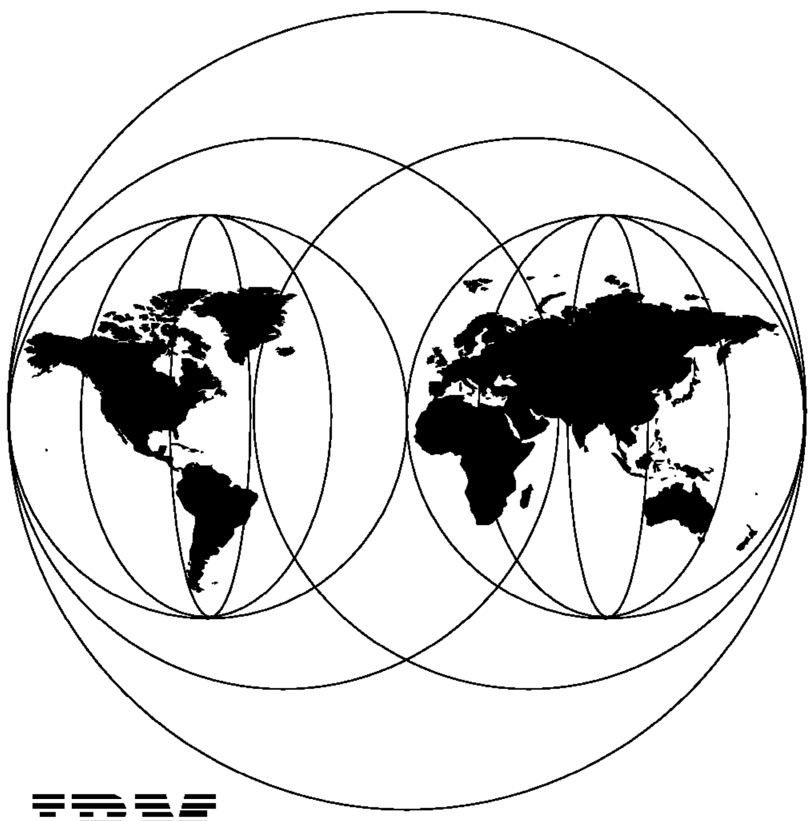
SG24-4270-01

TCP/IP Solutions for VSE/ESA with OpenConnect Systems Software

August 1996



International Technical Support Organization
Boeblingen Center

International Technical Support Organization

SG24-4270-01

TCP/IP Solutions for VSE/ESA with OpenConnect Systems Software

August 1996

Take Note!

Before using this information and the product it supports, be sure to read the general information in Appendix B, "Special Notices" on page 353.

First Edition (August 1996)

This edition applies to:

- Version 2.2.4.1 of OCS TCP/IP for VSE/ESA FTP Server from OpenConnect Systems for use with the VSE/ESA Operating System
- Version 2.2.4.1 of OCS TCP/IP for VSE/ESA FTP Client from OpenConnect Systems for use with the VSE/ESA Operating System
- Version 2.1.5 of OCS TCP/IP for VSE/ESA RSH Client from OpenConnect Systems for use with the VSE/ESA Operating System
- Version 4.11 of OCS TCP/IP for VSE/ESA Telnet Client Full Screen from OpenConnect Systems for use with the VSE/ESA Operating System
- Version 2.1 of OCS TCP/IP for VSE/ESA Socket Access Method from OpenConnect Systems for use with the VSE/ESA Operating System
- Version 3.7.5 of OpenConnect Server II (OCS II) and TELNET Server 3270 from OpenConnect Systems for use with the AIX Operating System
- Version 1.3 of OpenConnection for Channel (OCC) from OpenConnect Systems
- Version 1.1.0 of OCS/Line Printer Deamon (LPD) from OpenConnect Systems
- Version 1.2 of OC/Print Server for AIX from OpenConnect Systems

Comments may be addressed to:
IBM Corporation, International Technical Support Organization
Dept. 3222 Building 71032-02
Postfach 1380
71032 Böblingen, Germany

Contents

Figures . . . ix

Tables . . . xv

Preface . . . xvii
How This Redbook Is Organized . . . xvii
The Team that Wrote this Redbook . . . xix
Comments Welcome . . . xx

Part 1. Introduction . . . 1

Chapter 1. Introduction to TCP/IP . . . 3

Chapter 2. VSE/ESA and TCP/IP . . . 5
2.1 OCS TCP/IP for VSE . . . 5
2.1.1 OC/Telnet . . . 6
2.1.2 OC/FTP Client . . . 6
2.1.3 OC/FTP Server . . . 7
2.1.4 OC/RSH Client . . . 7
2.1.5 OC/SAM . . . 8
2.1.6 OC/Line Printer Daemon (LPD) . . . 8
2.1.7 OCS II Gateway . . . 8
2.1.8 OCS Print Server for AIX . . . 8
2.2 OpenConnection for Channel (OCC) . . . 8
2.3 Hardware and Software Requirements for OCS TCP/IP . . . 9
2.3.1 Hardware Requirements . . . 9
2.3.2 Software Requirements . . . 10
2.3.3 OCC Hardware . . . 11

Part 2. OCS TCP/IP for VSE Installation and Implementation . . . 13

Chapter 3. OCS Software Functional Overview . . . 15
3.1 OCS II Gateway Functional Overview . . . 16
3.2 OC/FTP Server Functional Overview . . . 18
3.3 OC/FTP Client Functional Overview . . . 20
3.4 OC/Line Printer Daemon Functional Overview . . . 22
3.5 OC/RSH Client Functional Overview . . . 24
3.6 OC/TELNET FS Functional Overview . . . 26
3.7 OC/SAM Functional Overview . . . 28
3.8 OCS Software Functional Overview . . . 30

Chapter 4. Sample Environment . . . 31
4.1 Hardware . . . 31
4.2 Software . . . 32
4.3 Network Configuration . . . 33
4.3.1 OCS II Gateway Attachment Configurations . . . 35
4.3.2 IP Address and Name Assignments . . . 35
4.4 Router Functions . . . 36
4.5 PC/DOS Windows Client in our Test Environment . . . 38

Chapter 5. Defining the OCS II Gateway to the Host 41
5.1 VM/VSE with OCS II Gateway Attached via the IBM 3174 41
5.1.1 Customization in VM/ESA 44
5.1.2 Customization in VM VTAM 45
5.1.3 Customization in VSE/ESA 48
5.1.4 Customization in VSE VTAM 48
5.2 VSE/ESA with OCS II Gateway Attached via the Token-Ring Adapter . . . 50
5.2.1 Customization of IBM 9221 Token-Ring Adapter 52
5.2.2 Customization in VSE/ESA 52
5.2.3 Customization in VSE VTAM 53
5.3 VSE/ESA with OSCII Gateway Attached via the OCC 56
5.3.1 Customization of OpenConnection for Channel 58
5.3.2 Customization in VSE/ESA 58
5.3.3 Customization in VSE VTAM 58

Chapter 6. OC/FTP Server Installation and Customization 61
6.1 OC/FTP Server Installation 61
6.1.1 Prepare the Installation 61
6.1.2 Installation Procedures 63
6.2 OC/FTP Server Customization 64
6.2.1 VSAM Customization 64
6.2.2 VTAM Customization 66
6.2.3 FTP Server Customization 67
6.2.4 Startup Customization 73
6.2.5 Customizing the Translation Tables 77
6.2.6 User Exits 79

Chapter 7. OC/FTP Client Installation and Customization 81
7.1 OC/FTP Client Installation 81
7.1.1 Installation Preparation 81
7.1.2 Installation Procedures 82
7.2 OC/FTP Client Customization 82
7.2.1 VSAM Customization 82
7.2.2 VTAM Customization 83
7.2.3 CICS Customization 88
7.2.4 FTP Client Customization 92
7.2.5 Startup Customization 102
7.2.6 Customizing the OC/FTP Client Translation Tables 105

Chapter 8. OC/Line Printer Daemon Installation and Customization 109
8.1 OC/Line Printer Daemon Installation 109
8.1.1 Installation Preparation 109
8.1.2 Installation Procedures 110
8.2 OC/Line Printer Daemon Customization 110
8.2.1 VTAM Customization 110
8.2.2 LPD Customization 112
8.2.3 Startup Customization 115
8.2.4 Customizing the OC/Line Printer Daemon Translation Tables . . . 118
8.2.5 Customizing the OC/Line Printer Daemon Print Banner 119
8.2.6 Queue Definition in AIX 121

Chapter 9. OC/RSH Client Installation and Customization 123
9.1 OC/RSH Client Installation 123
9.1.1 Installation Preparation 123
9.1.2 Installation Procedures 123

9.2 OC/RSH Client Customization . . . 124
9.2.1 VTAM Customization . . . 124
9.2.2 CICS Customization . . . 126
9.2.3 RSH Client Customization . . . 128
9.2.4 Prepare for Remote Command Execution Access at an AIX Host . 128
9.2.5 Startup Customization . . . 129

Chapter 10. OC/TELNET FS Installation and Customization . . . 133
10.1 OC/TELNET FS Installation . . . 133
10.1.1 Installation Preparation . . . 133
10.1.2 Installation Procedures . . . 133
10.2 OC/TELNET FS Customization . . . 134
10.2.1 VTAM Customization . . . 134
10.2.2 TELNET Client Customization . . . 137
10.2.3 Startup Customization . . . 143

Chapter 11. OC/SAM Installation and Customization . . . 145
11.1 OC/SAM Installation . . . 145
11.1.1 Installation Preparation . . . 145
11.1.2 Installation Procedures . . . 145
11.1.3 Installation Verification . . . 149
11.2 OC/SAM Customization . . . 150
11.2.1 GET SERVICE BY NAME Table Customization . . . 150

Chapter 12. OCS II Gateway Installation and Customization . . . 153
12.1 OCS II Gateway Overview . . . 153
12.2 Preinstallation Tasks . . . 153
12.3 Installation . . . 154
12.4 Customization . . . 156
12.4.1 Local Area Network (TCP/IP) Configuration . . . 160
12.4.2 SNA Level Data Link Configuration . . . 163
12.4.3 Physical/Logical Unit Configuration . . . 164
12.4.4 System Wide Configuration Options . . . 168
12.4.5 OCS II RISC/6000 Data Link Configuration . . . 174
12.4.6 Generate a Configuration Listing . . . 178
12.4.7 Load Configuration File Set . . . 182
12.4.8 Save Configuration File Set . . . 183
12.5 Special Considerations . . . 184
12.5.1 TCP/IP Node Considerations . . . 184
12.5.2 Host Access Table (HAT) . . . 185
12.6 Set up Alternative Ports for AIX TELNET and FTP Servers . . . 186

Chapter 13. OCS II Gateway Operation . . . 189
13.1 Operation Samples (start,stop,status....) . . . 189

Chapter 14. OCS Print Server Installation and Customization . . . 193
14.1 OCS Print Server Overview . . . 193
14.2 OCS Print Server Installation . . . 193
14.3 OCS Print Server Customization . . . 195
14.3.1 Customization of the OCS Print Server . . . 196
14.3.2 Customization of the OCS II Gateway for OCS Print Server . . . 200
14.3.3 Customization in VSE/ESA . . . 200
14.3.4 VTAM Customization . . . 200
14.3.5 CICS Customization . . . 202

Chapter 15. OCS Print Server Operation 205
15.1 Operation Samples (start,stop,status....) 205
15.2 Printing Files from CICS Report Controller 207

Chapter 16. OpenConnection for Channel Installation and Customization . . 209
16.1 Installation Summary 209
16.1.1 RISC/6000 Host Software Installation 209
16.1.2 Installing OCC Hardware 210
16.1.3 Configuring OCC Hardware 210
16.1.4 OCC Customization 212
16.1.5 Configure the SNA Host for OCS II Gateway 215

Chapter 17. OpenConnection for Channel Operation 217
17.1 OCC Operation Samples 217
17.1.1 Starting the OCC Administration Daemon 217

Chapter 18. OC/FTP Server Operation and Examples 219
18.1 Operation 219
18.1.1 VSE Host (FTP Server) 219
18.1.2 TCP/IP FTP Clients 220
18.2 Operation Samples 226
18.2.1 Establish Connection to the OC/FTP Server 226
18.2.2 Using SITE 227
18.2.3 Switching/Displaying Libraries (Librarian) 228
18.2.4 Switching/Displaying VSAM Data Sets 230
18.2.5 Accessing POWER 231
18.2.6 Submitting a Job to VSE/ESA 231
18.2.7 Sending Print Lists to VSE/ESA 233
18.2.8 Receiving Print Lists from VSE/ESA 234
18.2.9 Transferring Files to/from VSE/ESA 236

