350\STARTUP.CMD

; These OS/2 files must be writeable.
Z:\OS2\BOOT\ARCHBASE.*  \\AUSSRV\WRKFILES\BB350\OS2
.
.
```

Figure 107. BB350.FIT

- Using the logon script

You can use add instructions to the logon script to start the DDNS client support for TCP/IP for WorkSpace On-Demand as follows. The name of logon script can be whatever you like. In this example, the name of the logon script file is PROFILE.CMD.

1. Edit or create the WorkSpace On-Demand client logon script file.

The statements shown in Figure 108 should be included in your logon script file.

```
start Z:MPTNBINDDNSCFG.EXE
exit
```

Figure 108. Sample Logon Script File - PROFILE.CMD

You need to place this script file in the IBMLAN directory structure.

**Note**

The default path for logon script is d:IBMLANREPLIMPORTSCRIPTS. This default path is specified in the IBMLAN.INI file in the SCRIPTS= part of the [net\logon] section of the IBMLAN.INI file.

2. Specify the logon script file for a user

To specify the logon script file name for a user, issue the following command:

```
net user <user_name> /script:<path>PROFILE.CMD
```

The <path> is relative from the path specified in the SCRIPTS= statement in IBMLAN.INI file. The default path to the logon script is d:IBMLANREPLIMPORTSCRIPTS. You only need to input additional directories to this default path information if you have created additional directories for users. If you have placed the script file in the directory specified in the SCRIPTS= parameter, you do not need to specify any additional path information.

---

## Appendix C. Interesting Web Pages to Read

This Appendix presents some interesting material related to WorkSpace On-Demand which could be found on the Internet at the time this book was published. There is no guarantee that these references will be available for any length of time.

---

### C.1 WorkSpace On-Demand Home Page

This URL is the WorkSpace On-Demand home page provided by IBM WorkSpace On-Demand development.

- WorkSpace On-Demand Home Page

<http://www.software.ibm.com/os/warp/workspace/>

---

### C.2 WorkSpace On-Demand SAPR Guide

This URL is for the IBM Personal Solutions Systems Center, which provides Product Assurance Guides and other technical documentation including white papers.

- WorkSpace On-Demand Product Assurance Checklist

<http://pssc.sl.dfw.ibm.com>

---

### C.3 Understanding Network-Computing Fundamentals

The following pages provide information to help you understand the basic and background of WorkSpace On-Demand and network-computing.

- Good Overview for remote boot in corporate networks

[http://www.incom.de/facts/whitepaper1\\_en.shtml](http://www.incom.de/facts/whitepaper1_en.shtml)

- The NetPC and network computer explained

[http://www.incom.de/facts/netpcexplain\\_en.shtml](http://www.incom.de/facts/netpcexplain_en.shtml)

- Determining the total cost of NC ownership: How to start

<http://www.ncworldmag.com/ncworld/ncw-08-1997/ncw-08-tco.html>

---

### C.4 Web Pages That Referred to Performance

These pages provide performance information for WorkSpace On-Demand, OS/2 Warp Server and for their competitors.

- WorkSpace On-Demand Remote-IPL performance

<http://www.software.ibm.com/os/warp/workspace/whitepaper/wsodperf.html>

- IBM OS/2 Warp Server and Microsoft Windows NT Server 4.0 performance Comparison

<http://www.software.ibm.com/os/warp/performance/wslcomp.htm>

- Top Tips to improve OS/2 Warp Version 4 performance

<http://www.software.ibm.com/os/warp/performance/toptips.htm>

---

## **C.5 WorkSpace On-Demand Web Pages for Related Products**

Web pages that provide related product information for WorkSpace On-Demand and OS/2 Warp Server are listed below.

- The IBM Software Choice home page

<http://service.software.ibm.com/asd-bin/doc/index.htm>

- IBM Network Station

<http://www.internet.ibm.com/computers/networkstation/>

- Wake on LAN--An Administrator's Perspective

<http://www.networking.ibm.com/tra/trawolwp.html>

---

## Appendix D. Network Adapter Considerations and Work-Arounds

This Appendix discusses adding new network adapters to your WorkSpace On-Demand environment. Whenever you obtain new network adapters for your environment, it is possible that the new adapters are fundamentally different than the adapters you've implemented so far. If that is the case, there are a few steps you need to take before these network adapters can be integrated into your WorkSpace On-Demand environment.

When manufacturers deliver new or improved network adapters, they also ship updated versions of their drivers with the hardware. The important drivers you must obtain for WorkSpace On-Demand are the NDIS-compliant (DOS) and NDIS-compliant (OS/2) drivers. Check the documentation provided by the manufacturer for the names and locations of these drivers; they should be included on the diskette provided by the adapter manufacturer.

---

### D.1 IBM Turbo Token-Ring Card Issues

When attempting to utilize the Turbo version of the IBM shared-RAM ISA 16/4 Token-Ring Adapters (including the Wake On LAN version, which is IBM Auto Wake Token-Ring ISA Adapter), we encountered a failure during the first, or DOS phase, of initialization. The second phase of the RIPL process, using the NDIS-compliant OS/2 driver, completes successfully. The work-around that we used was to substitute the non-NDIS, DXM-drivers for the NDIS-compliant DOS drivers, leaving the NDIS-compliant OS/2 drivers in place.

The problem with using the Turbo cards is that the IBMTOK.DOS that is shipped with WorkSpace On-Demand does not work with the Turbo card. The solution is to get the latest device driver diskettes for the Turbo card via the Web and install the updated IBMTOK.DOS and LT2.MSG files. You can do this by the following:

1. Connect the IBM Networking Web site at <http://www.networking.ibm.com>
2. Select the Hardware category.
3. Select the Token Ring category.
4. Select the Auto/Turbo 16/4 Adapters category
5. Select the Downloads option.
6. Select the Turbo 16/4 Product.

7. Download disk 2, TRI164T2.EXE
8. Run TRI164T2 to create a diskette image.
9. Copy IBMTOK.DOS and LT2.MSG from the DOS directory on the diskette to the d:\IBMLAN\DOSLAN\LSP\DOS directory.

**Note**

At the time we did the research on getting the IBM Auto Wake Token-Ring ISA Adapter to remotely boot a WorkSpace On-Demand client, we had not found the solution for getting the card to work correctly listed above. These steps have been placed in this redbook for those cases where you are having problems booting a particular adapter and you may want to utilize this approach in an attempt to enable your adapter.

### D.1.1 Step One: Copy the Updated Device Drivers to WorkSpace

On-Demand Server

Figure Figure 109 shows the instructions in the adapter documentation for copying the updated non-NDIS DOS device drivers to the the appropriate directory on the WorkSpace On-Demand server:

```
COPY A:\LSPDXMAOMOD.SYS C:\IBMLANDOSLANLSP
COPY A:\LSP\DXMCOMOD.SYS C:\IBMLAN\DOSLAN\LSP
```

*Figure 109. Non-NDIS DOS Device Drivers. Copying Updated Device Drivers from Media to WorkSpace On-Demand server.*

### D.1.2 Step Two: Update RPL.MAP - ADD OTKTTR

Edit C:\IBMLANRPLRPL.MAP file and add a server record for a new adapter. Figure Figure 110 shows the new record added to RPL.MAP. Note that, for the purpose of showing the information in this figure, the record has been split into two lines here, but would be only one line in your RPL.MAP file.

```
yyyyyyyyyyyyy bb1usttr.cnf 3 10 N ~ BLUEBIRD~ 1.0~ IBM~ TURBO~ TOKEN~ RING~
(line continued) NON-NDIS~ ~,,,~ R_BB10_US_OTKTTR~ ~
```

*Figure 110. RPL.MAP. New Entry for IBM Non-NDIS Token-Ring Adapter.*

### D.1.3 Step Three: Edit Boot Block Definition File (.CNF)

We need a new CNF file for IBM non-NDIS Token-Ring adapter. First, we copy C:\IBMLANRPLBB1USNTR.CNF to C:\IBMLANRPLBB1USTTR.CNF. Edit C:\IBMLANRPLBB1USNTR.CNF to change the initial device driver statements to the appropriate DXM support, and remove the NDIS device driver-related statements. Figure 111 shows in bold the lines that have added and changed in BB1USTTR.CNF.

```
; OS/2 Boot Block Configuration
; (IBM Token-ring non-NDIS Adapter)
RPL DOS\RPLBOOT.SYS
DAT DOS\MFSD20.SYS
ORG 1000H
LDR BB10.US\OS2LDR ~ OS2LDR UFSD.SYS MFSD20.SYS
DAT DOS\UFSD.SYS
DAT C:\IBMLAN\DOSLAN\LSP\DXM.MSG
;**NETBIOS and IEEE 802.2 NON NDIS*****
DRV C:\IBMLAN\DOSLAN\LSP\DXMT0MOD.SYS
PBA=0~ S=12~ ST=12~ C=14~ O=N ~ ~
DRV C:\IBMLAN\DOSLAN\LSP\DXMC0MOD.SYS ~ 10 ~
DRV C:\IBMLAN\DOSLAN\LSP\DXMA0MOD.SYS 001 ~ ~
;**NETBIOS and IEEE 802.2*****
```

Figure 111. BB1USTTR.CNF. Modifications for IBM Non-NDIS Token-Ring Adapter.

### D.1.4 Step Four: Edit NDISDD.PRO

Edit C:\IBMLANRPLNDISDD.PRO to point the OTKTTR to the IBMDXM.NIF file. Figure 112 shows in bold the lines that have added and changed in NDISDD.PRO.

```
;NDIS device driver profile records are composed of 4 fields.
;They are described below.
;Field 1 This field specifies the DOS NDIS device driver name.
;Field 2 This field specifies the subdirectory name that
; contains the configuration file(s) needed to remote
; boot the RIPL client.
; d:\IBMLAN\RPL\IBMCOM\xxx\ contains the configuration
; files needed to RIPL OS/2 clients.
; d:\IBMLAN\RPL\DOS\xxx\ contains the configuration file(s)
; needed to RIPL both DOS and OS/2 clients.
; d - the drive on which the RIPL component has been installed
; xxx - the text located in the second field of a given line
;Field 3 NIF file name for OS/2 device driver
;Field 4 CNF file name id reference for server record id
IBMTOK.DOS TOKENRG IBMDXM.NIF OTKTTR
```

Figure 112. NDISDD.PRO. Modifications for IBM Non-NDIS Token-Ring Adapter.

### D.1.5 Step Five: Edit IBMDXM.NIF

Editing C:\IBMLANRPLBB10\USIBMCOMM\ACS\IBMDXM.NIF is necessary in order to have the correct new name available in the LAN Server GUI. Remember that you select a Network Card when creating a new WorkSpace On-Demand client. All other configuration information concerning the OS/2 device driver was left untouched.

```
•IBMTOK"
Type = NDIS
Title = "IBM T-R Adapter (NON-NDIS Bootup)"
Version = 3.2
DriverName = IBMTOK$
Xports = NETBEUI LANDD
Copyfile = LT2.MSG, LT2H.MSG
```

Figure 113. IBMDXM.NIF. Modifications for IBM Non-NDIS Token-Ring Adapter.

---

## D.2 DOS Network Driver Errors

This section contains information about errors we encountered when attempting to utilize the IBM PCI Wake On LAN Token-Ring Adapter and how we resolved the errors.

### D.2.1 IBM Auto Wake Token-Ring ISA Adapter

This is the Wake On LAN version of the Turbo Token-Ring ISA Adapter. In working with the IBM Auto Wake Token-Ring ISA Adapter, we found that we needed to make the same modifications that were done for the IBM Turbo 16/4 Token-Ring Adapter described in D.1, "IBM Turbo Token-Ring Card Issues" on page 271.

### D.2.2 IBM PCI Wake On LAN Token-Ring Adapter

We started by using a copy of standard token-ring adapter's CNF file and NIF file. For this example, we use BB1USTRP.CNF as the CNF file name and IBMTRP.NIF as the NIF file name for Wake On LAN adapter. Note that this information only applies to the IBM PCI Wake On LAN Token-Ring Adapter.

1. DXMJ0 41E: The protocol manager could not be opened.

This error comes from IBMTRP.DOS driver initialization failure. IBMTRP.DOS, which is the DOS NDIS driver for the Wake On LAN Adapter, needs the appropriate parameter with it; and LA1.MSG, which is the message file for this driver, needs to be included in this CNF file. We have to change BB1USTRP.CNF to activate these files as shown in Figure 114 on page 275.