Chapter 19. OC/FTP Client Operation and Examples 239
19.1 FTP Client Command Summary 239
19.2 Batch Operation Samples 245
19.2.1 Get a File from PC/DOS to VSE Library 245
19.2.2 Send a VSE/VSAM KSDS File to AIX 248
19.2.3 Sending a VSE/VSAM SAM File to AIX 253
19.2.4 Remote Directory Operations 259
19.3 Online Operation Samples 262
19.3.1 Obtaining Multiple Files from AIX 262
19.3.2 Transferring Multiple Files to AIX 266
19.3.3 Moving an AIX Data File into a VSAM KSDS 268
19.3.4 Transferring an AIX Data File into a VSAM SAM File 271

Chapter 20. OC/Line Printer Daemon Operation and Examples 273
20.1 Operation 273
20.1.1 VSE Host (Line Printer Server) 273
20.1.2 Displaying the Status of OC/Line Printer Daemon Queues 274
20.1.3 Printing from Remote TCP/IP Clients 275
20.1.4 Operation Examples 277

Chapter 21. OC/RSH Client Operation and Examples 281
21.1 OC/RSH Client Online Operation Examples 282
21.2 Batch Operation Samples 284
21.2.1 RSH Batch Example 284

Chapter 22. OC/TELNET FS Operation and Examples . . . 287
22.1 Operation Samples . . . 287
22.2 Working with Telnet Panels . . . 288

Chapter 23. OCS II Telnet Server Operation and Examples . . . 295
23.1 General Considerations . . . 295
23.2 Operation Examples . . . 295
23.2.1 Telnet Server Welcome Panel . . . 296
23.2.2 Telnet Server Commands Panel . . . 297
23.2.3 Display OCS II Telnet Server Clients . . . 299
23.2.4 Display OCS II Telnet Server Configuration . . . 299
23.2.5 Display OCS II Telnet Server HAT Table . . . 299
23.2.6 Display OCS II Telnet Server Keycodes . . . 300
23.2.7 Display OCS II Telnet Server ′trace′ Facility . . . 300
23.2.8 Display OCS II Telnet Server Services . . . 300
23.2.9 Display OCS II Telnet Server Terminal Types . . . 301
23.2.10 Display OCS II Telnet Server Status . . . 301
23.2.11 Display Terminal Characteristics . . . 302
23.2.12 OCS II Telnet Server Help Information . . . 302

Chapter 24. OC/SAM Programming . . . 305
24.1 TCP/IP Application Programming Tutorial . . . 305
24.1.1 Network Programming Overview . . . 305
24.1.2 Socket Programming . . . 305
24.1.3 Socket Addresses . . . 306
24.1.4 Application Protocols . . . 306
24.2 Running the Sample Programs . . . 310
24.2.1 VTAM Customization . . . 310
24.2.2 OCS Sample Programs . . . 311
24.3 SMTP Mail Example . . . 316
24.3.1 Mail Example Overview . . . 316
24.3.2 Message Transfer Agent . . . 317
24.3.3 SMTP Server . . . 318
24.3.4 Receiving the Mail . . . 320
24.3.5 Message Transfer Agent Program Implementation and Setup . . . 321

Chapter 25. Implementation Summary . . . 325
25.1 VTAM and OCS II Gateway Definitions . . . 326
25.2 OC/FTP Server Definitions . . . 327
25.3 OC/FTP Client, OC/RSH Client and OC/LPD Definitions . . . 328
25.3.1 VTAM and CICS Definitions . . . 328
25.3.2 VTAM, OC/FTP Client and OCS II Gateway Definitions . . . 329
25.4 OC/TELNET FS Definitions . . . 330
25.5 Issues and Concerns . . . 331

Part 3. Summary . . . 333

Chapter 26. OpenConnect Systems Products Functions and Limitations Summary . . . 335
26.1 OpenConnect Systems Products Functions Summary . . . 335
26.1.1 File Transfer Functions . . . 335
26.1.2 Remote Command Execution . . . 335
26.1.3 Socket API . . . 336
26.1.4 Line Printer Daemon Functions . . . 336

26.1.5 TELNET Functions 336
26.2 Additional OpenConnect Systems Products Summary 336
26.2.1 OpenConnection for Channel Usage 336
26.2.2 OCS Print Server for AIX 336
26.2.3 OC://WebConnect Overview 336
26.3 OpenConnect Systems Products Limitations 337

Part 4. Job Streams and Definition Examples 339

Appendix A. Job Streams and Definition Examples for OCS Products 341
A.1 Product Installation Job Streams 341
A.2 VSE VTAM Start Options List 341
A.3 VSE TCPSW.B Switched Major Node 342
A.4 VTAM APPL Major Node for OCS Products 344
A.5 VTAM USSTAB 345
A.6 OCSBIND MODETAB for OCS Products 351

Appendix B. Special Notices 353

Appendix C. Related Publications 357
C.1 International Technical Support Organization Publications 357
C.2 Other Publications 357

How To Get ITSO Redbooks 359
How IBM Employees Can Get ITSO Redbooks 359
How Customers Can Get ITSO Redbooks 360
IBM Redbook Order Form 361

Index 363

Figures

1. Comparison of the OSI and the TCP/IP Model 4
2. OCS TCP/IP for VSE Overview 6
3. OCC Environment 11
4. OCS II Gateway Functions 16
5. OCS II Gateway TELNET Server Functions 18
6. OC/FTP Server Functions 19
7. OC/FTP Client Functions 21
8. OC/Line Printer Daemon Functions 23
9. OC/RSH Client Functions 25
10. OC/TELNET FS Functions 27
11. OC/SAM Functions 29
12. OCS Software Functional Overview 30
13. OCS TCP/IP Network Diagram 34
14. OCS TCP/IP Network Logical View 37
15. IBM TCP/IP Windows Applications 38
16. Telnet Windows Application on PC/DOS 38
17. FTP Windows Applications on PC/DOS 39
18. Mail Windows Application on PC/DOS 40
19. Configuration and Definition Summary for 3174 Connection 43
20. Device Address Definitions in SYSTEM CONFIG File 44
21. VM VTAM Path Table 45
22. VM VTAM CA Major Node for VCTCA 45
23. VM VTAM CDRM Major Node 46
24. VM VTAM LOCAL SNA Major Node for IBM 3174 and RISC/6000 47
25. VSE VTAM Path Table 48
26. VSE VTAM CA Major Node VCTCA 49
27. VSE VTAM CDRM Major Node 49
28. VSE VTAM CDRSC Major Node for OCS II Gateway 50
29. Configuration and Definition Summary for the TRA Connection 51
30. VSE VTAM LAN Major Node 53
31. VSE VTAM Switched Major Node Definition 55
32. Configuration and Definition Summary for the OCC Connection 57
33. PU Definition for the OCC Connection 59
34. VSE/ESA Library and Sublibraries for OCS Products 62
35. Sample Job to Scan Product Tapes 63
36. Sample Retrace Product Job 64
37. ESDS, SAM and KSDS Model Definitions for OC/FTP Server 65
38. MODETAB Entry for OC/FTP Server 66
39. VTAM APPL Statement for OC/FTP Server 66
40. Logical Unit Definition for OC/FTP Server 67
41. Parameter Library DLBL 67
42. CONFIG Member for OC/FTP Server 68
43. SITE Member for OC/FTP Server 70
44. PASSWD Member for OC/FTP Server 72
45. Build the Executable Phases for OC/FTP Server 74
45. Build the Executable Phases for OC/FTP Server 74
46. Site Protection Job for OC/FTP Server 76
47. OC/FTP Server Startup Job 76
48. Assemble and Link-edit NEWXLTE Translation Table 78
49. MODETAB Entries for OC/FTP Client 83
50. OC/FTP Client Applications and LU Relationships 85

51. VTAM APPL Statements for OC/FTP Client 86
52. Logical Unit Definitions for OC/FTP Client 87
53. DYNALL Member for OC/FTP Client 93
54. FAPPL Member for OC/FTP Client 95
55. Definition and Relationships between NETWORKS and LU List Members 96
56. NETWORKS Member for OC/FTP Client 97
57. NETWORK LU List Member for IPFNET NETWORKS Member 97
58. NETWORK LU List Member for SUNNET NETWORKS Member 98
59. NETWORK LU List Member for AIXNET NETWORKS Member 98
60. HOSTS Member for OC/FTP Client 99
61. CACHE Member for OC/FTP Client 100
62. PACING Member for OC/FTP Client 100
63. NETMASK Member for OC/FTP Client 101
64. RESOLVER Member for OC/FTP Client 102
65. Build the Executable Phases for OC/FTP Client 103
66. Site Protection Job for OC/FTP Client 104
67. OC/FTP Client Startup Job 104
68. Sample OC/FTP Client Batch Job 105
69. Job to Catalog Translation Table Source Code 106
70. Job to Assemble and Catalog the Translation Object Module 107
71. VTAM APPL Statement for OC/Line Printer Daemon Receive Connection 111
72. Logical Unit Definitions for OC/Line Printer Daemon 112
73. LPDCONF Member for OC/Line Printer Daemon 113
74. Build the Executable Phases for OC/Line Printer Daemon and Apply Site Protection 116
75. OC/Line Printer Daemon Startup Job 117
76. Job to Catalog Translation Table Source Code 119
77. Job to Catalog Print Banner Source Code 120
78. OC/Line Printer Daemon Queue Definition on OCS II Gateway 121
79. OC/RSH Client Operation 126
80. Sample /etc/hosts.equiv File 129
81. Sample $HOME/.rhosts File for User wack 129
82. Build the Executable Phases for OC/RSH Client 129
83. Site Protection Job for OC/RSH Client 130
84. OC/RSH Client Startup Job 130
85. Sample OC/RSH Client Batch Job 131
86. MODETAB Entry for OC/TELNET FS 134
87. APPL Definition Sample for OC/TELNET FS 135
88. Logical Unit Definitions for OC/TELNET FS 136
89. NETWORKS File for OC/TELNET FS 138
90. IPFNET LU List File for OC/TELNET FS 138
91. HOSTS File for OC/TELNET FS 139
92. SYSIN File for OC/TELNET FS 139
93. Sample User Command File 140
94. DFLTSYS File for OC/TELNET FS 141
95. DFLTUSR File for OC/TELNET FS 141
96. SYSCMDS File for OC/TELNET FS 142
97. OC/TELNET FS Startup Job 143
98. Sample Job to Scan Library Backup Tape 146
99. Restore VSE Library from Tape 146
100. New Name Specification 147
101. Tape Information 147
102. Tape Specification 148
103. OC/SAM Library Restore Job 148

104. Copy Librarian Member into ICCF Library 149
105. Catalog ICCF Member into Sublibrary 149
106. Sample RUNIVP Job 150
107. Add Entry to 'GET SERVICE BY NAME' Table 151
108. Sample VSETABJ Job 152
109. Installation Exec Main Menu 154
110. OCS and OCC Directories Contents 155
111. Sample Entry for an Instance File 157
112. Configuration Main Menu 158
113. Configure OCSNA Instance Menu 158
114. Instance Configuration Menu 159
115. LAN Configuration Menu 160
116. Internet Address Configuration Menu 161
117. SMIT TCP/IP Configuration Panel 162
118. TCP/IP Keepalive Option Menu 162
119. SNA DLC Configuration Menu 163
120. PU/LU Configuration Menu 164
121. Physical Unit Configuration 165
122. Logical Unit Configuration 166
123. Logical Unit Configuration 167
124. System Wide Options Configuration Menu 168
125. Miscellaneous System Wide Parameters 169
126. TN and TELNET Server Configuration 170
127. TELNET Server Configuration 171
128. TN 3270 Server Configuration 172
129. TELNET 3270 Server Configuration 173
130. PU T2.1 Node Configuration 173
131. Instance Configuration Menu 174
132. RISC/6000 Link Configuration 175
133. RISC/6000 Token Ring Data Link Configuration 175
134. Token-Ring Configuration 177
135. AIX SMIT Token-Ring Adapter Configuration Panel 178
136. Configuration Listing 179
137. OCC Configuration Listing 181
138. Load Instance Files 183
139. Save Configuration Updates 184
140. Sample Host Access Table (HAT) 'tstext' 185
141. Sample Host Access Table (HAT) 'octermcap' 186
142. Sample for TCP/IP Service Port Number Assignments 187
143. OCSNA Command Sample 189
144. AIX Inittab Sample for OCSNA Auto Startup 190
145. SNA_STATUS Command Sample 191
146. OCSNA Shutdown Sample 191
147. Defaults File Used in Our Installation 196
148. Config File Used in Our Installation 198
149. OCS Print Server Startup Shell Script 200
150. Logmode Entries Used for OCS Print Server 201
151. LU Used for OCS Print Server 201
152. OCS Print Server Startup Sample 205
153. eps_status -f Sample Output 206
154. Stopping the Licence Manager 207
155. PSTART DEV Sample Console Output 207
156. PSTART DEV Sample Console Output 207
157. OCC Installation Exec Main Menu 210
158. Logical Unit Configuration 213