```

; OS/2 Boot Block Configuration
; (IBM Wake on LAN Token-Ring PCI Adapter)
RPL DOS\RPLBOOT.SYS
DAT DOS\MFSD20.SYS
ORG 1000H
LDR BB10.US\OS2LDR ~ OS2LDR UFSD.SYS MFSD20.SYS
DAT DOS\UFSD.SYS
DAT DOS\IBMTRP\OS2\PROTOCOL.INI
DAT C:\IBMLAN\DOSLAN\LSP\DXM.MSG
DAT C:\IBMLAN\DOSLAN\LSP\DOS\LA1.MSG
EXE C:\IBMLAN\DOSLAN\LSP\NETBIND.COM ~ ~ ~
; **NETBIOS and IEEE 802.2**
; DRV C:\IBMLAN\DOSLAN\LSP\DXMT0MOD.SYS PBA=0~S=12~ST=12~C=14~O=N ~ ~
; DRV C:\IBMLAN\DOSLAN\LSP\DXME0MOD.SYS ~ 10 ~
; **NETBIOS and IEEE 802.2**
;
; **NETBIOS**
DRV C:\IBMLAN\DOSLAN\LSP\DXMJ0MOD.SYS ~ 6 ~
; **NETBIOS**
DRV C:\IBMLAN\DOSLAN\LSP\DXMA0MOD.SYS 001 ~ ~
DRV C:\IBMLAN\DOSLAN\LSP\DOS\IBMTRP.DOS ~ 16 ~
DRV C:\IBMLAN\DOSLAN\LSP\PROTMAN.DOS /I: ~ ~

```

Figure 114. BB1USTRP.CNF. Modifications for IBM PCI Wake On LAN Token-Ring Adapter Support.

## 2. . NET2501 Error: The specified program needs extra memory KB: 5K

The combination of IBMTRP.DOS, which is NDIS MAC driver, and DXMJ0MOD.SYS, which is a NetBIOS protocol driver with default parameters, causes this error that indicates NetBEUI exceeds memory limit. We have to edit the NetBEUI parameters in the PROTOCOL.INI file and minimize the resource for NetBEUI. Figure Figure 115 on page 276 shows the changes in bold to the PROTOCOL.INI.

```

;#####
;\IBMLAN\RPL\DOS\TOKENTRP\PROTOCOL.INI
;#####
[PROTMAN_MOD]
    DriverName = PROTMAN$
[DXMJO_MOD]
    DriverName = NETBEUI$
    Bindings = IBMTRP_MOD
    PiggyBackAcks = 0
    Sessions = 6
    NCBS = 6
    Names = 6
    Stacksize = 512
    Maxdatarcv = 512
[DXMEO_MOD]
    DriverName = DXMEO$
    Bindings = IBMTRP_MOD
[IBMTRP_MOD]
; Warning: If you change parameters that affect the amount of RAM
; this driver needs at run time (such as RcvBuffSize), you will
; need to change the *TRP.cnf files space allocated with the driver
    DriverName = IBMTRP$
    EnableTxEOFInt = YES
    MaxTransmits = 2
    MaxTxFrameSize = 2048
    MulticastNum = 1
    MinRcvBufs = 5
    RcvBuffSize = 1120
    DataRate = AUTO
    FullDuplex = YES
;NetAddress defaults to the adapter Universal burned-in address
;(a locally-administered network address may be specified here)
;    NetAddress = "xxxxxxxxxx"
;

```

Figure 115. *PROTOCOL.INI. Modifications to Minimize the Resource Usage for NetBEUI.*

## Appendix E. RIPL Workstation Error/Status Messages

When a Remote IPL workstation is started a number of indicators are displayed on the screen. This Appendix describes the indicators and their descriptions. For further, in-depth information, see the *IBM Token-Ring Network Remote Program Load Version 1.0*, SK2T-0333.

Table 21 describes the various messages displayed on a token-ring RIPL workstation's monitor, along with their explanations or descriptions. These messages are specifically for IBM equipment; however, most of them are commonly used in the industry.

Table 21 (Page 1 of 3). Description of the Token-Ring RPL Messages	
Message	Explanation
ET-00:00:00	Elapsed Time. A continuously updated field indicating the elapsed time since the RPL Feature gained control.
ID-nnn	<p>Identification. An indication of which adapter is using the RPL function, where nnn is the adapter identifier.</p> <ul style="list-style-type: none"><li>• A value of 249 indicates an IBM LANStreamer MC 32 Adapter or an IBM LANStreamer MC 16 Adapter.</li></ul> <p>There will be a two-digit number following the value of 249. These two digits refer to the slot number of the adapter.</p> <ul style="list-style-type: none"><li>• A value of 16n identifies all other Token-Ring adapters where:<ul style="list-style-type: none"><li>– 166 indicates the primary adapter.</li><li>– 167 indicates the alternate adapter.</li></ul></li></ul>
CE-0085	Configuration Error. A configuration error was found during the computer's Power-On Self Test (POST). If this message is displayed, you will continue to see this message every time the computer is powered on, unless you take action to correct your computer's configuration.
BU-0000	Bring-Up. This field is displayed as X'0000' if the adapter has been successfully initialized and opened. If not, a code other than X'0000' is displayed, and the field is highlighted see the adapter manual for more information on the error.
AA-10005 A000567	Adapter Address. The permanently encoded address of the token-ring adapter in the Requesting Device. This address is always 12 hexadecimal characters (6 bytes) long.
AL-00 0000 B001253	Adapter Level. The Engineering Change (EC) level of the code in the RPL Feature.
BL-00112233	BIOS Level (Module Level). The Engineering Change (EC) level of the code in the RPL Feature.
MM-CC00 02	Memory (Read-Only Memory). The first 4 digits are the hexadecimal ROM address set on the adapter. The last 2 digits are the interrupt level set on the adapter.

<i>Table 21 (Page 2 of 3). Description of the Token-Ring RPL Messages</i>	
<b>Message</b>	<b>Explanation</b>
IO-8000	I/O Address. This field specifies the base Micro Channel address in the memory map.
SR-D800 08	Shared RAM. The first 4 digits are the hexadecimal shared RAM address that the adapter uses. The last 2 digits are the shared RAM size (in decimal). "08" indicates 8 kilobytes (8 KB) of memory, "16" indicates 16 KB of memory, "32" indicates 32 KB of memory, and "64" indicates 64 KB of memory.  This message is not valid for the LANStreamer family of adapters
OP-0000 04	Open Return Code. The first 4 digits are X'0000', and the last two digits identify the adapter data rate, if the adapter has been successfully opened and attached to the ring. If not, a code other than X'0000' is displayed, and the error field is highlighted, flashing or in reverse video.
RQ-0001	Request Count (FIND Frame Count). The number (in hexadecimal) of FIND frames that have been transmitted. An excessive request count (greater than 10, for instance) indicates that the server is either not present, is congested, or the workstation address has not been properly entered at the server.
SF-0001	SEND.FILE.REQUEST Frame Count. The number of SEND.FILE.REQUEST frames that have been transmitted. An excessive SEND.FILE.REQUEST frame count (greater than 10, for instance) indicates that the server is not responding after having been found.
SN-0023	File Response Sequence Number. This value is displayed when the server has responded to the SEND.FILE.REQUEST. It indicates how many times valid FILE.DATA.RESPONSE frames have been received.
DS-0080	DLC Status. The message provided by the adapter if an unexpected Data Link Control (DLC) condition occurs.
RS-0040	Ring Status. This field displays a code indicating the status of the ring. The field will be highlighted if the operation cannot continue; it will not be highlighted if processing can continue.
PC-4020	IBM Personal Computer or IBM PS/2 computer error. This field displays an error code indicating that the adapter has difficulty in functioning with the computer. In most cases, the screen will be frozen, and this field will be highlighted because the adapter cannot continue. For more information, see the adapter and/or workstation hardware manual.
AC-0040 0000 0000 0000	Adapter Check. The adapter has detected an internal error and cannot continue. Restart your computer. If the error persists, have the hardware checked or replaced.

<i>Table 21 (Page 3 of 3). Description of the Token-Ring RPL Messages</i>	
<b>Message</b>	<b>Explanation</b>
AE-nnn XX-0011	<p>Adapter Error. The adapter in the workstation could not establish communication with the server. The adapter identifier is nnn, where nnn can have one of the following values:</p> <ul style="list-style-type: none"> <li>• A value of 249 indicates an IBM LANStreamer MC 32 Adapter or IBM LANStreamer MC 16 Adapter.</li> <li>• A value of 16n identifies all other token-ring adapters where: <ul style="list-style-type: none"> <li>– 166 indicates the primary adapter.</li> <li>– 167 indicates the alternate adapter.</li> </ul> </li> </ul> <p>The reason for this error is indicated in the XX message to the right of AE-nnn XX may be either BU or OP. The BU and OP messages were described previously.</p>

Table 22 describes the messages that are unique to Ethernet, along with their explanations or descriptions. These messages are specifically for IBM equipment; however, most of them are commonly used in the industry:

<i>Table 22. Description of the Ethernet-Specific Messages</i>	
<b>Message</b>	<b>Explanation</b>
ID-0064	Identity. 064 indicates the IBM Ethernet Adapter.
VR-150	Indicates the POST/RPL version level.
BI 0280 03	Base I/O. The first four digits are the hexadecimal base I/O address set on the adapter. The last two digits are the interrupt level set on the adapter.
ES-0000	This field displays a code indicating the status of Ethernet. Refer to the adapter manual for a description of each of the status codes.



---

## Appendix F. Tools for Modifying and Managing CONFIG.SYS and INI Files in Machine Classes

This Appendix contains the source for the REXX tools that were used in the building of Machine Class examples in Chapter 8, “WorkSpace On-Demand Machine Classes” on page 149. These tools are provided to assist you in creating the customized INI files needed to support a new Machine Class.

Because OS2.INI files contain binary data, they can be very difficult to work with. The tools provided in this Appendix enable you to create a functional OS2.INI file for your new Machine Class.

---

### F.1 Creating a Delta INI File

Use INICREATEDELTA4 with OS2vga.ini and OS2svga.ini files to create a DELTA.INI file. This REXX procedure calls the CopyApplication.CMD. The source for INICREATEDELTA4 is shown in Figure 116.

```
/* */
call RxFuncAdd 'SysLoadFuncs', 'RexxUtil', 'SysLoadFuncs'
call SysLoadFuncs

parse arg source1 source2 delta .
if words(delta)==0 then do
    say "INICREATEDELTA4 <SOURCE1.INI> <SOURCE2.INI> <DELTA.INI>"
    say "This procedure assumes that source1 is a subset of source2"
    exit
end

call SysIni source1, 'All:', 's1apps.'
if Result == 'ERROR:' then do
    say "Error: Could not read applications from" source1"."
    exit
end

call SysIni source2, 'All:', 's2apps.'
if Result == 'ERROR:' then do
    say "Error: Could not read applications from" source2"."
    exit
end
```

Figure 116 (Part 1 of 2). INICREATEDELTA4.CMD Create Delta INI File REXX Utility

```

do i2=1 to s2apps.0
  /* Check if Application is new */
  newApplication=1
  do i1=1 to slapps.0
    if s2apps.i2==slapps.i1 then do
      newApplication=0; i1=slapps.0
    end
  end
  if newApplication==1 then do
    say s2apps.i2"::: -> new application found!"
    "@start /c /min CopyApplication" source2 delta s2apps.i2
  end
  else do
    call SysIni source1, s2apps.i2, 'All:', 's1keys.'
    call SysIni source2, s2apps.i2, 'All:', 's2keys.'
    do j2=1 to s2keys.0
      /* Check if Key is new */
      newKey=1
      do j1=1 to s1keys.0
        if s2keys.j2==s1keys.j1 then do
          newKey=0; j1=s1keys.0
        end
      end
      if newKey==1 then do
        say s2apps.i2":::"s2keys.j2 "-> new key found."
        "@start /c /min CopyApplication" source2 delta s2apps.i2
        say RIGHT("Further scan of Keys in this Application skipped.",79)
        j2=s2keys.0
      end
    end
    else do
      /* Check if Key Value is different */
      call SysIni source1, s2apps.i2, s1keys.j2
      keyvalue1=result
      call SysIni source2, s2apps.i2, s1keys.j2
      keyvalue2=result
      if keyvalue1<>keyvalue2 then do
        say s2apps.i2":::"s2keys.j2 "-> modified key found."
        "@start /c /min CopyApplication" source2 delta s2apps.i2
        say RIGHT("Further scan of Keys in this Application skipped.",79)
        j2=s2keys.0
      end
    end
  end
end
end
end
end
exit

```

Figure 116 (Part 2 of 2). INICREATEDELTA4.CMD Create Delta INI File REXX Utility

CopyAppication is called by INICREATEDELTA4. The source for this utility is shown in Figure 117 on page 283.



```

/* */
call RxFuncAdd 'SysLoadFuncs', 'RexxUtil', 'SysLoadFuncs'
call SysLoadFuncs

  parse arg source target application
  call SysIni source, application, 'All:', 'Key'
  if Result \= 'ERROR:' then
    do j=1 to key.0
      call SysIni source, application, key.j
      keyvalue.j=Result
    end
    do j=1 to key.0
      call SysIni target, application, key.j, keyvalue.j
    end
  say RIGHT("Application has been copied.",79," ");
  return

```

Figure 117. CopyApplication.CMD Copy Application REXX Utility

## F.2 Merging the Target and Delta INI Files

Run INIMERGE with your TARGET.INI and your DELTA.INI file. The delta gets merged into your target. The source for INIMERGE is shown in Figure 118.

```

/* */
call RxFuncAdd 'SysLoadFuncs', 'RexxUtil', 'SysLoadFuncs'
call SysLoadFuncs

  parse arg target delta .
  if words(delta)==0 then do
    say "INIMERGE <TARGET.INI> <DELTA.INI>"
    exit
  end

  call SysIni delta, 'All:', 'apps.'
  if Result == 'ERROR:' then do
    say "Error: Could not read applications from" delta"."
    exit
  end

```

Figure 118 (Part 1 of 2). INIMERGE.CMD Merge INI File REXX Utility

```

do i=1 to apps.0
  call SysIni delta, apps.i, 'All:', 'keys.'
  do j=1 to keys.0
    say apps.i":::"keys.j "..."
    call SysIni delta, apps.i, keys.j
    call SysIni target, apps.i, keys.j, result
  end
end
end

```

Figure 118 (Part 2 of 2). INIMERGE.CMD Merge INI File REXX Utility

### F.3 Copying an Application from One INI File to Another

INICOPYAPPLICATION is a tool to copy just one application's INI information from one .INI file to another. The source for INICOPYAPPLICATION is shown in Figure 119.

```

/* */
call RxFuncAdd 'SysLoadFuncs', 'RexxUtil', 'SysLoadFuncs'
call SysLoadFuncs

parse arg source target ApplicationName .
if words(ApplicationName)=0 then do
  say "INICOPYAPPLICATION <SOURCE.INI> <TARGET.INI> <applicationname>"
  exit
end /* do */

call SysIni SOURCE, 'All:', 'Apps.'
if Result \= 'ERROR:' then
do i = 1 to Apps.0
  if Apps.i==ApplicationName then do
    call SysIni source, Apps.i, 'All:', 'Keys'
    if Result \= 'ERROR:' then
      do j=1 to Keys.0
        val = SysIni(source, Apps.i, Keys.j)
        call SysIni TARGET, Apps.i, Keys.j, val
      end
    end
  end
end
end

```

Figure 119. INICOPYAPPLICATION.CMD Copy INI File REXX Utility

---

## F.4 Comparing Two CONFIG.SYS Files

COMPCFS.COMD is a sample REXX utility that you can use to compare two different CONFIG.SYS files. You can use this tool to note the changes that were made to a CONFIG.SYS file when installing support for an adapter or application. The source for COMPCFG is shown in Figure 120.

```

/*****
/* Name      : COMPCFG.COMD
/* Function   : Utility to compare two CONFIG.SYS files
/*
/* Parameters : FILE1 FILE2 OUTPUTFILE
/*
*****/

parse arg file1 file2 outfile .
if words(outfile)=0 then do
  say "COMPCFG <CONFIG1> <CONFIG2> <Delta File>"
  say "For example: COMPCFG C:\CONFIG.SYS C:\CONFIG.001 DELTA.002"
  say "This procedure compares <CONFIG1> and <CONFIG2> and"
  say " produces the delta file <Delta File>."
  say "The delta file contains the differences between the two files"
  exit
end /* do */

/*****
/* Compare the selected files
*****/

      /* INITIALIZE FILE IMAGE BUFFERS AND OUTPUT BUFFER */

CALL XINIT

      /* STANDARD CONFIGURATION FILES COMPARISON ROUTINE */
SAY Starting Comparison

CALL LINEOUTX "Differences between the two files where :"
CALL LINEOUTX "      1 = "||file1
CALL LINEOUTX "      2 = "||file2
CALL LINEOUTX LEFT('t',74,'t')
```

Figure 120 (Part 1 of 12). COMPCFS.COMD Compare Two CONFIG.SYS Files REXX Utility

```

SAY "checking first file"
DO WHILE rec1 \> words(str1)      /* Perform sequential read to */
    data = XLINEIN1()              /* end of second file      */

    PARSE UPPER VAR data string "=" rest
    IF string = "DEVICE" | string = "BASEDEV" | string = "RUN" |,
        string = "DEVINFO" | rest = "" THEN
    DO
        search = data
        IF string = "DEVICE" | string = "RUN" THEN
        DO
            PARSE UPPER VAR search x ' ' y
            search = x
            DO UNTIL y = ""
                PARSE UPPER VAR search x '\ ' y
                IF y <> "" THEN
                    search = y
            END
            search = '\ ' || search
        END
        ELSE
            search = string || '='
/*retc = SysFileSearch(search,file2,"found")*/
CALL SYSFILESEARCH1
IF found.0 = 0 THEN
DO
    line = "1 " || data
    CALL LINEOUTX line
    line = "2 "
    CALL LINEOUTX line
    line = ""
END
ELSE
    SELECT
        WHEN string = "IFS" THEN
        DO
            IF found.0 = 1 THEN
            DO
                PARSE UPPER VAR data x ' ' w
                PARSE UPPER VAR found.1 y ' ' z

```

Figure 120 (Part 2 of 12). COMPCFRS.CMD Compare Two CONFIG.SYS Files REXX Utility

```

IF x <> y THEN
DO
    line = "1 "||data
    CALL LINEOUTX line
    line = "2 "
    CALL LINEOUTX line
END
ELSE
IF TRANSLATE(data) <> TRANSLATE(found.1) THEN
DO
    line = "1 "||data
    CALL LINEOUTX line
    line = "2 "||found.1
    CALL LINEOUTX line
END
END
IF found.0 > 1 THEN
DO
    PARSE UPPER VAR data x ' ' w
    i = 0
    DO UNTIL x = y | i > found.0
        i = i + 1
        PARSE UPPER VAR found.i y ' ' z
    END
    IF i > found.0 THEN
    DO
        line = "1 "||data
        CALL LINEOUTX line
        line = "2 "
        CALL LINEOUTX line
        line = ""
    END
    IF x = y & TRANSLATE(data) <> TRANSLATE(found.i) THEN
    DO
        line = "1 "||data
        CALL LINEOUTX line
        line = "2 "||found.i
        CALL LINEOUTX line
        line = ""
    END
    END
    CALL LINEOUTX " "
END
WHEN string = "LIBPATH" THEN
DO
    CALL LINEOUTX " "
    pos = 9
    IF found.0 = 1 THEN
        CALL EQUAL_PATH found.1,data,pos
    CALL LINEOUTX LEFT('t',74,'t')
END

```

Figure 120 (Part 3 of 12). COMPCFRS.CMD Compare Two CONFIG.SYS Files REXX Utility

```

WHEN string = "SET PATH" THEN
DO
    CALL LINEOUTX "  "
    pos = 10
    IF found.0 = 1 THEN
        CALL EQUAL_PATH found.1,data,pos
    CALL LINEOUTX LEFT('t',74,'t')
END
WHEN string = "SET DPATH" THEN
DO
    CALL LINEOUTX "  "
    pos = 11
    IF found.0 = 1 THEN
        CALL EQUAL_PATH found.1,data,pos
    CALL LINEOUTX LEFT('t',74,'t')
END
WHEN string = "SHELL=" THEN
DO
    DO i = 1 UNTIL x = string | i > found.0
        PARSE UPPER VAR found.i x '=' y
        x = x||'='
    END
    IF i > found.0 THEN
    DO
        line = "1 "||data
        CALL LINEOUTX line
        line = "2 "
        CALL LINEOUTX line
    END
END
OTHERWISE
DO
    IF found.0 = 1 & TRANSLATE(data) <> TRANSLATE(found.1) THEN
    DO
        line = "1 "||data
        CALL LINEOUTX line
        line = "2 "||found.1
        CALL LINEOUTX line
    END
    IF found.0 > 1 THEN
    DO
        fnd = 0
        DO i = 1 to found.0
            IF TRANSLATE(data) = TRANSLATE(found.i) THEN
                fnd = 1
            END
        END
    END

```

*Figure 120 (Part 4 of 12). COMPCFRS.CMD Compare Two CONFIG.SYS Files REXX Utility*

```

        IF fnd = 0 THEN
        DO
            line = "1 "||data
            CALL LINEOUTX line
            line = "2 "
            CALL LINEOUTX line
            line = ""
        END
    END
    END /* otherwise */
END /* select */
END /* while */

/*
retc = LINEOUT(file1,' ')
retc = LINEOUT(file1)
retc = STREAM(file1,'c','seek = 0')
retc = STREAM(file2,'c','seek = 0')
*/

REC1 = 1; REC2 = 1

/*CALL LINEOUTX ' '*/
/*retc = LINEOUT(outfile)*/

CALL LINEOUTX ''

/*+++++*/
/* Perform the extraction of statements in the */
/* second file that are not found in the first file. */
/*+++++*/
SAY "checking second file"
DO WHILE rec2 \> words(str2) /* Perform sequential read to */
    data = XLINEIN2()

    PARSE UPPER VAR data string "=" rest
    IF string = "DEVICE" | string = "BASEDEV" | string = "RUN" |,
        string = "DEVINFO" | rest = "" THEN
    DO
        search = data
        IF string = "DEVICE" | string = "RUN" THEN
        DO
            PARSE UPPER VAR search x ' ' y
            search = x
            DO UNTIL y = ""
                PARSE UPPER VAR search x '\ ' y
                IF y <> "" THEN
                    search = y
                END
            search = '\ '||search
        END
    END

```

Figure 120 (Part 5 of 12). COMPCFRS.CMD Compare Two CONFIG.SYS Files REXX Utility

```

END
ELSE
    search = string||"="
    retc = STREAM(outfile,'c','seek = 0')
/*retc = SysFileSearch(search,outfile,"rpted")*/
CALL SYSFILESEARCH2
IF retc <> 0 THEN
DO
    CALL LINEOUTX ''
/* retc = LINEOUT(outfile)*/
/*retc = SysFileSearch(search,outfile,"rpted")*/
    CALL SYSFILESEARCH2
END
IF rpted.0 = 0 THEN
DO
    retc = STREAM(outfile,'c','seek < 0')
/* retc = SysFileSearch(search,file1,"found")*/
    REC1 = 1
    CALL SYSFILESEARCH3
    IF found.0 = 0 THEN
    DO
        line = "1 "
        CALL LINEOUTX line
        line = "2 "||data
        CALL LINEOUTX line
        line = ""
    END
    ELSE
    SELECT
        WHEN string = "IFS" THEN
        DO
            IF found.0 = 1 THEN
            DO
                PARSE UPPER VAR data x ' ' w
                PARSE UPPER VAR found.1 y ' ' z
                IF x <> y THEN
                DO
                    line = "1 "
                    CALL LINEOUTX line
                    line = "2 "||data
                    CALL LINEOUTX line
                END
            ELSE
                IF TRANSLATE(data) <> TRANSLATE(found.1) THEN
                DO
                    line = "1 "||found.1
                    CALL LINEOUTX line
                    line = "2 "||data
                    CALL LINEOUTX line
                END
            END
        END
    END
END

```

Figure 120 (Part 6 of 12). COMPCFRS.CMD Compare Two CONFIG.SYS Files REXX Utility



```

IF found.0 > 1 THEN
DO
  PARSE UPPER VAR data x ' ' w
  i = 0
  DO UNTIL x = y | i > found.0
    i = i + 1
    PARSE UPPER VAR found.i y ' ' z
  END
  IF i > found.0 THEN
  DO
    line = "1 "
    CALL LINEOUTX line
    line = "2 "||data
    CALL LINEOUTX line
    line = ""
  END
  IF x = y & TRANSLATE(data) <> TRANSLATE(found.i) THEN
  DO
    line = "1 "||found.i
    CALL LINEOUTX line
    line = "2 "||data
    CALL LINEOUTX line
    line = ""
  END
  END
  CALL LINEOUTX " "
END
WHEN string = "LIBPATH" THEN
DO
  NOP
END
WHEN string = "SET PATH" THEN
DO
  NOP
END
WHEN string = "SET DPATH" THEN
DO
  NOP
END
WHEN string = "SHELL=" THEN
DO
  DO i = 1 UNTIL x = string | i > found.0
    PARSE UPPER VAR found.i x '=' y
    x = x||'|'
  END

```

Figure 120 (Part 7 of 12). COMPCFRS.CMD Compare Two CONFIG.SYS Files REXX Utility