159. OCC Configurator Main Menu 214
160. Configuration File sca.ini 215
161. Control Unit Definitions 216
162. Startsca Script Sample 217
163. SCAAD and OCSNA Startup Sample 218
164. SCAAD and OCSNA Shutdown Sample 218
165. OC/FTP Server Startup Sample 219
166. Console Log of Status Operation Command 220
167. OC/FTP Server Logon Sample 226
168. Quote SITE Command Sample 227
169. Quote SITE Verify Command Sample 228
170. CD and DIR Librarian Command Samples 229
171. CD and DIR VSAM Command Samples 230
172. POWER Command Sample 231
173. Submitting a Job to VSE/ESA POWER from AIX 232
174. VSE/ESA Console Log after Submitting a Job from AIX 232
175. Sending a Print List to VSE/ESA POWER from AIX 233
176. VSE/ESA Console Log after Sending a Print List from AIX 233
177. Receiving Print Lists from VSE/ESA POWER 235
178. Transfering Files to/from VSE/ESA from an AIX FTP Client 236
179. VSE/ESA FTP Client Batch Job Example: Get a File from Windows . . 246
180. FTP Client Batch Job Example Output: Get a File from Windows 246
181. Define and Initialize a VSAM KSDS File 249
182. Load and Print the VSAM KSDS File 250
183. Job to Transfer a VSAM KSDS File to AIX 251
184. VSE/ESA FTP Client Batch Example: Send a VSAM KSDS File to AIX . 252
185. Define and Initialize a VSAM SAM File 254
186. Load and Print the VSAM SAM File 254
187. Load and Print VSAM SAM File Output 255
188. Send a VSAM SAM File from a VSE/ESA FTP Client to AIX 256
189. Output of FTP Client Send VSAM SAM File to AIX Operation 257
190. VSE/ESA OC/FTP Client Batch Job for OS/2 2.1 Directory Operations . 259
191. Output from OC/FTP Client Batch Job for OS/2 2.1 Directory Operations 260
192. OC/FTP Client Startup Sample 262
193. DEFAULT Command Sample 264
194. MGET Command Sample 265
195. DEFAULT Command Sample 267
196. MPUT Command Sample 267
197. Console Log of MPUT Command 268
198. Transferring a File from AIX into a VSAM KSDS 269
199. Display VSAM KSDS after FTP File Transfer 270
200. Transferring a File from AIX into a VSAM SAM 271
201. Display VSAM SAM after FTP File Transfer 272
202. OC/Line Printer Daemon Startup Sample 273
203. Stopping OC/Line Printer Daemon by Cancelling the Partition 273
204. lpstat Sample Output 274
205. LPQ Sample Output 275
206. LPR Sample Output 275
207. Test File with Standard ASCII Characters 32-126 (X′20-X′7E). 277
208. VSAM Print Output of LPDFILE 279
209. E-mail Notification from OC/Line Printer Daemon 279
210. OC/RSH Client Startup Sample 282
211. List AIX Directory Using OC/RSH Client 283
212. File Manipulation Sample 283

213. RSH Batch Example 284
214. RSH Batch Example Output Listing 284
215. OC/TELNET FS Startup Job Sample 287
216. TELNET Primary Menu 288
217. Terminal Setup Sample 289
218. Translation Table Sample 291
219. Keyboard Definition Sample 292
220. Host Selection Sample 293
221. AIX Welcome Menu Sample 294
222. OC/TELNET FS AIX Emulation Sample 294
223. Example of Connecting to the OCS II Telnet Server 295
224. OCS II Telnet Server Welcome Panel 296
225. OCS II Telnet Server Commands 297
226. Display Clients Example 299
227. Display Configuration Example 299
228. Display HAT Table Example 299
229. Display Keycodes Example 300
230. Display ′trace′ Example 300
231. Display Services Example 300
232. Display Terminal Types Example 301
233. Status Display Example 301
234. Display Terminal Characteristics Example 302
235. Telnet Server Sample 303
236. Telnet Server Help Sample 303
237. Typical Sockets API Functions for UDP 307
238. Typical Sockets API Functions for TCP 309
239. APPL Definitions for OC/SAM Samples 310
240. LU Definitions for the OC/SAM Samples 311
241. Assemble and Run the UDP Server Program 312
242. Assemble and Run the UDP Client Program 313
243. Assemble and Run the TCP Server 314
244. Assemble and Run the TCP Client 314
245. Assemble and Run the TCPSMTP Mail Client 315
246. VSE/ESA SMTP Mail Client Structure 316
247. Entering SMTP Mail on VSE/ESA 317
248. Message Input to the VSE/ESA Message Transfer Agent 318
249. Layout of VSE MTA Control Buffer 318
250. Communication between the VSE/ESA MTA Client and the SMTP Server 320
251. Receiving VSE/ESA SMTP Mail Using OS/2 LaMail 321
252. Reading VSE/ESA SMTP Mail Using OS/2 LaMail 321
253. Assemble Message Transfer Agent Program IESMTA 323
254. SMTP Entry in Default ′GET SERVICE BY NAME′ Table 324
255. OCS II Gateway and VTAM Definitions 326
256. OC/FTP Server Definitions Summary 327
257. OC/FTP Client and OC/RSH Client Definitions Summary - VTAM and CICS 328
258. OC/FTP Client Definitions Summary - VTAM, FTP Client and OCS II Gateway 329
259. OC/TELNET FS Definitions Summary 330

Tables

1. OCS TCP/IP for VSE Network Connection Options 9
2. OCS TCP/IP for VSE RISC/6000 Software Requirements 10
3. IP Address and Name Assignment for the OCS Implementation 36
4. Routing Table for TCP/IP Nodes 37
5. SITE Member Parameters and Default Values 70
6. LPD Queue Definitions used in our Environment 115
7. Default File Parameters and Defaults 197
8. Config File Parameters and Defaults 199
9. OC/FTP Server Host Side Commands 220
10. OC/FTP Server Client Site Commands Summary 222
11. OC/FTP Server CD and DIR Commands 225
12. OC/FTP Client Commands Summary 239
13. OC/FTP Client Job Commands 262
14. OC/RSH Client Job Commands 282

Preface

This redbook describes how VSE/ESA users can participate in Transmission Control Protocol/Internet Protocol (TCP/IP) networks using the OpenConnect Systems (OCS) gateway solution for VSE/ESA. This product provides connectivity between the TCP/IP network environment and the IBM SNA environment, including bi-directional file transfer services, terminal emulation, and printing.

We guide you through all steps required to install, customize and start using the TCP/IP gateway solution for VSE/ESA, covering the VSE/ESA host system applications and the RISC/6000 gateway software.

This document is written for system engineers or programmers responsible for implementing VSE/ESA host integration into TCP/IP networks. It assumes a basic knowledge of TCP/IP concepts and protocols and a working knowledge of VM/ESA, VSE/ESA, ACF/VTAM and AIX.

How This Redbook Is Organized

The redbook is organized as follows:

1. Part 1, ″Introduction″
   - Chapter 1, "Introduction to TCP/IP"

     This chapter contains a very brief introduction to TCP/IP.
   - Chapter 2, "VSE/ESA and TCP/IP"

     This chapter provides an overview of the TCP/IP solution for VSE discussed in this document, OCS TCP/IP for VSE.
2. Part 2, ″OCS TCP/IP for VSE Installation and Implementation″
   - Chapter 3, "OCS Software Functional Overview"

     This chapter contains a functional overview of the OCS TCP/IP for VSE software.
   - Chapter 4, "Sample Environment"

     This chapter describes the laboratory environment at the ITSO Center Boeblingen where OCS TCP/IP for VSE was installed and tested.
   - Chapter 5, "Defining the OCS II Gateway to the Host"

     This chapter discusses the preparation steps necessary to connect the VSE host and the OCS II Gateway to each other.
   - Chapter 6, "OC/FTP Server Installation and Customization"

     This chapter provides detailed information on how to install and customize the OpenConnect/File Transfer Program Server software.
   - Chapter 7, "OC/FTP Client Installation and Customization"

     This chapter provides detailed information on how to install and customize the OpenConnect/File Transfer Program Client software.
   - Chapter 8, "OC/Line Printer Daemon Installation and Customization"

This chapter provides information on how to install and customize the Line Printer Daemon.

- Chapter 9, “OC/RSH Client Installation and Customization”

  This chapter provides detailed information on how to install and customize the OpenConnect/Remote Shell Client software.

- Chapter 10, “OC/TELNET FS Installation and Customization”

  This chapter provides detailed information on how to install and customize the OCS II Telnet Server 3270 software.

- Chapter 11, “OC/SAM Installation and Customization”

  This chapter provides detailed information on how to install and customize the OpenConnect/Socket Access Method software.

- Chapter 12, “OCS II Gateway Installation and Customization”

  This chapter provides detailed information on how to install and customize the OCS II Gateway software on the RISC/6000.

- Chapter 14, “OCS Print Server Installation and Customization”

  This chapter provides detailed information on how to install and customize the OCS Print Server.

- Chapter 13, “OCS II Gateway Operation”

  This chapter shows how to start and stop the OCS II Gateway.

- Chapter 15, “OCS Print Server Operation”

  This chapter shows how to start and stop the OCS Print Server.

- Chapter 16, “OpenConnection for Channel Installation and Customization”

  This chapter describes the OpenConnection for Channel installation and customization.

- Chapter 20, “OC/Line Printer Daemon Operation and Examples”

  This chapter includes information on the OC/Line Printer Daemon operation and examples.

- Chapter 17, “OpenConnection for Channel Operation”

  This chapter discusses the OpenConnection for Channel operation.

- Chapter 18, “OC/FTP Server Operation and Examples”

  This chapter contains examples of how to use the OpenConnect/File Transfer Program Server software.

- Chapter 19, “OC/FTP Client Operation and Examples”

  This chapter contains examples of how to use the OpenConnect/File Transfer Program Client software.

- Chapter 21, “OC/RSH Client Operation and Examples”

  This chapter contains examples of how to use the OpenConnect/Remote Shell Client software.

- Chapter 22, “OC/TELNET FS Operation and Examples”

  This chapter provides operation and test examples for OpenConnect/TELNET Client Full Screen software.

- Chapter 23, “OCS II Telnet Server Operation and Examples”

  This chapter provides operation and test examples for OCS II Telnet Server 3270 software.

- Chapter 24, “OC/SAM Programming”

  This chapter provides general information about writing OpenConnect/Socket Access Method application programs for VSE and contains OpenConnect/Socket Access Method examples.

- Chapter 25, “Implementation Summary”

  This chapter provides summary charts showing the relationships of the interrelated parameters in the subsystems affected.

3. Part 3, ″Summary″

   - Chapter 26, “OpenConnect Systems Products Functions and Limitations Summary”

     This chapter summarizes the functions of the OCS TCP/IP for VSE solution and lists its limitations from a user point of view.

4. Part 4, ″Job Streams and Definition Examples″

   This part provides jobstreams and definitions used during implementation of OCS TCP/IP for VSE.

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 Böblingen Center.

The advisors and authors for the *first* version were:

Werner Stieber, from the International Technical Support Organization Böblingen Center, was the project leader.