```

        IF i > found.0 THEN
        DO
            line = "1 "
            CALL LINEOUTX line
            line = "2 "||data
            CALL LINEOUTX line
        END
    END
    OTHERWISE
    DO
        IF found.0 = 1 & TRANSLATE(data) <> TRANSLATE(found.1) THEN
        DO
            line = "1 "||found.1
            CALL LINEOUTX line
            line = "2 "||data
            CALL LINEOUTX line
        END
        IF found.0 > 1 THEN
        DO
            fnd = 0
            DO i = 1 to found.0
                IF TRANSLATE(data) = TRANSLATE(found.i) THEN
                    fnd = 1
                END
            END
            IF fnd = 0 THEN
            DO
                line = "1 "
                CALL LINEOUTX line
                line = "2 "||data
                CALL LINEOUTX line
            END
        END
    END /* otherwise */
END /* select */
END /* IF */
CALL LINEOUTX ''
/*retc = LINEOUT(outfile)*/
END

CALL LINEOUTX LEFT('',74,'')
retc = LINEOUT(file1)
retc = LINEOUT(file2)

outbuffer = substr(outbuffer,1,length(outbuffer)-2)
retc = LINEOUT(outfile, outbuffer)
retc = LINEOUT(outfile)

return

```

Figure 120 (Part 8 of 12). COMPCFRS.CMD Compare Two CONFIG.SYS Files REXX Utility

```

/*
EQUAL_PATH:
  ARG path1, path2, pos

  path = SUBSTR(path1,pos)
  DO i = 1 BY 1 UNTIL path = ""
    PARSE UPPER VAR path dirs.i ';' path
    dirs.i = dirs.i||';'
  END
  dirs.0 = i
  nfnd = ""
  path = SUBSTR(path2,pos)
  DO i = 1 BY 1 UNTIL i = dirs.0
    PARSE UPPER VAR path x (dirs.i) y
    IF y = "" & LENGTH(path) = LENGTH(x) THEN
      nfnd = nfnd || dirs.i
      path = x||y
    END
  END
  IF path2 <> "" | nfnd <> "" THEN
    DO
      type = SUBSTR(path1,1,pos-2)
      line = "1 " || type || " - NOT IN 1 : "||nfnd
      CALL LINEOUTX line
      line = "2 " || type || " - NOT IN 2 : "||path
      CALL LINEOUTX line
    END
  END
  return

XLINES1:
  if rec1 > words(str1) then return(0)
  return(1)

XLINEIN1:
  xarg = TRANSLATE(word(STR1,rec1),' ','-')
  if xarg <> '' then rec1 = rec1 + 1
  return(XARG)

XLINES2:
  if rec2 > words(str2) then return(0)
  return(1)

XLINEIN2:
  xarg = translate(word(str2,rec2),' ','-')
  if xarg <> '' then rec2 = rec2 + 1
  return(XARG)

```

Figure 120 (Part 9 of 12). COMPCFRS.CMD Compare Two CONFIG.SYS Files REXX Utility

```

LINEOUTX:
arg xarg

    lng = length(xarg)

    if lng = 0 then return

    if xarg = '' then xarg = left('', lng) /* string of spaces */
    outbuffer = outbuffer||xarg||'0d0a'x

return

SYSFILESEARCH1:
/*retc = SysFileSearch(search,file2,"found")*/
    IDX = 0
    RETC = 0
    FOUND.0 = IDX
    WSEARCH = TRANSLATE(SEARCH,'-',' ')
    PSN = POS(WSEARCH ,STR2)
    IF PSN = 0 THEN RETURN

    DO REC = 1 FOR WORDS(STR2)
        WRD = WORD(STR2,REC)
        PSN = POS(WSEARCH,WRD)
        IF PSN > 0 THEN
            DO
                IDX = IDX + 1
                FOUND.IDX = TRANSLATE(WRD,' ','-')
            END
        END
    FOUND.0 = IDX
RETURN

SYSFILESEARCH2:
/*retc = SysFileSearch(search,outfile,"rpted")*/
    rpted.0 = POS(search,OUTBUFFER)
    RETC = 0
RETURN

SYSFILESEARCH3:
/* retc = SysFileSearch(search,file1,"found")*/
    IDX = 0
    RETC = 0
    FOUND.0 = IDX
    WSEARCH = TRANSLATE(SEARCH,'-',' ')
    PSN = POS(WSEARCH ,STR1)
    IF PSN = 0 THEN RETURN

```

Figure 120 (Part 10 of 12). COMPCFRS.CMD Compare Two CONFIG.SYS Files  
REXX Utility

```

DO REC = 1 FOR WORDS(STR1)
  WRD = WORD(STR1,REC)
  PSN = POS(WSEARCH,WRD)
  IF PSN > 0 THEN
    DO
      IDX = IDX + 1
      FOUND.IDX = TRANSLATE(WRD,' ','-')
    END
  END
  FOUND.0 = IDX
RETURN

XINIT:
  outbuffer = ''; rec1 = 1; rec2 = 1; str1 = ''; str2 = ''; T1. = ''; T2. = ''

  dat1 = translate(charin(file1,1,stream(file1,'c','query size')))
  dat2 = translate(charin(file2,1,stream(file2,'c','query size')))

  dat1 = translate(dat1,'- ','200d0a1a'x)
  dat2 = translate(dat2,'- ','200d0a1a'x)

  retc = LINEOUT(file1)
  retc = LINEOUT(file2)

  /* drop duplicate spaces */
  do forever
    parse var dat1 dat1 '- ' rem
    if rem = '' then leave
    dat1 = strip(strip(dat1||'- '||strip(strip(rem,'- ')),'- '))
  end

  do forever
    parse var dat2 dat2 '- ' rem
    if rem = '' then leave
    dat2 = strip(strip(dat2||'- '||strip(strip(rem,'- ')),'- '))
  end

  /* build tables & strings*/
  do q = 1 for words(dat1)
    wrd = strip(left(word(dat1,q),247))
    T1.wrd = T1.wrd q
  end

```

Figure 120 (Part 11 of 12). COMPCFRS.CMD Compare Two CONFIG.SYS Files  
REXX Utility

```

do q = 1 for words(dat2)
  wrd = strip(left(word(dat2,q),247))
  if T1.wrd <> '' then T1.wrd = strip(delword(T1.wrd,1))
  else
    T2.wrd = T2.wrd q
  end
end

do q = 1 for words(dat1)
  wrd = strip(left(word(dat1,q),247))
  if T1.wrd <> '' then
    do
      if left(wrd,3) <> 'REM' then str1 = str1 word(dat1,q)
      T1.wrd = ''
    end
  end
end

do q = 1 for words(dat2)
  wrd = strip(left(word(dat2,q),247))
  if T2.wrd <> '' then
    do
      if left(wrd,3) <> 'REM' then str2 = str2 word(dat2,q)
      T2.wrd = ''
    end
  end
end

return

```

*Figure 120 (Part 12 of 12). COMPCFRS.CMD Compare Two CONFIG.SYS Files REXX Utility*

---

## Appendix G. Samples for Managing WorkSpace On-Demand in the Enterprise

This Appendix contains the source for the REXX tools and sample response files that were used in for the examples in Chapter 9, "Administering WorkSpace On-Demand in the Enterprise" on page 193. to automate the WorkSpace On-Demand environment. These tools are provided as examples of how to set up procedures and processes to assist you in creating new machine definitions in WorkSpace On-Demand in an enterprise environment.

---

### G.1 WSODOAF.RSP: Sample OS/2 Access Feature Response File

The response file, shown in Figure 121 on page 298, was used to create an OS/2 Access Feature configuration for another workstation. This is useful if you want to automatically create OS/2 Access Feature configurations for many workstations. Using response file like this makes it possible to do this unattended and remotely. The following command can be used to create configurations for WorkSpace On-Demand Clients:

```
CMSETUP /r:wsodoaf.rsp /11:wsodoaf.11 /12:wsodoaf.12
```

The keyword CMUSERCFG can point to any existing directory on your WorkSpace On-Demand server. The OS/2 Access Feature has to be installed on the machine where you run this command.

```

*****
* This is a response file that represents a Communications Manager
* configuration. To apply changes from this response file, execute the CMSETUP
* /R command.
* Keywords and values are documented in:
*
* Response File Reference (Online book).
*
* The Communications Manager configuration can be changed by editing this
* response file. Additional functions can be added or removed from the
* Communications Manager installation by adding the appropriate keywords to
* this response file. To perform other CID processing, one or more of the
* following keywords may need to be specified:
* CMUPDATETYPE =
* CMMODELCFG =
* CMSTOPCOMMUNICATIONS =
* CMINSTALLCURRENTFEATURES =
* CMINSTALLALLCOMMUNICATIONS =
* CMDYNAMICUPDATE =
* UPDATECONFIGSYS =
* MIGRATE =
* CMTARGET =
* CMSOURCE =
*****
* Installation keywords:

    CMUSERCFG = V:\WSODCM2\ITSOLAB2
    CMUPDATETYPE = 4

LAN_DLC = (
    NAME = 0
    ADAPTER_TYPE = 2
    MAX_LINK_STATION = 004
    PER_INCOMING_CALLS = 000
    FREE_LINK = 0
    MAX_I_FIELD_SIZE = 02224
    SEND_COUNT = 4
    RECEIVE_COUNT = 4
    CSM_LAN_ID = CMLANID
    SEND_ALERT = 0
    LINK_ESTABLISHMENT_RETRANSMISSION = 8
    MAX_ACTIVATION_ATTEMPTS = 0
    LOCAL_SAP = 04
    RETRANSMISSION_THRESHOLD = 8
    DISABLE_REACT = 0
)

```

Figure 121 (Part 1 of 4). WSODOAF.RSP OS/2 Access Feature Client Response File



```

WORKSTATION = (
    COMMENT = WorkSpace test of OS/2 AF
    * TRANS_TABLE is either not used or not configured.
      LOAD_SNA_APPC = 1
      LOAD_SRPI = 1
      LOAD_ACDI = 1
      USE_ACDI_API = 0
      USE_ARTIC_ADAPTER_WITHOUT_X25 = 0
      USE_LAN_PROTOCOLS_USING_ISDN = 0
)

ATTACH_MANAGER = (
    START = 1
)

LOCAL_CP = (
    NAME = ITSO.WSODCLT
    * COMMENT is either not used or not configured.
      CP_ALIAS = WSODCLT
    * HOST_FP_LINK_NAME is either not used or not configured.
      HOST_FP_SUPPORT = 1
      NAU_ADDRESS = 0
      NODE_ID = 05D03103
      NODE_TYPE = 3
      NW_FP_SUPPORT = 0
    * NW_SERVER_NAME is either not used or not configured.
      MAX_COMPRESSION_LEVEL = 0
      MAX_COMPRESSION_TOKENS = 0
      FREE_UNUSED_SESSIONS = 0
      FREE_UNUSED_SESSIONS_TIMEOUT = 10
      DLUR_MULTISUBNET_SUPPORT = 1
)

* LOCAL_LU is not configured.

LOGICAL_LINK = (
    NAME = LINK0001
    * COMMENT is either not used or not configured.
      DLC_NAME = IBMTRNET
      ACTIVATE_AT_STARTUP = 1
      CP_CP_SESSION_SUPPORT = 1
      ADAPTER_NUMBER = 0
      ADJACENT_NODE_TYPE = 2
      PREFERRED_NN_SERVER = 1
      DESTINATION_ADDRESS = 400009FF07A204

```

Figure 121 (Part 2 of 4). WSODOAF.RSP OS/2 Access Feature Client Response File

```

*   INIT_WITH_SNRM is either not used or not configured.
    ETHERNET_FORMAT = 0
*   OUTGOING_DIR_ENTRY_NAME is either not used or not configured.
*   PERMANENT_CONNECTION_NAME is either not used or not configured.
*   X25_DIR_ENTRY_NAME is either not used or not configured.
*   FQ_ADJACENT_CP_NAME is either not used or not configured.
    EFFECTIVE_CAPACITY = -1
    COST_PER_BYTE = -1
    COST_PER_CONNECT_TIME = -1
    PROPAGATION_DELAY = -1
    SECURITY = -1
    USER_DEFINED_1 = -1
    USER_DEFINED_2 = -1
    USER_DEFINED_3 = -1
    LIMITED_RESOURCE = 0
    LINK_STATION_ROLE = -1
*   PU_NAME is either not used or not configured.
    SOLICIT_SSCP_SESSION = 1
*   NODE_ID is either not used or not configured.
*   ADJACENT_NODE_ID is either not used or not configured.
    MAX_ACTIVATION_ATTEMPTS = 0
    USE_PUNAME_AS_CPNAME = 0
    HPR_SUPPORT = 0
    CONNECTION_TYPE = 0
*   NON_XID_REPOLL_COUNT is either not used or not configured.
    LIMITED_RESOURCE_TIMEOUT = -1
    INACTIVITY_TIMEOUT = -1
    AUTO_REACTIVATE = 0
    HOST_BACKUP_LINK = 0
)

LUA = (
    NAME = LU2A
    COMMENT = WorkSpace Test
    HOST_LINK_NAME = LINK0001
    NAU_ADDRESS = 2
    LU_MODEL_NAME is either not used or not configured.
)