Francesco Camilucci (75819783 at ITHVM06), from IBM Italy

Pramot Hemstapat, from IBM Thailand

Klaus Maruschka (MARUSCH at BOEVM3), from IBM Deutschland Entwicklung GmbH, Germany

Klaus-Dieter Wacker (WACKER at BOEVM3), from IBM Deutschland Entwicklung GmbH, Germany

Johnny Yiu (JYIU at CANVM2), from IBM Canada

The authors of the *second* edition of this document were:

Stefanie Grimme (STEFFI at BOEVM4), from the International Technical Support Organization Böblingen Center, was the project leader.

Dirk Breidt (BREIDT at BOEVM4), from IBM Deutschland Entwicklung GmbH, Germany

Leif-Erik Kristiansen, from IBM Deutschland, Germany

Comments Welcome

We want our redbooks to be as helpful as possible. Should you have any comments about this or other redbooks, please send us a note at the following address:

redbook@vnet.ibm.com

Your comments are important to us!

Part 1. Introduction

This document describes the installation and customization aspects of OpenConnect Systems (OCS) TCP/IP for VSE, a product providing access to TCP/IP networks for VSE/ESA environments. The product uses a ″gateway″ approach connecting to the TCP/IP network on one side and to the SNA network on the other. OCS TCP/IP for VSE uses a RISC/6000 as gateway processor.

This part provides:

- a brief introduction to TCP/IP
- a general description of OCS TCP/IP for VSE

Chapter 1. Introduction to TCP/IP

Transmission Control Protocol/Internet Protocol (named TCP/IP after its two main standards) is a suite of protocols that can be used to communicate across any set of interconnected networks called ″internets″. The word ″internet″ must not be confused with the word ″Internet″ (capital I). The Internet is a special internet connecting several million computers in many different countries.

Resulting from research funded by the Department of Defense (DoD), TCP/IP has gained increasing importance since its first implementation around 1980.

The TCP/IP development began in 1971 by the ′Defense Advanced Research Projects Agency′ (DARPA) - 10 years earlier than the work on the Open Systems Interconnection (OSI) Reference Model started.

TCP/IP is a peer-to-peer or host-to-host architecture. All systems, regardless of size, appear the same to all other systems on the network. Systems using TCP/IP functions are logically paired in a client-server relationship. In general, TCP/IP requests are issued from the client, or local host, and forwarded through an internet to the server, or foreign host. The server performs the function and returns the result to the client. TCP/IP is considered today as the industry-wide standard for connecting heterogeneous systems.

The TCP/IP protocol suite consists of four functional layers, plus the physical link between the hosts:

1. ***The Application Layer.***
   Provides for application-to-application cooperation on the same host, network or across a set of interconnected networks. It is built on the services of the transport layer. Examples are Telnet (protocol for remote terminal connection), FTP (File Transfer Protocol), RSH (Remote Shell) and SMTP (Simple Mail Transfer Protocol) for electronic mail, NFS (Network File System), as well as for business specific client/server processing.

2. ***The Transport Layer.***
   Provides for reliable end-to-end data transfer. It is built on the services of the internetwork layer. Protocol examples are Transmission Control Protocol (connection-oriented) (TCP) and User Datagram Protocol (connectionless) (UDP). Both segment the data to be sent into packets, attach a header containing control information, pass the packets to the internetwork layer, and reassemble the data at the receiving end. In addition, (only) TCP ensures data integrity and ensures that each packet is received and that all packets are reassembled in proper order. If a packet is not received, TCP retransmits the packet. UDP offers no guarantee of data delivery, but has a minimum of protocol overhead.

3. ***The Internet Layer.***
   This layer is built on the services of the network interface layer to communicate with other TCP/IP hosts within the same network, and with hosts in other networks in an internet. The internet layer selects the next node to forward data to the final destination. It does not include provisions for reliable data transmission, flow control, or error recovery, since these services are provided by the transport layer. The most important protocol in this layer is IP.

4. ***The Network Interface Layer.***
Interfaces to the actual network hardware. TCP/IP supports many network interfaces, for example Token-Ring, ETHERNET, X.25, and Serial Line. The TCP/IP software will vary according to the type of media.

The typical OSI model consists of seven layers. A comparison of the layers in OSI and in TCP/IP is shown in Figure 1.

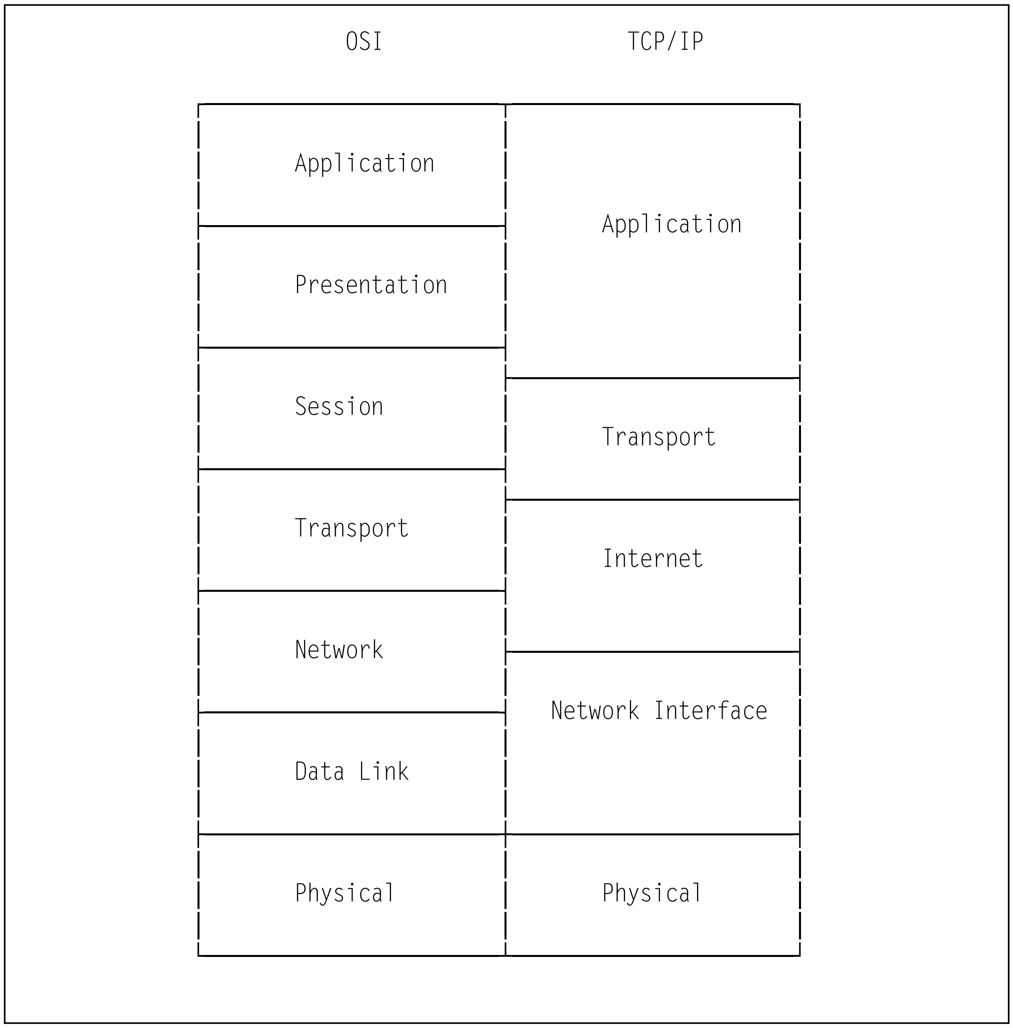


Figure 1. Comparison of the OSI and the TCP/IP Model

There are several books that describe TCP/IP in detail, for example:

- *TCP/IP Tutorial and Technical Overview, GG24-3376*
- *Inside TCP/IP, SR28-5701*

Chapter 2. VSE/ESA and TCP/IP

The options for VSE/ESA to participate in TCP/IP networks are:

- ***OCS TCP/IP for VSE***
  Available from OCS (OpenConnect Systems), Inc. as a part of the IBM Cooperative Software Program. This is the gateway solution.

- ***TCP/IP for VSE***
  Available from Connectivity Systems Inc., providing a native solution. For a description of this implementation please refer to *TCP/IP Native Solution for VSE/ESA with Connectivity Systems Software - Implementation Guide* (available 1Q 1997).

This document describes and explains the OCS TCP/IP for VSE gateway solution.

2.1 OCS TCP/IP for VSE

OCS TCP/IP for VSE lets VSE/ESA participate in an internet network by providing client/server support for Telnet, FTP and RSH. TCP/IP Telnet and FTP clients as well as servers have now the ability to communicate with VSE/ESA. This is achieved via OCS software running on a RISC/6000, called the ′OCS II Gateway′, which provides basic connectivity to VSE/ESA, that is, Telnet 3270 terminal access. In addition, there are six optional features available which execute on the VSE/ESA host:

1. the **OpenConnect/TELNET Client Full Screen** (OC/TELNET FS), a VTAM application which provides ASCII terminal emulation for 3270 terminals.
2. the **OpenConnect/File Transfer Program Client** (OC/FTP Client), a VTAM application which allows VSE users to initiate file transfer operations to and from remote TCP/IP nodes (in both batch or online mode).
3. the **OpenConnect/File Transfer Program Server** (OC/FTP Server), running in a VSE dynamic batch partition and allowing remote TCP/IP nodes to move files to or get them from the VSE host.
4. the **OpenConnect/Remote Shell Client** (OC/RSH Client), a VTAM application which allows VSE users to execute commands on a remote TCP/IP node (in both batch or online mode).
5. the **OpenConnect/Socket Access Method** (OC/SAM), a development toolkit that provides the socket application interface on the VSE/ESA system to enable development of client/server applications distributed between VSE and TCP/IP systems.
6. the **OC/Line Printer Daemon** (OC/LPD), a line printer server that allows clients on a TCP/IP network to use the VSE/ESA line printer as a remote printer.

An optional feature executing on the OCS II Gateway is:

- The **OCS Print Server for AIX** (OCS Print Server), installed in the RISC/6000 emulates multiple 3287 CICS Report Controller terminal printers.

A conceptual view of the OCS TCP/IP for VSE solution is shown in Figure 2 on page 6.

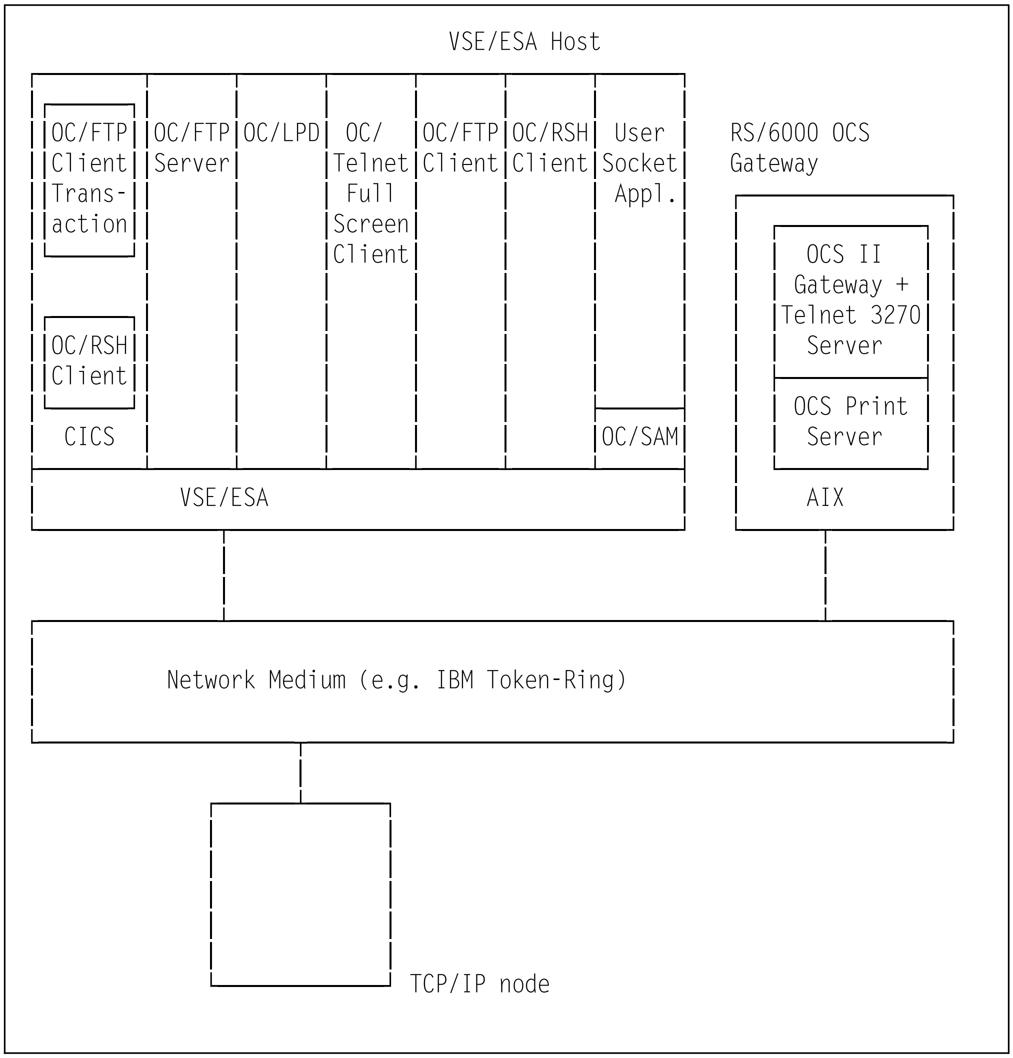


Figure 2. OCS TCP/IP for VSE Overview

2.1.1 OC/Telnet

OC/Telnet Server Manager is a gateway server that includes TN3270 and TN5250 datastream servers, and 3278 and 5250 emulation servers. Host sessions with SNA hosts are monitored and controlled through the server manager. The datastream servers allow you to establish sessions with SNA host applications (through the gateway) on your TCP/IP terminal, PC, or UNIX host. The emulation servers allow you to use a TELNET client on your TCP/IP host, freeing up processing on the TCP/IP host, since the emulation processing occurs in the gateway.

2.1.2 OC/FTP Client

With OC/FTP Client installed on the VSE system, you can log on to an FTP server on a TCP/IP host that is connected with an OpenConnect Server, and transfer files between the two hosts (VSE and TCP/IP).

OC/FTP Client provides access to the following VSE file types:

- VSAM ESDS
- VSAM KSDS
- VSAM SAM
- VSE Library Members (ESA mode only)
- VSE non-VSAM sequential data sets (SD files)

OC/FTP Client may be used in on-line mode via the standard CICS command interface, or in off-line mode using JCL execution statements.

2.1.3 OC/FTP Server

With OC/FTP Server installed on the VSE system, you can log on to the VSE system from an TCP/IP FTP client on a TCP/IP network that is connected with an OpenConnect Server, and transfer files between the two hosts (VSE and TCP/IP).

OC/FTP Server provides access to the following VSE file types:

- VSAM ESDS
- VSAM KSDS
- VSAM SAM
- VSE Library Members (ESA mode only)
- VSE POWER queues (ESA mode only)
- VSE non-VSAM sequential data sets (SD files)

In the VSE/ESA environment, through the OC/FTP Server, FTP clients can submit jobs to VSE/POWER for execution on the VSE/ESA system and send files to the VSE/POWER queues for printing or punching. The FTP client can also retrieve data from the VSE/POWER queues. In addition, the FTP clients can send commands to VSE/POWER to display the contents of a queue, and release jobs for execution.

2.1.4 OC/RSH Client

OC/RSH Client is a software program that allows System/370 users to access hosts on a TCP/IP network for execution of commands.

The OC/RSH Client program delivers commands to a remote TCP/IP host for execution by the remote command server program running on the TCP/IP host. The remote command server program validates the user′s authorization for remote execution, executes the command, and returns the results of the command to the RSH Client for presentation to the user. The command cannot perform interactively, the user has no ability to deliver any data to the command other than specified on the command line itself. The command must read its input from the standard UNIX input **(stdin)**, and write any messages for the user to the standard UNIX output **(stdout)**

2.1.5 OC/SAM

The OC/SAM (Socket Access Method) software product allows an IBM mainframe computer to communicate with a local area network using an OpenConnect Server. The OC/SAM program on the IBM mainframe communicates with a partner program on the LAN, using either the User Datagram Protocol (UDP) or Transmission Control Protocol (TCP).

The SAM offers functions that take the complexity out of programming in the VTAM, CICS, and LAN environments. To start the process, the mainframe application obtains the internet address and port number of the partner program on the LAN. The data is then sent and received. The transmission control headers, VTAM protocol, and CICS requests are handled by the OC/SAM software product. The only programming required on the mainframe is related to the protocol established by the partner program.

2.1.6 OC/Line Printer Daemon (LPD)

The OC/LPD allows terminal users on a TCP/IP network to use the VSE/ESA line printer as a remote printer for their system. The OC/LPD product provides TCP/IP network users a method of printing ASCII text files on mainframe attached printers. Print files submitted to a standard LPR/LPD defined remote print queue in the TCP/IP network are passed through the gateway to the VSE/ESA host, where the resultant text is printed using host-based printer resources.

2.1.7 OCS II Gateway

The OCS II Gateway is a software-only version of OpenConnect Systems′ wide range family of connectivity products that provide gateway, router, and protocol conversion functions.

OCS II Gateway software provides System Network Architecture (SNA) connectivity for UNIX workstations or allows a UNIX workstation to function as a gateway between TCP/IP networks and SNA networks. OCS II Gateway provides the capability for data transfer between the SNA host system and intelligent devices within the TCP/IP network.

2.1.8 OCS Print Server for AIX

The OCS Print Server is an AIX process to emulate multiple 3287 CICS terminal printers simultaneously. The print output is automatically converted from EBCDIC to ASCII and routed to any local or remote printer defined on the AIX system.

2.2 OpenConnection for Channel (OCC)

OpenConnect System′s OpenConnection for Channel (OCC) provides a mainframe channel attachment for the OpenConnect Systems software product OCS II Gateway.

OCC is a combination of software and hardware in its own enclosure, providing connectivity between a Small Computer System Interface (SCSI) on a UNIX host system and an IBM mainframe computer via a System/370 I/O Interface Channel.

The OCC is a channel attached hardware device that resides in the data path between an IBM channel and a SCSI interface on a UNIX workstation. OCC provides the physical and electrical connection between the UNIX workstation and the IBM bus and tag channel cables.

2.3 Hardware and Software Requirements for OCS TCP/IP

The following will briefly specify the required hardware and software in order to install and run OCS TCP/IP.

2.3.1 Hardware Requirements

The following hardware is required for OCS TCP/IP for VSE:

1. ***OCS Gateway (RISC/6000)***
   - OCS TCP/IP for VSE supports the RISC/6000 Models 320 or higher
   - Depending on the type of connection to TCP/IP and the VSE host, the appropriate hardware is required as shown in Table 1

Connection	Hardware Requirement	Required For
TCP/IP	IBM Ethernet High-Performance LAN Adapter (entN) IBM Token-Ring High-Performance Adapter (tokN)	Downstream TCP/IP workstation connectivity
SNA SDLC	IBM 4-Port Multiprotocol Communications Controller (mpqaN) IBM 4-Port Multiprotocol Interface Cable	SNA host communications using SDLC
Token-Ring	IBM Token-Ring High-Performance Adapter (tokN)	SNA host communications over Token-Ring

Table 1. OCS TCP/IP for VSE Network Connection Options

2. ***VSE Host***
   - The optional features running on the host can be installed on any processor supporting VSE/ESA
   - A tape drive (for example, 3480) is required to load the appropriate distribution tape
   - One of the following connections is required to attach the host to the OCS gateway:
     - IBM Token-Ring
     - SNA SDLC

2.3.2 Software Requirements

The following software is required for OCS TCP/IP for VSE:

1. ***OCS Gateway (RISC/6000)***
   - AIX Version 3.1 or higher
   - depending on the type of connection to TCP/IP and the VSE host, the appropriate software is required as shown in Table 2

Product Name	Description/BOS Section	Required For
bosnet.tcpip.obj	TCP/IP Applications BOS Network Support Facilities	OCS Gateway and TCP/IP LAN communications
bosext2.dlctoken.obj	Token-Ring Data Link Control BOS Extensions 2 Interface Cable	SNA host communications over Token-Ring
bosext2.dlcsdlc.obj	SDLC Data Link Control BOS Extensions 2	SNA host communications using SDLC

Table 2. OCS TCP/IP for VSE RISC/6000 Software Requirements

2. ***VSE Host***
   - VSE/ESA Version 1.3.2, or later
   - ACF/VTAM Version 3.4.0, or later
   - CICS/VSE Version 2.2, or later

A typical OCC environment with OCS II serving as a gateway between a TCP/IP network and a channel attached IBM mainframe is shown in Figure 3 on page 11.

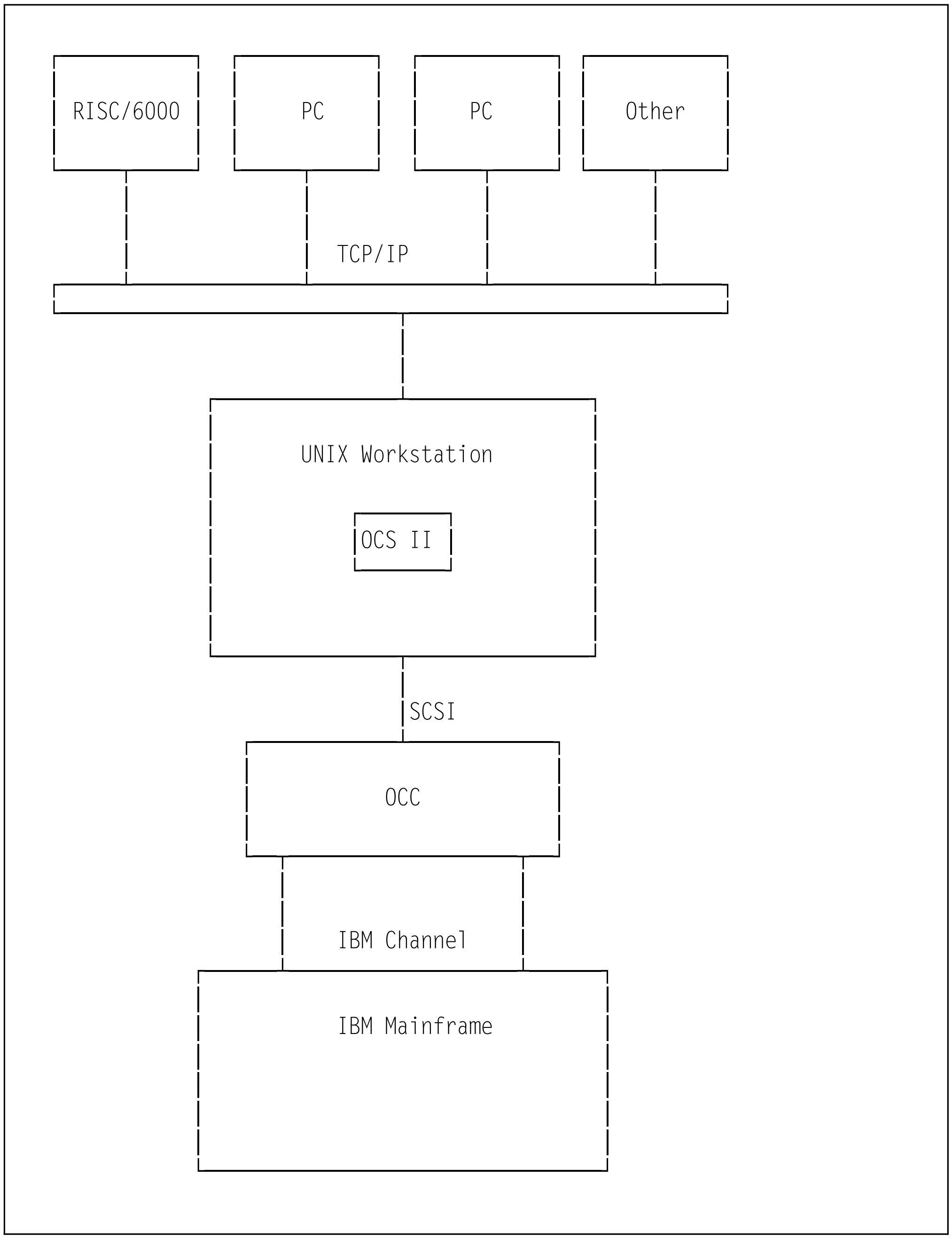


Figure 3. OCC Environment

2.3.3 OCC Hardware

The OpenConnection for Channel (OCC) is a microcomputer system with electronics and logic components for directly connecting SCSI and System/370 I/O channel. The OCC product is shipped with external power supply, channel cabling, and SCSI cabling.