LUA = (
    NAME = LU2B
*   COMMENT is either not used or not configured.
    HOST_LINK_NAME = LINK0001
    NAU_ADDRESS = 4
    LU_MODEL_NAME is either not used or not configured.
)

```

Figure 121 (Part 3 of 4). WSODOAF.RSP OS/2 Access Feature Client Response File

```

SNA_DEFAULTS = (
*   COMMENT is either not used or not configured.
    DEFAULT_TP_CONV_SECURITY_RQD = 0
    DEFAULT_TP_OPERATION = 2
    DEFAULT_TP_PROGRAM_TYPE = 0
    DIRECTORY_FOR_INBOUND_ATTACHES = *
    IMPLICIT_INBOUND_PLU_SUPPORT = 1
    DEFAULT_MODE_NAME = BLANK
*   DEFAULT_LOCAL_LU_ALIAS is either not used or not configured.
    MAX_HELD_ALERTS = 10
    MAX_MC_LL_SEND_SIZE = 32767
    IMPLICIT_LINK_HPR_SUPPORT = 1
    RETRY_COUNT = 6
    ALIVE_TIMER = 60
    PATH_SWITCH_TIMER_LOW = 480
    PATH_SWITCH_TIMER_MEDIUM = 240
    PATH_SWITCH_TIMER_HIGH = 120
    DEFAULT_ROUTING_PREFERENCE = 0
)

```

Figure 121 (Part 4 of 4). WSODOAF.RSP OS/2 Access Feature Client Response File

## G.2 PCOMOS2.WS: Sample Personal Communication/3270 for OS/2 Configuration File

The Personal Communication/3270 for OS/2 configuration file shown in Figure 122 was used to run Personal Communication/3270 for OS/2 on a WorkSpace On-Demand client. This configuration can only run in conjunction with an OS/2 Access Feature configuration as created by a response file shown in Figure 121 on page 298. The 3270 emulation runs over the 3270-LUA definition from the OS/2 Access Feature.

```

[Profile]
ID=WS
Description=

[Translation]
IBMDefaultView=Y
DefaultView=
IBMDefaultDBCS=Y
DefaultDBCS=

```

Figure 122 (Part 1 of 4). PCOMOS2.WS Personal Communication/3270 for OS/2 Configuration File

```
[Communication]
AutoConnect=N
Link=rui
Session=3270

[SLAN]
Adapter=0
DestinationSAP=04
GatewayAddress=400009FF07A2
Identifier=05D03103
LinkStations=1
ReceiveBufferCount=12
ReceiveBufferSize=521
SourceSAP=04
TraceName=snalantrace0
TransmitFrameSize=521
CPName=
ExchangeID=N

[LU]
CompEnabled=N
CompBufSize=4096
SLE=N

[Telnet3270]
HostName=wtscpok
HostPortNumber=23
LUName=
AutoReconnect=N
ExtendedColor=Y
ATTN=6CFF0000
SYSREQ=F0FF00
```

*Figure 122 (Part 2 of 4). PCOMOS2.WS Personal Communication/3270 for OS/2 Configuration File*

```

[3270]
ScreenSize=43x80
SessionType=Display
HostGraphics=Y
NativeGraphics=N
LoadPSS=N
QueryReplyMode=Auto
CharacterCellSize=9x16
HostCodePage=037-U
LtNumber=FF
LuNumber=FF

[Keyboard]
TypeAhead=Y
ResetInsertbyAttn=N
SpotConversion=Y
CaretSizeControl=Y
CheckCodepage=Y
Language=United-States
IBMDefaultKeyboard=N
DefaultKeyboard=C:\PCOMOS2\PRIVATE\indran.KMP

[Transfer]
HostSystem=1
DefaultVMDisk=a
DefaultDataSet=
DefaultLibrary=qgp1
DefaultDirectory=C:\PCOMOS2
ResumeSendFileName=
ResumeRecvFileName=

[Window]
SessFlags=8C6A
ViewFlags=4E00
CaptionFormat=2A -
UserTitle=PCSW
RuleLinePos=0 0
InsertCursor=Y
MFIColor=N

```

Figure 122 (Part 3 of 4). PCOMOS2.WS Personal Communication/3270 for OS/2 Configuration File

```
[Graphics]
Redraw=Retained
CursorStyle=1
CursorMovement=0

[RUI]
LUName=LU2A
StartCM=Y
StopCM=Y
AutoReconnect=Y
```

*Figure 122 (Part 4 of 4). PCOMOS2.WS Personal Communication/3270 for OS/2 Configuration File*

---

### **G.3 WSODINST.CMD: Installing WorkSpace On-Demand Unattended**

We created the WSODINST.CMD, shown in Figure 123 on page 305, to run the WorkSpace On-Demand installation unattended. The program creates the NCINSTALL.RSP for the feature install by concatenating the variable and fixed parts for this file. It runs the INSTALL.EXE program for the WorkSpace On-Demand installation and logs the output to two log files.

```

/* REXX script for installation of WorkSpace On-Demand */
call RxFuncAdd 'SysLoadFuncs', 'RexxUtil', 'SysLoadFuncs'
call SysLoadFuncs

BootDrive = ""
NN_path   = ""
Logfile1  = ""
Logfile2  = ""
WsodRsp   = ""
Ncvarrsp  = ""
Ncfixrsp  = ""
CID       = 1

parse upper arg parms

call Defaults
call ParseArgs(parms)
say "Logging to :" Logfile1
call LogMsg "Invocation parameters: "parms

select
  when WsodRsp = "" then
    do
      call Logmsg "/R2 parameter missing"
      call Usage
    end /* do */
  when Ncfixrsp = "" then
    do
      call Logmsg "/RF parameter missing"
      call Usage
    end /* do */
  when Ncvarrsp = "" then
    do
      call Logmsg "/RV parameter missing"
      call Usage
    end /* do */

```

Figure 123 (Part 1 of 8). WSODINST.CMD - REXX Batch to Install WorkSpace On-Demand

```

when BootDrive = "" then
do
    call Logmsg "/B parameter missing"
    call Usage
end /* do */
when Logfile1 = "" then
do
    call Logmsg "/L1 parameter missing"
    call Usage
end /* do */
when Logfile2 = "" then
do
    call Logmsg "/L2 parameter missing"
    call Usage
end /* do */
when NN_path = "" then
do
    call Logmsg "/NN parameter missing"
    call Usage
end /* do */
when Sourcepath = "" then
do
    call Logmsg "/S parameter missing"
    call Usage
end /* do */

otherwise
do
    call Logmsg "Logfile1      : " || Logfile1
    call Logmsg "Logfile2      : " || Logfile2
    call Logmsg "WsodRsp       : " || WsodRsp
    call logmsg "Ncfixrsp      : " || Ncfixrsp
    call logmsg "Ncvarrsp     : " || Ncvarrsp
    call logmsg "NN_path       : " || NN_path
end /* do */

end /* select */

```

Figure 123 (Part 2 of 8). WSODINST.CMD - REXX Batch to Install WorkSpace On-Demand



```

ncinstrsp = BootDrive || "\OS2\INSTALL\NCINSTAL.RSP"
SourceDrive = left(Sourcepath,2)

SourceDrive

call Logmsg "Creating NCINSTAL.RSP : " || ncinstrsp
cmd = 'copy 'ncvarrsp || '+' || ncfixrsp || ' ' || ncinstrsp
call logmsg "Using command : "cmd
cmd
if RC<>0 then
do
    call Logmsg "Error executing : "cmd
    exit x2d('0812')
end /* do */

call logmsg "Changing to the Sourcedirectory"
cmd = 'cd ' Sourcepath
cmd

if RC<>0 then
do
    call Logmsg "Sourcedirectory not found "
    exit x2d('0812',4)
end

call Logmsg "WorkSpace On-DemandInstallation starts here !!!"
cmd = 'install /B:' || BootDrive ,
      || ' /L1:'Logfile1 ,
      || ' /L2:'Logfile2 ,
      || ' /NN:'NN_path ,
      || ' /R2:'WsodRsp ,
      || ' /r:'ncinstrsp ,
      || ' /X'
call logmsg "Program invocation for WorkSpace On-Demand Installation:"
call logmsg cmd

cmd

```

Figure 123 (Part 3 of 8). WSODINST.CMD - REXX Batch to Install WorkSpace On-Demand

```

rc2 = rc

if (d2x(rc) <> 'FE00') then
do
    call Logmsg "Returncode of : "cmd
    call Logmsg "Rc = "rc2
    exit (rc2)
end /* do */
exit (rc2)

/*****/

Defaults: procedure expose BootDrive Logfile1 Logfile2 NN_path

    BootDrive ="E:"
    Logfile1  = BootDrive || "\wsod.l1"
    Logfile2  = BootDrive || "\wsod.l2"
    NN_path   = BootDrive || "\NETSCAPE"
    return

/*****/

ParseArgs: procedure expose WsodRsp Ncfixrsp Ncvarrsp BootDrive ,
    Logfile1 Logfile2 NN_path Sourcepath CID

    parse arg parms

    parms = translate(parms)
    numparms = words(parms)
    do i=1 to numparms by 1
        parm = word(parms,i)
        select

```

Figure 123 (Part 4 of 8). WSODINST.CMD - REXX Batch to Install WorkSpace On-Demand

```

when left(parm,3) = "/B:" then
do
    say "/B: Parameter found: "
    BootDrive=right(parm,length(parm)-3);
    say "BootDrive: " BootDrive
end
when left(parm,4) = "/RF:" then
do
    say "/RF: Parameter found: "
    Ncfixrsp=right(parm,length(parm)-4);
    say "Ncfixrsp value : "Ncfixrsp
end
when left(parm,4) = "/RV:" then
do
    say "/RV: Parameter found: "
    Ncvarrsp=right(parm,length(parm)-4);
    say "Ncvarrsp value : "Ncvarrsp
end
when left(parm,4) = "/R2:" then
do
    say "/R2: Parameter found: "
    WsodRsp=right(parm,length(parm)-4);
    say "WsodRsp value : "WsodRsp
end
when left(parm,4) = "/L1:" then
do
    say "/L1: Parameter found: "
    Logfile1=right(parm,length(parm)-4);
    say "Logfile1 value : " Logfile1
end
when left(parm,4) = "/L2:" then
do
    say "/L2: Parameter found: "
    Logfile2=right(parm,length(parm)-4);
    say "Logfile2 value : " Logfile2
end

```

Figure 123 (Part 5 of 8). WSODINST.CMD - REXX Batch to Install WorkSpace On-Demand

```

when left(parm,4) = "/NN:" then
do
    say "/NN: Parameter found: "
    NN_path=right(parm,length(parm)-4);
    say "NN_path value :" NN_path
end
when left(parm,3) = "/S:" then
do
    say "/S: Parameter found: "
    Sourcepath=right(parm,length(parm)-3);
    say "Sourcepath value :" Sourcepath
end
when left(parm,2) = "/"? then
do
    call Usage
end
when left(parm,2) = "/X" then
do
    CID = 1
end
otherwise
do
    say "Incorrect parameter : "parm
    call Usage
end /* do */
end
end
return

```

*Figure 123 (Part 6 of 8). WSODINST.CMD - REXX Batch to Install WorkSpace On-Demand*

```

/*****/

Usage:

say ""
say "Correct Program Invocation"
say "+++++"
say "WSODINST [parameters]"
say ""
say "The following parameters are possible:"
say ""
say "/B:BootDrive"
say "/RF:Responsefile"
say "    The fix part of NCINSTALL.RSP"
say "/RV:Responsefile"
say "    The variable part of NCINSTALL.RSP"
say "/R2:Responsefile"
say "    The WorkSpace On-Demand configuration response file"
say "/NN:Path"
say "    Defines the path for the Netscape Navigator"
say "/S:Path"
say "    Defines the Sourcepath of WorkSpace On-Demand"
say "/L1:Logfile"
say "    Historylog"
say "/L2"
say "    Errorlog"
say "/"
say "This output is shown."
exit x2d("1600",4)

```

Figure 123 (Part 7 of 8). WSODINST.CMD - REXX Batch to Install WorkSpace On-Demand

```

/*****/

LogMsg: procedure expose Logfile1
parse arg msg
if (stream( Logfile1 , 'c', 'query exists') = "") then
do
    call stream Logfile1, 'c', 'open write'
    call lineout Logfile1 ,,1
end
else
do
    call stream Logfile1, 'c', 'open write'
    call lineout LogFile, ""
end
msg = '[' || date() || ' ' || time() || ' ] ' || msg
say msg
call lineout Logfile1, msg
call stream Logfile1, 'c', 'close'
return

```

*Figure 123 (Part 8 of 8). WSODINST.CMD - REXX Batch to Install WorkSpace On-Demand*

## G.4 QUERYCFG.CMD: Gathering the Client's Hardware Configuration

We created a batch named QUERYCFG.CMD, shown in Figure 124 on page 313, that is used as the default shell batch to be executed from the CMD.EXE. This batch asks for the station ID, runs the TME 10 NetFinity System Information Tool and stores the data to a directory on the server. After the server process of defining the new client ends, the TSHUTDOWN.EXE is called to reboot the client.