The OCC enclosure consists of the following components:

- Main Circuit Board
- SCSI Bus/Tag Transition Board

- Front Panel Assembly
- Enclosure Base with Rear Panel
- Enclosure Cover

2.3.3.1 Main Circuit Board

All logic components of the OCC reside on the main circuit board which is mounted horizontally in the enclosure. The major components are:

- CPU/Microcontroller
- Memory
- SCSI Interface
- Channel Interface
- Rear Panel Components

2.3.3.2 SCSI Bus/Tag Transition Board

The SCSI bus/tag transition board mounts vertically on the main circuit board at the rear of the OCC enclosure, and contains the following connectors:

- SCSI Interface
- Channel Interface
- Main Board Connection

The SCSI-2 connectors on the OCC enclosure are bi-directional. Either of these connectors can be used for connection of the SCSI In cable, the SCSI Out cable, or for the SCSI terminator as required.

Note: If the OCC is the only SCSI device connected to the SCSI host, or the last SCSI device in a chain, the SCSI terminator plug must be installed in the unused SCSI connector on the OCC rear panel.

Part 2. OCS TCP/IP for VSE Installation and Implementation

This part covers the implementation of a TCP/IP network for VSE/ESA using the software offered by OpenConnect Systems (OCS). It provides:

- A functional overview of the OCS products described in this book
- A description of the hardware, software and network used in our implementation
- The definition of the OCS II Gateway to VM/VSE and native VSE host environments
- The definitions of the SCSI to Channel environment (OCC)
- The installation and customization of OpenConnect/File Transfer Program Server (OC/FTP Server)
- The installation and customization of OpenConnect/Line Printer Daemon (OC/LPD)
- The installation and customization of OpenConnect/File Transfer Program Client (OC/FTP Client)
- The installation and customization of OpenConnect/Remote Shell Client (OC/RSH Client)
- The installation and customization of OpenConnect/TELNET Client Full Screen (OC/TELNET FS)
- The installation and customization of OpenConnect/Socket Access Method (OC/SAM)
- The installation and customization of OpenConnect Server II Gateway on the RISC/6000 system (OCS II Gateway)
- The installation and customization of OCS Print Server for AIX on the RISC/6000 system (OCS Print Server)
- An implementation summary
- The operation procedures and test samples of the OCS software
- A list of issues and considerations reflecting our experiences

Chapter 3. OCS Software Functional Overview

OCS TCP/IP for VSE is marketed by IBM through the Cooperative Software Program (CSP). Under this program, IBM is not responsible for, nor does IBM represent or warrant, the performance of the products. Technical support and assistance are provided by OCS.

OCS TCP/IP for VSE consists of the following optional products:

- OpenConnect Server II Gateway (OCS II) software installed in the RISC/6000. Different program numbers are used, depending on the number of Logical Units (LUs) supported:
  - 5758-PC3 for 16 LUs
  - 5758-PC4 for 32 LUs
  - 5758-PC5 for 64 LUs
  - 5758-PC6 for 128 LUs
  - 5758-PC7 for 256 LUs

  The TELNET Server 3270 feature of the OCS II allows TCP/IP workstations to connect to VSE/ESA via the OCS II Gateway, and emulates as an IBM 3278 terminal. This is an optional feature and must be ordered if the terminal emulation function for ASCII terminals is required.
- OCS Print Server installed on the RISC/6000
- OC/FTP Server (5758-PC0) installed in VSE/ESA
- OC/FTP Client (5758-PC1) installed in VSE/ESA
- OC/LPD (5758-PC4) installed in VSE/ESA
- OC/RSH Client (5758-PC8) installed in VSE/ESA
- OC/TELNET FS (5758-PC2) installed in VSE/ESA
- OC/SAM (5758-PC9) installed in VSE/ESA

Note

The program numbers listed above are for the U.S. and Canada only. The program numbers for OCS products may be different in other countries. Please consult the program announcement letter in your country for country specific information.

These software products allow VSE/ESA to communicate with hosts and workstations attached to the TCP/IP network, performing file transfer, terminal emulation and remote command execution functions.

The OC/SAM allows VSE/ESA users to write network applications communicating with other applications on TCP/IP hosts.

3.1 OCS II Gateway Functional Overview

OCS II Gateway software resides in the RISC/6000 system. It provides gateway functions between SNA and TCP/IP. The OCS II Gateway uses SNA protocols when communicating with the VSE/ESA SNA node; and standard TCP/IP protocols when communicating with the TCP/IP nodes. With the protocol translation and gateway function provided by OCS II Gateway, communication between VSE/ESA and TCP/IP nodes is made possible. TN3270 Server is supplied which allows interactive terminal sessions using TN3270 and TN3174G clients.

For users having a TELNET client or ASCII terminals, the optional feature TELNET Server 3270 of OCS II Gateway allows connection to VSE/ESA, and emulates the TCP/IP terminal as an IBM 3278 terminal.

Figure 4 illustrates a simplified view of the OCS II Gateway functions:

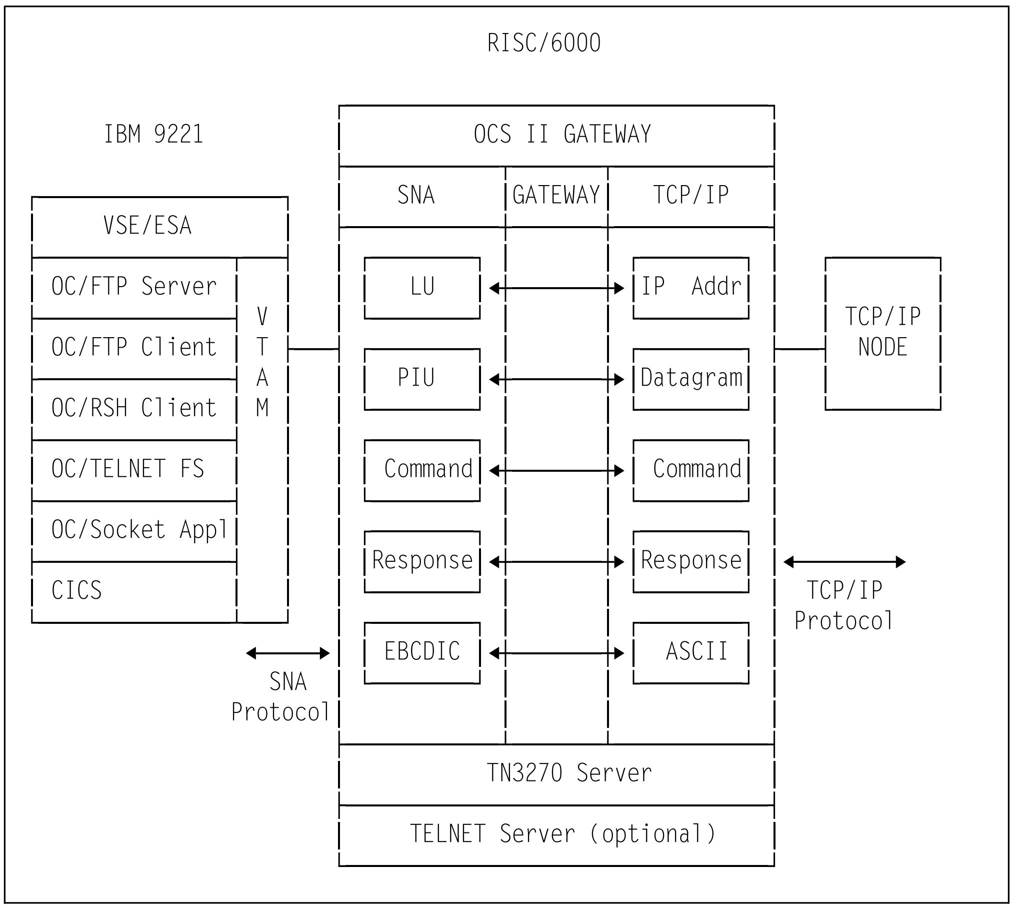


Figure 4. OCS II Gateway Functions

From the SNA side of the gateway, OCS II Gateway:

- Provides LU to LU sessions with VTAM applications and terminals
- Uses SNA Path Information Unit (PIU) as the transmission unit between OCS II Gateway and VTAM

- Uses standard SNA commands and responses between OCS II Gateway and VTAM
- Uses EBCDIC data

From the TCP/IP side of the gateway, OCS II Gateway:

- Communicates with TCP/IP nodes using IP addresses and host names
- Provides IP routing to the TCP/IP nodes
- Uses standard TCP/IP commands and responses between OCS II Gateway and the TCP/IP nodes
- Uses ASCII data

The gateway function of OCS II Gateway provides session management and protocol translation between these two different communication protocols. It provides:

- Managing and mapping of LUs to/from IP addresses
- Conversion and translation of PIUs to/from IP datagrams
- Interpretation and translation of commands and responses between the SNA and TCP/IP protocols
- EBCDIC to/from ASCII data translation
- TN3270 Server allowing interactive terminal sessions using TN3270 and TN3179G

TELNET Server 3270: This optional feature resides in the OCS II Gateway. It allows the TCP/IP TELNET clients to initiate a session with the VSE/ESA SNA host, and log on to VTAM applications as an IBM 3278 terminal. The terminal keyboard mapping can be customized for terminal definitions that are not provided by the software. Please refer to Chapter 12, “OCS II Gateway Installation and Customization” on page 153 for the customization details.

The OC/TELNET Server is a gateway server that includes TN3270 and TN5250 datastream servers, and 3278 and 5250 emulation servers. Host sessions with SNA hosts are monitored and controlled through the server manager. The datastream servers allow you to establish sessions with SNA host applications on your TCP/IP terminal, PC, or UNIX host. The emulation servers allow you to use a TELNET client on your TCP/IP host.

Figure 5 on page 18 illustrates the connection of a TCP/IP TELNET client to VSE/ESA using TELNET Server.

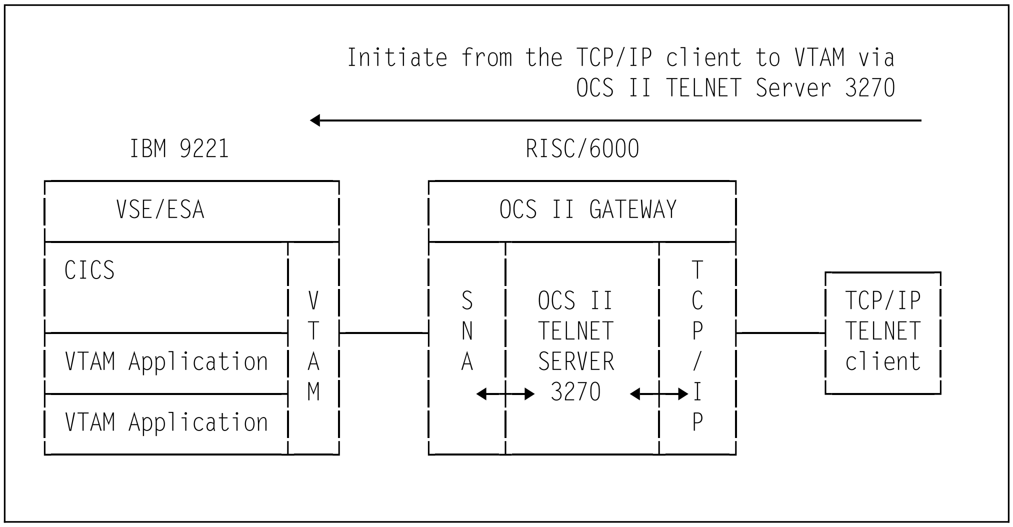


Figure 5. OCS II Gateway TELNET Server Functions

3.2 OC/FTP Server Functional Overview

OC/FTP Server runs in a VSE/ESA dynamic partition as a VTAM application. It allows the FTP clients in the TCP/IP network to initiate an FTP connection with VSE/ESA.

Once connected, the TCP/IP FTP clients may perform file transfer functions between the:

- TCP/IP FTP client file system
- VSE/ESA library
- SAM/ESDS files in VSAM managed space
- VSAM Data Sets (Entry Sequenced Data Sets (ESDS), Key Sequenced Data Sets (KSDS))
- POWER spool files

The TCP/IP FTP client may also issue commands to POWER (for example PALTER, PDISPLAY). User commands for OC/FTP Server are discussed in Chapter 18, “OC/FTP Server Operation and Examples” on page 219.