```

Configuration
/* QuerySystem */
'z:'
'cd \util'
ok="new"
do while ok<>" "
    say "Please enter the station ID :"
    pull id
    id=translate(id)
    idfile= "Z:\" || id || ".data"
    okfile = "Z:\" || id || ".OK"
    call stream okfile, 'c', 'query exists'
    ok=result
    if ok<>" " then
        do
            say ""
            say "Duplicate name, Please try again !"
            say ""
        end /* do */
    end /* do */
end /* do */

say "Querying system information. "
say "Please wait..."
'cd \netfin.40\bin'
cmd = 'sinfg30 /P:' || idfile || ' /B /NOLOGO'
cmd
cmd = 'echo =====>' idfile
do 3
    cmd
end
cmd = ' echo LAN adapter info added by LANTRAN.LOG >> ' idfile
cmd
cmd = 'echo =====>' idfile
do 3
    cmd
end
cmd = 'type lantran.log >> ' idfile
cmd

okfile = "Z:\" || id || ".OK"
say "Waiting for Server Process to create my definition"
do while stream(okfile,'c','query exists') = ""
    NOP
end
'Tshutdwn'

```

Figure 124. QUERYCFG.CMD - REXX Batch for Gathering Client's Hardware

## G.5 ADDNEWCL.CMD: Automatted WorkSpace On-Demand Client Definition

The ADDNEWCL.CMD program, shown in Figure 125, automatically adds WorkSpace On-Demand clients to a server. The configuration data is generated with the program QUERYCFG shown in Figure 124 on page 313.

```
/* Sample program to automatically add new clients to an
   existing WorkSpace On-Demand Server */

call RxFuncAdd 'SysLoadFuncs', 'RexxUtil', 'SysLoadFuncs'
call SysLoadFuncs

/* This program requires an administrator LOGON */
'logon admin /p:wsod /v:d'

NewClBase='D:\NEWCLTS' /* Base Directory for Client.DATA */
RPL='d:\ibmlan\rpl\rpl.map'
AdpLine='Adapter 0 is using node address '
LogFile1='D:\NEWCLTS\NEWCLTS.LOG'

starttime=time()
stardate=date()
/* The names are copied from
   D:\IBMLAN\RPL\BB10.US\IBMCOM\MACS\*.NIF */

AdpList.0=2
AdpList.1.model="MODEL"
AdpList.1.name="IBM PCI Token-Ring Adapter (IBMTRP.OS2)"
AdpList.1.dummy="00062?????"
AdpList.2.model="MDGMODEL"
AdpList.2.name="Madge FastMac OS/2 NDIS driver for Smart 16/4 Ringnodes"
AdpList.2.dummy="0000F?????"
```

Figure 125 (Part 1 of 8). ADDNEWCL.CMD Program to Automatically Create WorkSpace On-Demand Clients



```

do while(1) /* Run forever */
  say "AddClts: [StartDate :\"Startdate\"/\"Starttime \"]" ,
    " [Last Scan:\"date()\"/\"time() \"]"
  Cltlist.0=0
  call SysFileTree NewCltBase\'*.data', 'CltList', 'SF0'
  i=0
  if CltList.0 <> 0 then
    do
      call Logmsg "*****"
      call logmsg "**** Creating New Client ****"
      call Logmsg "*****"
      do CltList.0
        i=i+1
        cltfile=filespec("name",CltList.i);
        CltList.i=left(cltfile,pos('.',cltfile)-1)
        call logmsg "Found data file :\"CltList.i"
      end /* do */

      /* CltList.0 contains the number of .DATA files found */

      actclt.0=0
      i=0
      do while queued() > 0
        pull entry
      end /* do */
      'net riplmach | rxqueue' /* Querying info to pipe */
      /* Reading list of active clients */

      do while queued() > 0
        pull entry
        i=i+1
        actclt.i=entry
        say "actClt.\"i\" = \"actclt.i"
        actclti = translate(actclti)
        if pos('ENABLED',actclti) = 0 & pos('DISABLED',actclti) = 0 then
          do
            i=i-1
          end
        else
          do
            /* reduce line to the client's name */
            actclti=word(actclti,1)
            call logmsg "Active Client <i> :\"actclt.i"
          end
        end /* do */
        actclt0=i
      end while
    end
  end
end /* do */

```

Figure 125 (Part 2 of 8). ADDNEWCL.CMD Program to Automatically Create WorkSpace On-Demand Clients

```

/* Testing the .DATA files against the existing clients */
i=0
do CltList.0
  found=0
  i=i+1
  do j=1 to actClc.0
    if CltList.i = actClc.j then
      do
        found=1
        leave
      end /* do */
    end /* do */
  if found=0 then
    do
      call logmsg "Client name : "ClcList.i
      call ScanClc ClcList.i
      call CreateClc(ClcList.i)
      /* Now rename the .DATA file to .OK
      to show it's correctly created
      and to reduce scan time for the
      next run */
      'ren 'NewClcBase'\ 'ClcList.i'.DATA *.OK'
    end /* do */
  end /* do */
end
call SysSleep 60 /* Waiting one minute */
end /* do */

exit

```

*Figure 125 (Part 3 of 8). ADDNEWCL.CMD Program to Automatically Create WorkSpace On-Demand Clients*

```

/*****/

LogMsg: procedure expose Logfile1
parse arg msg
if (stream( Logfile1 , 'c', 'query exists') = "") then
do
    call stream Logfile1, 'c', 'open write'
    call lineout Logfile1 ,,1
end
else
do
    call stream Logfile1, 'c', 'open write'
    call lineout Logfile1, ""
end
msg = '[' || date() || ' ' || time() || ' ] ' || msg
say msg
call lineout Logfile1, msg
call stream Logfile1, 'c', 'close'
return
/*****/

CreateClt: procedure expose NewCltBase RPL AdpLine ,
            Logfile1 address AdpList.

parse upper arg cltname

tmprpl='D:\IBMLAN\RPL\rp1.tmp'

'del 'tmprpl

call stream RPL , 'c' , 'open read'
if result <> 'READY:' then
do
    call LogMsg 'Error opening file : 'RPL
    exit x2d('0812',4)
end /* do */
call stream tmprpl , 'c' , 'open write'

```

*Figure 125 (Part 4 of 8). ADDNEWCL.CMD Program to Automatically Create WorkSpace On-Demand Clients*

```

if result <>'READY:' then
do
    call LogMsg 'Error opening file : 'tmprpl
    exit x2d('0812',4)
end /* do */

call LogMsg "Reading: "RPL
call LogMsg "Writing: "tmprpl

adptype=""

do while lines(RPL)
    dataline=linein(RPL)
    if pos(address,dataline) = 0 then
    do
        call lineout tmprpl, dataline
    end /* do */
    else
    do
        /* Adapter Address line found */
        call LogMsg "Address entry found !"
        call LogMsg "dataline : "dataline
        /* The second field contains the name of the model */
    end /* do */
    adp=0
    do AdpList.0
        adp=adp+1
        if pos(AdpList.adp.dummy,dataline) <> 0 then
        do
            tmpname=word(dataline,2)
            call QueryAdp(tmpname)
        end /* do */
    end /* do */
    end /* do */
    call stream RPL , 'c', 'close'
    call stream tmprpl , 'c', 'close'
    cmd = 'copy 'tmprpl RPL
    cmd

```

*Figure 125 (Part 5 of 8). ADDNEWCL.CMD Program to Automatically Create WorkSpace On-Demand Clients*

```

if rc<>0 then
do
    call LogMsg "Error copying files: "cmd
    exit x2d('1604',4)
end /* do */
class = 'ISAVGA'
remark = "'WsoD Client ' || cltname ||'"
call LogMsg "Creating new Client"
call LogMsg "Client Name      : " cltname
call LogMsg "Adapter-type    : " adptype
call LogMsg "Adapter address : " address
call LogMsg "Machine class   : " class
call LogMsg "Remark          : " remark

cmd = 'net riplmach ' ,
      || cltname ,
      ' /add /os:bb10.us ' ,
      ' /c:'class ,
      ' /mac:'address ,
      ' /rem:'remark ,
      ' /ada:"'adptype || '"" ' ,
      ' 1>>'LogFile1 ,
      ' 2>>&1'
call LogMsg "Using command : "cmd
cmd
rc2=rc
if rc2<>0 then
do
    call LogMsg "Error creating new client : "cltname
    call LogMsg "Return Code : "rc2
    return
end /* do */
else
do
    call LogMsg "Client <"cltname"> succesfully defined !"
    return
end /* do */

```

*Figure 125 (Part 6 of 8). ADDNEWCL.CMD Program to Automatically Create WorkSpace On-Demand Clients*

```

/*****/

ScanCl: procedure expose NewClBase RPL AdpLine ,
        Logfile1 address AdpList.

parse upper arg cltname

address = ""
cltdata=NewClBase || '\ ' || cltname || '.DATA'

call stream cltdata , 'c' , 'open read'
if result <> 'READY:' then
do
    call LogMsg 'Error opening file : 'cltdata
    exit x2d('0812',4)
end /* do */
call LogMsg "Client's Data file : "cltdata
do while lines(cltdata)
    dataline=linein(cltdata)
    if pos(AdpLine,dataline) <> 0 then
do
    /* Adapter Address line found */
    address=right(dataline,13)
    /*The end of line is a '.' */
    address=left(address,12)
    leave
end /* do */
end /* do */
call stream cltdata , 'c' , 'close'
return

```

*Figure 125 (Part 7 of 8). ADDNEWCL.CMD Program to Automatically Create WorkSpace On-Demand Clients*

```

/*****
QueryAdp: procedure expose AdpList. Logfile1 adptype

parse upper arg cltname
i=1
do AdpList.0
  if AdpList.1.model = cltname then
    do
      adptype=AdpList.i.name
      leave
    end
  end /* do */
end /* do */
if adptype="" then
do
  call logmsg "No definition in AdpList. found "
  exit x2d('1604',4)
end /* do */
return
*****/

```

*Figure 125 (Part 8 of 8). ADDNEWCL.CMD Program to Automatically Create WorkSpace On-Demand Clients*

## G.6 ADDAPPL.CMD: Creating WorkSpace On-Demand Applications Unattended

The REXX batch program ADDAPPL.CMD, shown in Figure 127 on page 329, was used to create WorkSpace On-Demand applications unattended. The response file shown in Figure 127 on page 329 was used as the input.

```

/* Create WorkSpace On-Demand application definitions
   with response files */
call RxFuncAdd 'SysLoadFuncs', 'RexxUtil', 'SysLoadFuncs'
call SysLoadFuncs

Logfile   = ""
RspFile   = ""
BootDrive = ""

parse upper arg parms

call Defaults
call ParseArgs(parms)
say "Logging to :" LogFile
call LogMsg "Invocation parameters: "parms

select
  when RspFile = "" then
  do
    call Logmsg "/R parameter missing"
    call Usage
  end /* do */
  otherwise
  do
    call Logmsg "Logfile       : " || Logfile
    call Logmsg "ResponseFile : " || RspFile
  end /* do */
end /* select */

```

*Figure 126 (Part 1 of 7). ADDAPPL.CMD - REXX Batch to Create WorkSpace On-Demand Application Definitions*



```

call Logmsg "Processing Response-File"
call CreateApp

exit

/*****/

Defaults: procedure expose BootDrive Logfile

    BootDrive = 'E:'
    Logfile   = BootDrive || "\MC_Class.log"
    return
/*****/

CreateApp: procedure expose Logfile RspFile

    call stream RspFile, 'c', 'open read'
    if result <> "READY:" then
        do
            call Logmsg "Error opening responsefile " || RspFile
            call Logmsg "exiting program"
            exit -1
        end

    do while lines(RspFile)
        appline=linein(RspFile)
        if appline="" | left(appline,1)="*" then
            iterate
        if pos("[",appline) <> 0 then
            do
                call Logmsg "Application entry found"
                param.0 = 0
                wsodparam.0 = 0
                ws = 0
                i = 0
                usr = 0
            end
        end
    end

```

Figure 126 (Part 2 of 7). ADDAPPL.CMD - REXX Batch to Create WorkSpace On-Demand Application Definitions

```

do while left(appline,1) <> '(' /* Data starts after ( */
  appline= translate(linein(RspFile))
  iterate
end
appline = translate(linein(RspFile))
do while left(appline,1) <> ')' /* Data ends with ) */
  if left(appline,1) = '*' then
    do
      appline = translate(linein(RspFile))
      iterate
    end /* do */
    if left(appline,2) = 'ID' then
      do
        applid = word(appline,3)
        appline = translate(linein(RspFile))
        iterate
      end /* do */
      if pos('PARAM', appline) <> 0 then
        do
          ws = ws + 1
          wsodparam.0 = ws
          wsodparam.ws.p = word(appline,2)
          wsodparam.ws.v = word(appline,4)
        end /* do */
      else
        if pos('USER', appline) <> 0 then
          do
            usr = usr + 1
            wsodusr.0 = usr
            wsodusr.usr.u = word(appline,3)
          end /* do */
        end
      end
    end
  end
end

```

*Figure 126 (Part 3 of 7). ADDAPPL.CMD - REXX Batch to Create WorkSpace On-Demand Application Definitions*

```

else
do
    i=i+1
    param.0 = i
    param.i.p = word(appline,1)
    param.i.v = word(appline,3)
end /* do */
appline= translate(linein(RspFile))
end /* do */
call Logmsg "Application found : "applid
cmd = 'cmd /c net app /add ' || applid
j=0
do param.0
    j = j + 1
    cmd = cmd || ' /' || param.j.p || ':' || param.j.v
end /* do */
cmd = cmd || ' 1>>'Logfile || ' 2>>&1'
call logmsg "Executing command " cmd
rc2=rc
if rc2 <> 0 then
do
    call logmsg "Error executing command "
    call stream RspFile , 'c' , 'close'
    call logmsg "RC = "rc2
    exit x2d('1604',4)
end /* do */
if ws <> 0 then
do
    j=0
    do wsodparam.0
        j=j+1
        cmd = 'cmd /c net appparm ' || applid || ' ' ,
            | wsodparam.j.p || '=' ,
            | wsodparam.j.v ,
            | ' /ADD /F:E ' ,
            | ' /PLACEMENT:E' ,
            | ' 1>>' Logfile || ' 2>>&1'
        call logmsg "Executing command " cmd
        cmd
    end do
end do

```

Figure 126 (Part 4 of 7). ADDAPPL.CMD - REXX Batch to Create WorkSpace On-Demand Application Definitions

```

        rc2=rc
        if rc2 <> 0 then
        do
            call logmsg "Error executing command "
            call logmsg "RC = "rc2
            call stream RspFile , 'c' , 'close'
            exit x2d('1604',4)
        end /* do */
    end /* do */
end /* do */
if wsodusr.0 <>0 then
do
    j=0
    do wsodusr.0
        j=j+1
        cmd = 'cmd /c net user ' || wsodusr.j.u ,
            || ' /ASSIGN PUBLIC:' || applid
        call logmsg "Assigning appl. to user :" cmd
        cmd
        /* Waiting for net.exe removed from memory */
        call SysSleep 2
        rc2=rc
        if rc2 <> 0 then
        do
            call logmsg "Error executing command "
            call logmsg "RC = "rc2
            call stream RspFile , 'c' , 'close'
            exit x2d('1604',4)
        end /* do */
    end /* do */
    end /* do if wsodusr.0 <> 0 */
end /* do */

call Logmsg "Applications successully created "
call stream RspFile , 'c' , 'close'
return

```

*Figure 126 (Part 5 of 7). ADDAPPL.CMD - REXX Batch to Create WorkSpace On-Demand Application Definitions*

```

/*****/

ParseArgs: procedure expose MC_Model RspFile MC_Name Logfile ReadMc

    parse arg parms

    parms = translate(parms)
    numparms = words(parms)
    do i=1 to numparms by 1
        parm = word(parms,i)
        select
            when left(parm,3) = "/R:" then
                do
                    say "/R: Parameter found: "
                    RspFile=right(parm,length(parm)-3);
                    say "RspFile value : "RspFile
                end
            when left(parm,3) = "/L:" then
                do
                    say "/L: Parameter found: "
                    Logfile=right(parm,length(parm)-3);
                    say "Logfile value : " Logfile
                end
            when left(parm,2) = "/" then
                do
                    call Usage
                end
            otherwise
                do
                    say "Incorrect parameter was used"
                    call Usage
                end /* do */
        end
    end
end
return

```

Figure 126 (Part 6 of 7). ADDAPPL.CMD - REXX Batch to Create WorkSpace On-Demand Application Definitions

```

/*****/

Usage:

say ""
say "Correct Program Invocation"
say "+++++"
say "ADDAPPL [parameters]"
say ""
say "The following parameters are possible:"
say ""
say "/R:Responsefile"
say "    The parameters for the new applications"
say "/L:Logfile"
say "    The processing is logged in the given Logfilename"
say "    Defaults to <BootDrive>:\MC_Class.log"
say "/?"
say "This output is shown."
exit x2d("1600",4)
/*****/

LogMsg: procedure expose Logfile
parse arg msg

if (stream( Logfile , 'c', 'query exists') = "") then
do
    call stream Logfile, 'c', 'open write'
    call lineout Logfile ,,1
end
else
do
    call stream Logfile, 'c', 'open write'
    call lineout LogFile, ""
end
msg = '[' || date() || ' ' || time() || ' ] ' || msg
say msg
call lineout Logfile, msg
call stream Logfile, 'c', 'close'
return

```

Figure 126 (Part 7 of 7). ADDAPPL.CMD - REXX Batch to Create WorkSpace On-Demand Application Definitions

---

## G.7 ADDAPPL.RSP: Responsefile for Application Creation

The response file, shown in Figure 127, was used as a sample response file to create WorkSpace On-Demand applications unattended using the batch REXX command shown in G.6, “ADDAPPL.CMD: Creating WorkSpace On-Demand Applications Unattended” on page 321.

```
Application Definitions
*Sample Response file for adding applications
[App]
(
* a simple command line
ID = Wsod_CMD
TYPE = WSOD
INTERFACE = VIO
APPDIR = PROGRAM
COMMAND = "CMD.EXE"
APPDRIVE = K:
DOMAIN = WSOD_1
PARAM DPATH = K:\
REMARK = "Command Line"
USER = USR1
USER = USR2
)
```

*Figure 127 (Part 1 of 2). WorkSpace On-Demand ADDAPPL.RSP - Response File to Create WorkSpace On-Demand*

```
[App]
(  
* PM Camera  
ID = PMCamera  
APPDIR = PROGRAM\PMCAM200  
COMMAND = "PMCAM200.EXE"  
TYPE = WSOD  
INTERFACE = PM  
APPDRIVE = P:  
REMARK = "PM Camera"  
DOMAIN = WSOD_1  
PARAM DPATH = D:\  
PARAM LIBPATH = D:\  
USER = USR3  
USER = USR4  
)
```

*Figure 127 (Part 2 of 2). WorkSpace On-Demand ADDAPPL.RSP - Response File to Create WorkSpace On-Demand*



---

## Appendix H. Special Notices

This publication is intended to help IBM business partners, customer technical LAN administrators, and IBM information technical specialists to install, configure and administrator WorkSpace On-Demand. The information in this publication is not intended as the specification of any programming interfaces that are provided by WorkSpace On-Demand or OS/2 Warp Server. See the PUBLICATIONS section of the IBM Programming Announcement for WorkSpace On-Demand for more information about what publications are considered to be product documentation.

References in this publication to IBM products, programs or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM product, program, or service is not intended to state or imply that only IBM's product, program, or service may be used. Any functionally equivalent program that does not infringe any of IBM's intellectual property rights may be used instead of the IBM product, program or service.

Information in this book was developed in conjunction with use of the equipment specified, and is limited in application to those specific hardware and software products and levels.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to the IBM Director of Licensing, IBM Corporation, 500 Columbus Avenue, Thornwood, NY 10594 USA.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact IBM Corporation, Dept. 600A, Mail Drop 1329, Somers, NY 10589 USA.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The information contained in this document has not been submitted to any formal IBM test and is distributed AS IS. The information about non-IBM ("vendor") products in this manual has been supplied by the vendor and IBM assumes no responsibility for its accuracy or completeness. The use of this information or the implementation of any of these techniques is a customer

responsibility and depends on the customer's ability to evaluate and integrate them into the customer's operational environment. While each item may have been reviewed by IBM for accuracy in a specific situation, there is no guarantee that the same or similar results will be obtained elsewhere. Customers attempting to adapt these techniques to their own environments do so at their own risk.

Any performance data contained in this document was determined in a controlled environment, and therefore, the results that may be obtained in other operating environments may vary significantly. Users of this document should verify the applicable data for their specific environment.

Reference to PTF numbers that have not been released through the normal distribution process does not imply general availability. The purpose of including these reference numbers is to alert IBM customers to specific information relative to the implementation of the PTF when it becomes available to each customer according to the normal IBM PTF distribution process.

The following terms are trademarks of the International Business Machines Corporation in the United States and/or other countries:

AS/400®	AT®
Business Partner	DatagLANce
DB2®	FFST/2
IBM®	Micro Channel®
NetFinity®	NetView®
OPC	OS/2®
OS/2 Warp Server	Presentation Manager®
PS/2®	S/390®
SP	Wake on LAN
WIN-OS/2®	Workplace Shell®
WorkSpace On-Demand	XGA®

The following terms are trademarks of other companies:

Java and HotJava are trademarks of Sun Microsystems, Incorporated.

Microsoft, Windows, Windows NT, and the Windows 95 logo are trademarks or registered trademarks of Microsoft Corporation.

Pentium, MMX, ProShare, LANDesk, and ActionMedia are trademarks or registered trademarks of Intel Corporation in the U.S. and other countries.

Other company, product, and service names may be trademarks or service marks of others.



---

## Appendix I. Related Publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this redbook.

---

### I.1 International Technical Support Organization Publications

For information on ordering these ITSO publications see "How to Get ITSO Redbooks" on page 337.

- *Inside OS/2 LAN Server 4.0*, SG24-4428
- *Getting to Know OS/2 Warp 4*, SG24-4758
- *Prepare for OS/2 Engineer Certification*, SG24-4869
- *OS/2 Workplace Shell Configuration Techniques*, GG24-4201
- *The Guide to OS/2 Warp Device Drivers*, SG24-4627
- *OS/2 Installation Techniques: The DIC Guide*, SG24-4295
- *Inside OS/2 Warp Server, Volume 1, Exploring the Core*, SG24-4602
- *Inside OS/2 Warp Server, Volume 2, System Management*, SG24-4702
- *LAN Problem Determination and Monitoring Using DatagLANce*, SG24-4546

---

### I.2 Redbooks on CD-ROMs

Redbooks are also available on CD-ROMs. **Order a subscription** and receive updates 2-4 times a year at significant savings.

CD-ROM Title	Subscription Number	Collection Kit Number
System/390 Redbooks Collection	SBOF-7201	SK2T-2177
Networking and Systems Management Redbooks Collection	SBOF-7370	SK2T-6022
Transaction Processing and Data Management Redbook	SBOF-7240	SK2T-8038
AS/400 Redbooks Collection	SBOF-7270	SK2T-2849
RS/6000 Redbooks Collection (HTML, BkMgr)	SBOF-7230	SK2T-8040
RS/6000 Redbooks Collection (PostScript)	SBOF-7205	SK2T-8041
Application Development Redbooks Collection	SBOF-7290	SK2T-8037
Personal Systems Redbooks Collection	SBOF-7250	SK2T-8042

---

### I.3 Other Publications

These publications are also relevant as further information sources:

- *Getting to Know OS/2 Warp 4*, ISBN 0-13-842147-1
- *Prepare for OS/2 Engineer Certification*, ISBN 9-655-61119-1

- *OS/2 Warp UNLEASHED*, ISBN 0-672-30545-3
- *IBM Token-Ring Network Remote Program Load Version 1.0*, SK2T-0333
- *OS/2 LAN Server Network Administrator Reference Volume 1: Planning, Installation and Configuration*, S10H-9680
- *OS/2 LAN Server Network Administrator Reference Volume 2: Performance Tuning*, S10H-9681
- *OS/2 LAN Server Network Administrator Reference Volume 3: Network Administrator Tasks*, S10H-9682
- *OS/2 WARP Server Easy Start*, S25H-8003
- *OS/2 WARP Server Up and Running*, S25H-8004
- *OS/2 WARP Server Advanced Easy Start (SMP)*, S28H-0151
- *OS/2 WARP Server Advanced Up and Running (SMP)*, S28H-0152

---

## How to Get ITSO Redbooks

This section explains how both customers and IBM employees can find out about ITSO redbooks, CD-ROMs, workshops, and residencies. A form for ordering books and CD-ROMs is also provided.

This information was current at the time of publication, but is continually subject to change. The latest information may be found at <http://www.redbooks.ibm.com>.

---

## How IBM Employees Can Get ITSO Redbooks

Employees may request ITSO deliverables (redbooks, BookManager BOOKs, and CD-ROMs) and information about redbooks, workshops, and residencies in the following ways:

- **PUBORDER** — to order hardcopies in United States
- **GOPHER link to the Internet** - type GOPHER.WTSCPOK.ITSO.IBM.COM
- **Tools disks**

To get LIST3820s of redbooks, type one of the following commands:

```
TOOLS SENDTO EHONE4 TOOLS2 REDPRINT GET SG24xxxx PACKAGE
TOOLS SENDTO CANVM2 TOOLS REDPRINT GET SG24xxxx PACKAGE (Canadian users only)
```

To get BookManager BOOKs of redbooks, type the following command:

```
TOOLCAT REDBOOKS
```

To get lists of redbooks, type one of the following commands:

```
TOOLS SENDTO USDIST MKTTOOLS MKTTOOLS GET ITSOCAT TXT
TOOLS SENDTO USDIST MKTTOOLS MKTTOOLS GET LISTSERV PACKAGE
```

To register for information on workshops, residencies, and redbooks, type the following command:

```
TOOLS SENDTO WTSCPOK TOOLS ZDISK GET ITSOREGI 1996
```

For a list of product area specialists in the ITSO: type the following command:

```
TOOLS SENDTO WTSCPOK TOOLS ZDISK GET ORGCARD PACKAGE
```

- **Redbooks Web Site on the World Wide Web**  
<http://w3.itso.ibm.com/redbooks>
- **IBM Direct Publications Catalog on the World Wide Web**  
<http://www.elink.ibm.link.ibm.com/pb1/pb1>  
IBM employees may obtain LIST3820s of redbooks from this page.
- **REDBOOKS category on INEWS**
- **Online** — send orders to: USIB6FPL at IBMMAIL or DKIBMBSH at IBMMAIL
- **Internet Listserver**

With an Internet e-mail address, anyone can subscribe to an IBM Announcement Listserver. To initiate the service, send an e-mail note to [announce@webster.ibm.link.ibm.com](mailto:announce@webster.ibm.link.ibm.com) with the keyword subscribe in the body of the note (leave the subject line blank). A category form and detailed instructions will be sent to you.

---

## How Customers Can Get ITSO Redbooks

Customers may request ITSO deliverables (redbooks, BookManager BOOKs, and CD-ROMs) and information about redbooks, workshops, and residencies in the following ways:

- **Online Orders** — send orders to:

	<b>IBMMAIL</b>	<b>Internet</b>
In United States:	usib6fpl at ibmmail	usib6fpl@ibmmail.com
In Canada:	caibmbkz at ibmmail	lmannix@vnet.ibm.com
Outside North America:	dkibmbsh at ibmmail	bookshop@dk.ibm.com

- **Telephone orders**

United States (toll free)	1-800-879-2755
Canada (toll free)	1-800-IBM-4YOU
Outside North America	(long distance charges apply)
(+45) 4810-1320 - Danish	(+45) 4810-1020 - German
(+45) 4810-1420 - Dutch	(+45) 4810-1620 - Italian
(+45) 4810-1540 - English	(+45) 4810-1270 - Norwegian
(+45) 4810-1670 - Finnish	(+45) 4810-1120 - Spanish
(+45) 4810-1220 - French	(+45) 4810-1170 - Swedish

- **Mail Orders** — send orders to:

IBM Publications Publications Customer Support P.O. Box 29570 Raleigh, NC 27626-0570 USA	IBM Publications 144-4th Avenue, S.W. Calgary, Alberta T2P 3N5 Canada	IBM Direct Services Sortemosevej 21 DK-3450 Allerød Denmark
--	--	--

- **Fax** — send orders to:

United States (toll free)	1-800-445-9269
Canada	1-403-267-4455
Outside North America	(+45) 48 14 2207 (long distance charge)

- **1-800-IBM-4FAX (United States) or (+1)001-408-256-5422 (Outside USA)** — ask for:

Index # 4421 Abstracts of new redbooks  
Index # 4422 IBM redbooks  
Index # 4420 Redbooks for last six months

- **Direct Services** - send note to [softwareshop@vnet.ibm.com](mailto:softwareshop@vnet.ibm.com)

- **On the World Wide Web**

Redbooks Web Site	<a href="http://www.redbooks.ibm.com">http://www.redbooks.ibm.com</a>
IBM Direct Publications Catalog	<a href="http://www.elink.ibm.link.ibm.com/pbl/pbl">http://www.elink.ibm.link.ibm.com/pbl/pbl</a>

- **Internet Listserver**

With an Internet e-mail address, anyone can subscribe to an IBM Announcement Listserver. To initiate the service, send an e-mail note to [announce@webster.ibm.link.ibm.com](mailto:announce@webster.ibm.link.ibm.com) with the keyword subscribe in the body of the note (leave the subject line blank).



---

## IBM Redbook Order Form

Please send me the following:

Title	Order Number	Quantity

---

First name	Last name
------------	-----------

---

Company
---------

---

Address
---------

---

City	Postal code	Country
------	-------------	---------

---

Telephone number	Telefax number	VAT number
------------------	----------------	------------

- Invoice to customer number

---

- Credit card number

---

---

Credit card expiration date	Card issued to	Signature
-----------------------------	----------------	-----------

**We accept American Express, Diners, Eurocard, Master Card, and Visa. Payment by credit card not available in all countries. Signature mandatory for credit card payment.**



---

## List of Abbreviations

<b>APA</b>	all points addressable	<b>LAA</b>	Local Administered Address
<b>CID</b>	Configuration, Installation and Distribution	<b>MIPS</b>	Million Instructions Per Second
<b>CLI</b>	Command Line Interface	<b>NCP</b>	NetWare Core Protocol
<b>DLC</b>	Data Link Control	<b>NPT</b>	Non-Programmable Terminal
<b>FIT</b>	File Index Table	<b>PM</b>	Presentation Manager
<b>IBM</b>	International Business Machines Corporation	<b>PROFS</b>	Professional Office System
<b>IS</b>	Information Systems	<b>RIPL</b>	Remote Initial Program Load
<b>ITSO</b>	International Technical Support Organization	<b>UAA</b>	Universal Administered Address



---

## Index

### Special Characters

#include 129

## A

abbreviations 341

Access control profiles 110

acronyms 341

Administration

    automating 216

    basic 98

        client setup 95

    client setup 98

    command line interface 229

    enabling/disabling client workstations 100

    network applications 108

    printer definitions 104

    TCP/IP configuration 105

    user setup 119

Alias 110

APPC support 199

Application parameters 117

Application roaming 15, 108

Active Registration Tool 81

## B

Background Bitmap 125

Backup 73

bibliography 335

Boot Block Definition file 35

BOOT-P 18

Bridging 213

## C

CD-ROM device support 177

Client

    enabling/disabling 100

    swap file 101

Client setup

    access control profiles 107

    Administration GUI 98

Client setup (*continued*)

    automating 229, 238

    CONFIG.SYS 202

    deleting 234

    file creation 107

    network adapter 103

    printer definitions 104

    steps 95

    TCP/IP configuration 105

    video adapter type/resolution 103

COM port 188

Command line interface 229

COMPCFS.CMD 285

CONFIG.SYS 123, 202, 281

## D

DFBB10US.FIT 128

Disk space requirements 61

## E

Enabling Adapters 139, 141

End-user environment 117, 120

End-user environments supported 11

End-user setup

    application parameters 120

    network applications 251

    setup 119

Environment variables 58

## F

Feature Install 74

FFST/2 200

FIT file

    #include statement 129

    combining FITs 129

    default files 38, 127, 128

    description 38

    Machine Classes 156, 169

    machine FIT 40, 107, 127, 128, 202

    user FIT 41, 127

## G

GETRPL 84

## H

Hard disk 60

Hardware prerequisites

disk size 60

disk space 61

memory 59

processor 59

## I

INICOPYAPPLICATION.CMD 284

INICREATEDELTA4.CMD 281

INIMERGE.CMD 283

Installation

Administrator Client 86

planning 11

post-installation procedures 84

server 72

unattended 84, 90, 217

## J

Java Applications 138

Java Runtime 136

## K

Keyboard support 185

## L

Load balancing 69

Local disk access, enabling/disabling 15

Local Printing 126

Logon Bitmap 125

LUA support 199

## M

Machine classes

command line interface 249

components 151

creating your own

CD-ROM device support 177

Machine classes (*continued*)

creating your own (*continued*)

choosing a prototype 155

CONFIG.SYS 169

feasibility tests 153

FIT files 169

INI files 158

keyboard driver 185

methodology 157

mouse support 182

OS2.INI 170

OS2SYS.INI 171

SCSI device support 172

serial device support 188

SYSTEM.INI 171

tools required 155

WIN.INI 171

directory structure 156, 167

File Index Tables 156

modifying 153

optimizing 195

overview 149

querying 249

shipped with WorkSpace On-Demand 152

Machine FIT 40

MAXOPENS 63

MAXSESSOPENS 63

MAXTHREADS 63

Memory requirements 59

Remote IPL 59

WorkSpace On-Demand image 62

Mouse driver 182

## N

NCBS 63

NDISDD.PRO file 42

NET RIPLMACH command 229

Netscape Navigator 133, 135

Network Adapter 139

Network adapter support 53

NDISDD.PRO file 42

Network applications

access control profiles 110

alias 110

compatible with WorkSpace On-Demand 57

end-user environment 117, 120

## Network applications (*continued*)

- environment variables 58
- FFST/2 200
- middleware 196
- migrating 59
- OS/2 Access Feature 199
- parameters 117
- Personal Communication/3270 for OS/2 203
- roaming 108
- setup 108
- shipped with WorkSpace On-Demand 57
- supported by WorkSpace On-Demand 57
- TME 10 NetFinity 206

Network capacity 55

Network infrastructure 211

Network Printing 126

Network resource requirements 62

NUMREQBUF 63

## O

OS/2 Access Feature 199

OS/2 Applications 132

## P

Performance 212

Performance planning 63

Personal Communication/3270 for OS/2 203

Planning 11

- backup and redundancy 69
- load balancing 69
- network capacity issues 55
- network infrastructure 211
- performance 212
- performance considerations 63

PMLOGON 14, 123

Prerequisites 52

- hardware 59
- network adapters 53
- network resources 62
- software 51, 52

Printer access 104

Printing 126

PROTOCOL.INI 142

## R

RAM requirements 59

RDRDEBUG 260

Registration Tool 81, 88

Restricted shell 14

RIPL

- concepts 19
- directory structure 44
- hardware requirements 20
- memory requirements 59
- network resource requirements 62

RPL.MAP file

- client setup 107
- description 26
- server record 31
- workstation record 29

## S

SCSI device support 172

Serial device support 188

Server record (RPL.MAP file) 31

Server redundancy 69

Server Tuning

- MAXOPENS 63
- MAXSESSOPENS 63
- MAXTHREADS 63
- NCBS 63
- NUMREQBUF 63
- SESSIONS 63

SESSIONS 63

Software prerequisites

- administration client 52
- server 51

Supported end-user environments 11

Swap file 101

## T

TCP/IP 264

TCP/IP configuration 105

TME 10 NetFinity 206

## U

Uninstall 73, 90

User FIT 41

User interface

application considerations 58

modifying 14

PMLOGON 14

User setup

application parameters 120

network applications 251

setup 119

## W

WorkSpace On-Demand

components 13

overview 9

Workstation record (RPL.MAP file) 29

WPINSTAL.LOG 83

Error Log

Log

WPINSTAL.LOG 83

WSOD\_LAUNCH\_MINIMIZED 137, 262

WSOD\_LAUNCH\_NOCLOSE 137

WSOD\_LAUNCH\_NODROP 262

WSOD\_LAUNCH\_SESSION 137

WSODERR.LOG 83

Error Log

Log

WSODERR.LOG 83

WSODHST.LOG 83

Error Log

Log

WSODHST.LOG 83



---

## ITSO Redbook Evaluation

WorkSpace On-Demand Handbook  
SG24-2028-00

Your feedback is very important to help us maintain the quality of ITSO redbooks. **Please complete this questionnaire and return it using one of the following methods:**

- Use the online evaluation form found at <http://www.redbooks.com>
- Fax this form to: USA International Access Code + 1 914 432 8264
- Send your comments in an Internet note to [redbook@vnet.ibm.com](mailto:redbook@vnet.ibm.com)

**Please rate your overall satisfaction** with this book using the scale:  
(1 = very good, 2 = good, 3 = average, 4 = poor, 5 = very poor)

Overall Satisfaction \_\_\_\_\_

**Please answer the following questions:**

Was this redbook published in time for your needs? Yes\_\_\_\_ No\_\_\_\_

If no, please explain:

---

---

---

---

What other redbooks would you like to see published?

---

---

---

**Comments/Suggestions:** ( THANK YOU FOR YOUR FEEDBACK! )

---

---

---

---

---



Printed in U.S.A.

SG24-2028-00