Some of the potential usages of this product for the TCP/IP FTP clients are:

- Putting data into VSE/ESA VSAM data sets
- Obtaining data files from VSE/ESA VSAM data sets
- Placing procedures and programs into VSE/ESA libraries
- Retrieving procedures and programs from VSE/ESA libraries
- Submitting jobs to execute in VSE/ESA through the POWER RDR queue
- Getting VSE/ESA jobs from the POWER RDR queue
- Retrieving listings and punch output from the POWER LST and PUN queues
- Manipulating the POWER RDR, PUN and LST queues with POWER commands

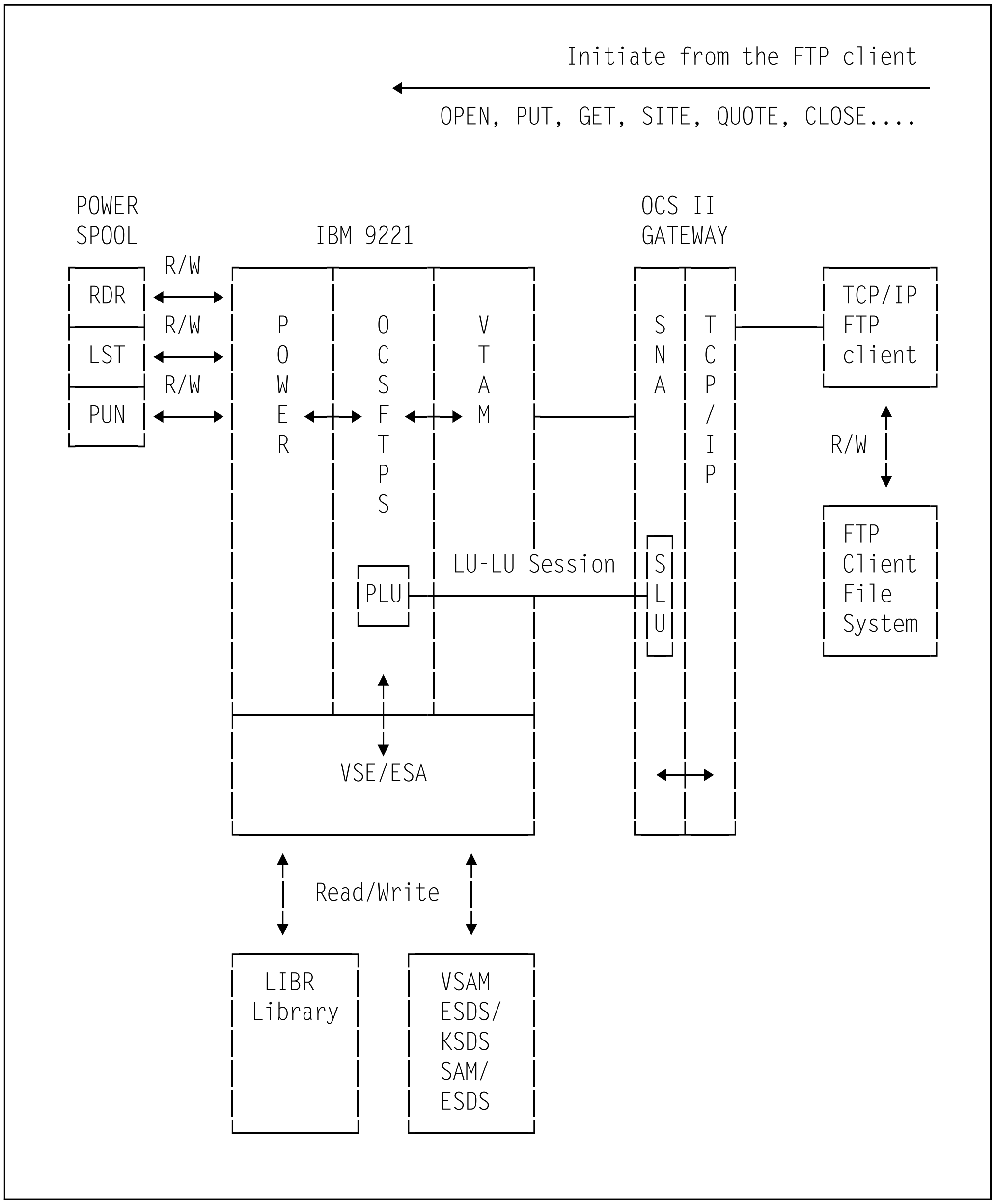


Figure 6. OC/FTP Server Functions

Figure 6 provides a simplified illustration of how OC/FTP Server functions:

- After VTAM, OC/FTP Server, and OCS II Gateway are initialized, the OC/FTP Server program establishes an LU-LU session with an LU in OCS II Gateway. The OC/FTP Server program acts as the Primary LU (PLU) and the OCS II Gateway LU acts as the Secondary LU (SLU). The system is ready to accept requests from the TCP/IP FTP clients
- The TCP/IP FTP client initiates an FTP session request to OC/FTP Server via OCS II Gateway
- OCS II Gateway translates and routes the OPEN command from the TCP/IP FTP client to OC/FTP Server via the LU-LU session
- After the FTP session request is accepted by OC/FTP Server, the TCP/IP FTP client may issue FTP commands such as PUT, GET and SITE. The FTP commands available to the TCP/IP FTP client depend on the software installed in the FTP client′s node

- The request is passed via the LU-LU session from OCS II Gateway to OC/FTP Server
- Depending on the nature of the request, OC/FTP Server may request the services of:
  - VSE Librarian, if the request is to access the VSE/ESA library
  - VSE/VSAM, if the request is to access a VSAM ESDS or KSDS file
  - VSE/POWER, if the request is a POWER command or a request to access the POWER RDR, LST or PUN file
- The requested file (files for MGET and MPUT commands) is routed to the target node via OCS II Gateway
- The FTP client may terminate the FTP session by a CLOSE command

3.3 OC/FTP Client Functional Overview

OC/FTP Client runs in a VSE/ESA dynamic partition as a VTAM application. It enables the VSE/ESA users to initiate an FTP connection with an FTP server node connected to the TCP/IP network. Once connected, the VSE/ESA FTP clients may perform file transfer functions between the:

- TCP/IP FTP server file system,
- VSE/ESA library and
- VSAM ESDS or KSDS
- SAM/ESDS files in VSAM managed space

User commands for OC/FTP Client are listed in Chapter 19, “OC/FTP Client Operation and Examples” on page 239.

Some of the potential usages of this product for the VSE/ESA FTP clients are:

- Putting data into VSE/ESA VSAM ESDS/KSDS or libraries from the TCP/IP FTP server file system
- Putting data into the TCP/IP FTP server file system from VSE/ESA libraries or VSAM ESDS/KSDS
- Obtaining files from the TCP/IP FTP server file system (that is databases, bulletins, fora)
- Retrieving procedures and programs from the TCP/IP FTP server file system to VSE/ESA libraries
- Placing procedures and programs from VSE/ESA libraries into the TCP/IP FTP server file system

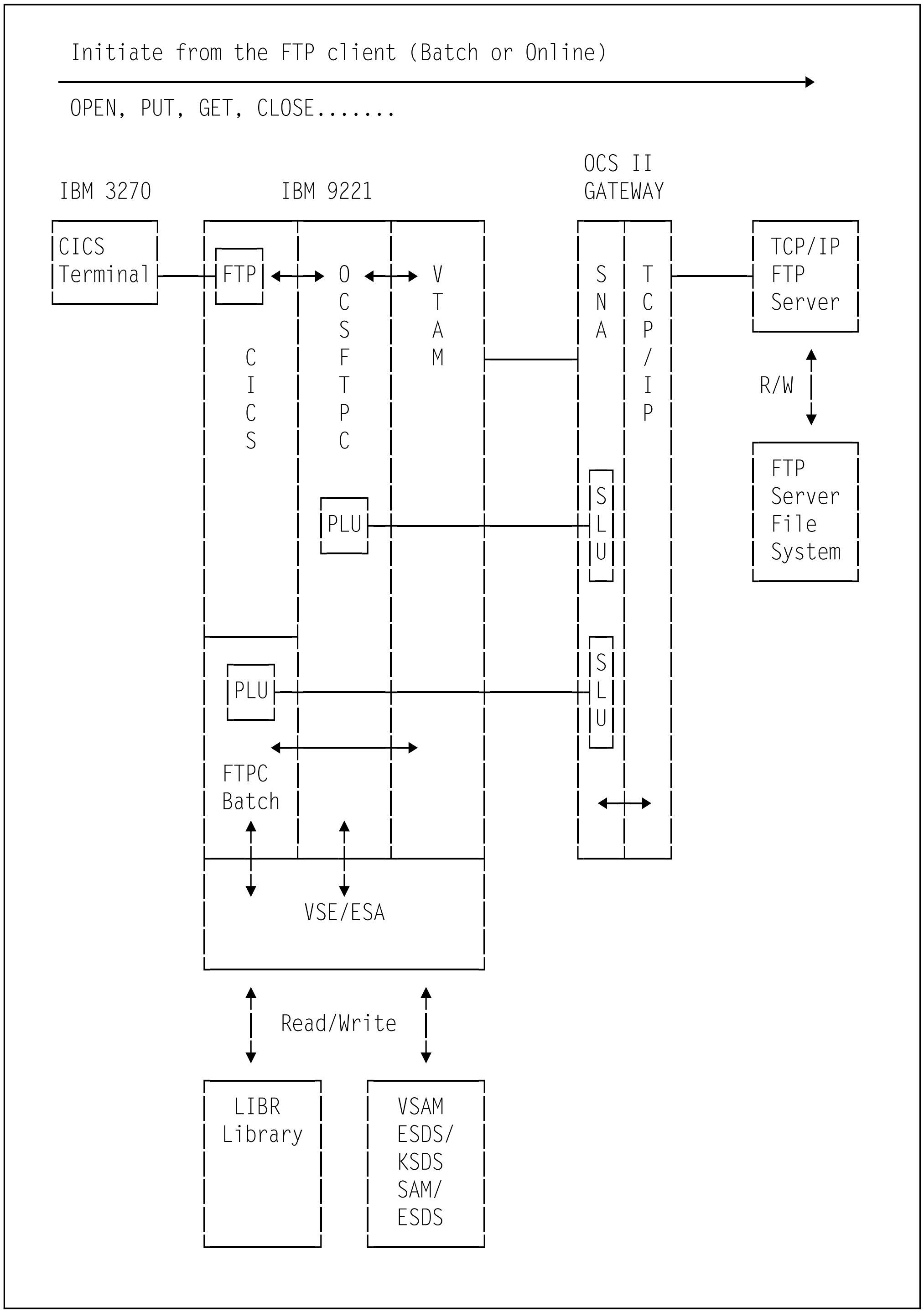


Figure 7. OC/FTP Client Functions

Figure 7 provides a simplified illustration of how OC/FTP Client functions:

- After VTAM, OC/FTP Client and OCS II Gateway are initialized, the system is ready to accept requests from the VSE/ESA users
- The VSE/ESA users may initiate an FTP session from a CICS transaction (FTP) or batch job. OC/FTP Client supports concurrent VSE/ESA online and batch FTP clients
- When the CICS FTP transaction is invoked from the CICS terminal, the online interface of OC/FTP Client is initialized in the OC/FTP Client partition. The

maximum number of the online FTP clients is determined at this stage, based on the specifications of the OC/FTP Client customization. Please refer to 7.2, “OC/FTP Client Customization” on page 82 for details

- When OC/FTP Client is invoked from a batch job, OC/FTP Client and its off-line interface are started in the batch dynamic partition
- When an FTP session request is issued from the VSE/ESA FTP client, an LU-LU session is created between the OC/FTP Client program and the OCS II Gateway LU. The OC/FTP Client program acts as the PLU, while the OCS II Gateway LU acts as the SLU
- OC/FTP Client interprets the session request and passes it to OCS II Gateway via the LU-LU session
- OCS II Gateway translates the request into TCP/IP format and routes it to the target TCP/IP FTP server
- Once the session request is accepted by the TCP/IP FTP server, the VSE/ESA FTP client is requested to log in to the FTP server
- The VSE/ESA FTP client may issue FTP commands such as PUT and GET, to initiate file transfer to/from the TCP/IP FTP server
- The request is translated and routed to the target TCP/IP FTP server via OCS II Gateway
- Depending on the nature of the request, OC/FTP Client may request the services of:
  - VSE Librarian, if the request is to access the VSE/ESA library
  - VSE/VSAM, if the request is to access a VSAM ESDS or KSDS file
  - The TCP/IP FTP server file system, if the request is to access the TCP/IP FTP server files
- The requested file (files for MGET and MPUT commands) is routed to the target system via OCS II Gateway
- The VSE/ESA FTP client may terminate the FTP session by a CLOSE command. This also terminates the LU-LU session between OC/FTP Client and the OCS II Gateway LUs

3.4 OC/Line Printer Daemon Functional Overview

OC/Line Printer Daemon runs in a VSE/ESA dynamic partition as a VTAM application. It allows the LPR clients in the TCP/IP network to initiate and send print data to VSE/ESA.

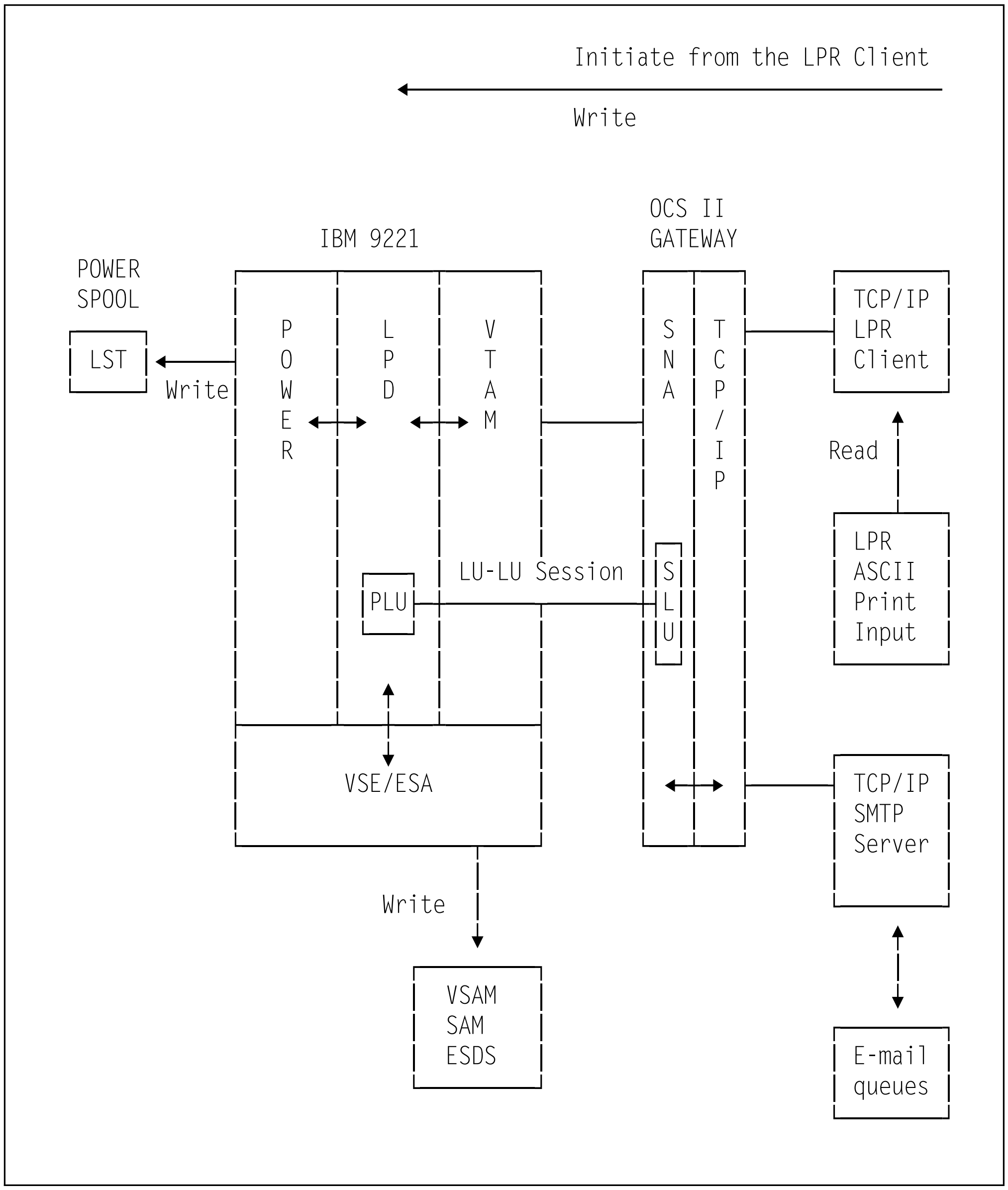


Figure 8. OC/Line Printer Daemon Functions

Figure 8 provides a simplified illustration of how OC/LPD functions:

- After VTAM, OC/LPD, and OCS II Gateway are initialized, the OC/LPD program establishes an LU-LU session with an LU in OCS II Gateway. The OC/LPD program acts as the Primary LU (PLU) and the OCS II Gateway LU acts as the Secondary LU (SLU). The system is ready to accept requests from the TCP/IP LPR clients
- The TCP/IP LPR client initiates an LPR print session request to OC/LPD via OCS II Gateway
- OCS II Gateway translates and routes the OPEN command from the TCP/IP LPR client to OC/Line Printer Daemon via the LU-LU session
- After the LPR session request is accepted by OC/Line Printer Daemon, the TCP/IP LPR client may send the print request
- The print data is spooled to the POWER LST queue or stored in a VSAM SAM ESDS file

- If the E-mail feature is enabled, OC/LPD establishes an LU-LU session with an LU in OCS II Gateway and connects to an SMTP server to queue mail to the E-mail address specified as originator of the print request

3.5 OC/RSH Client Functional Overview

OC/RSH Client runs in a VSE/ESA dynamic partition as a VTAM application. It enables the VSE/ESA users to execute commands on remote TCP/IP hosts. Once connected, the VSE/ESA RSH clients may execute commands on the remote TCP/IP host. The OC/RSH Client program is designed to invoke programs on remote TCP/IP hosts that obtain their input from the command line and write the output to standard output devices.

User commands for OC/RSH Client are listed in Chapter 21, "OC/RSH Client Operation and Examples" on page 281.

Some of the potential usages for VSE/ESA RSH clients are:

- Listing directories from remote TCP/IP systems without going through the FTP login process
- Manipulating files in the TCP/IP server file system

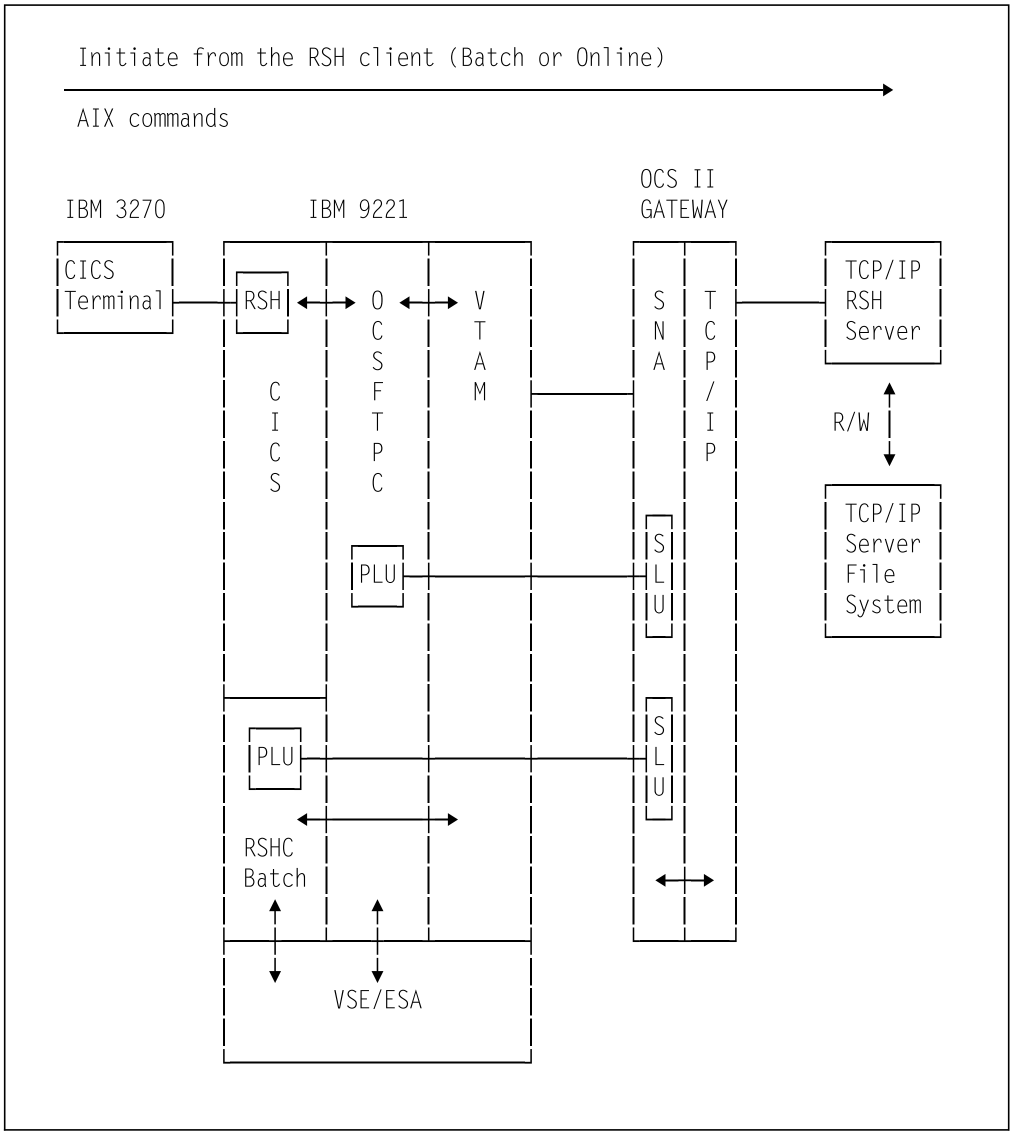


Figure 9. OC/RSH Client Functions

Figure 9 provides a simplified illustration of how the OC/RSH Client works:

- After VTAM, OC/RSH Client and OCS II Gateway are initialized, the system is ready to accept remote shell client requests from VSE/ESA users
- The VSE/ESA users may initiate an RSH command execution from a CICS transaction (RSH) or a batch job. OC/RSH Client supports concurrent VSE/ESA online and batch RSH clients
- When the CICS RSH transaction is invoked from the CICS terminal, the online interface of OC/RSH Client is initialized in the OC/RSH Client partition. The OC/RSH Client is, in fact, an extension of the OC/FTP Client; this is why Figure 9 shows 'OCSFTPC' as the application for RSH transactions and batch requests
- When OC/RSH Client is invoked from a batch job, OC/RSH Client and its off-line interface are started in the batch dynamic partition
- When an RSH request is issued from the VSE/ESA RSH client, an LU-LU session is created between the OC/RSH Client program and the OCS II

Gateway LU. The OC/RSH Client program acts as the PLU, while the OCS II Gateway LU acts as the SLU

- OC/RSH Client passes the command request to OCS II Gateway via the LU-LU session
- OCS II Gateway translates the command into TCP/IP format and routes it to the target TCP/IP RSH server
- The VSE/ESA RSH client holds the RSH session after the command execution for future command executions

3.6 OC/TELNET FS Functional Overview

OC/TELNET FS runs in a VSE/ESA dynamic partition as a VTAM application. It enables the VSE/ESA users from a VTAM terminal, to initiate a TELNET connection to a TELNET server connected to the TCP/IP network. OC/TELNET FS provides terminal functions to emulate an IBM 3278 type terminal as a VT100, VT200, page or line mode terminal. Default keyboard mapping is provided by OC/TELNET FS, and can be customized according to the users′ requirements. Customization of the keyboard mapping is discussed in 10.2, “OC/TELNET FS Customization” on page 134.

Once the TELNET session is established, the VTAM terminal is emulated as an ASCII workstation of the target TCP/IP TELNET server.

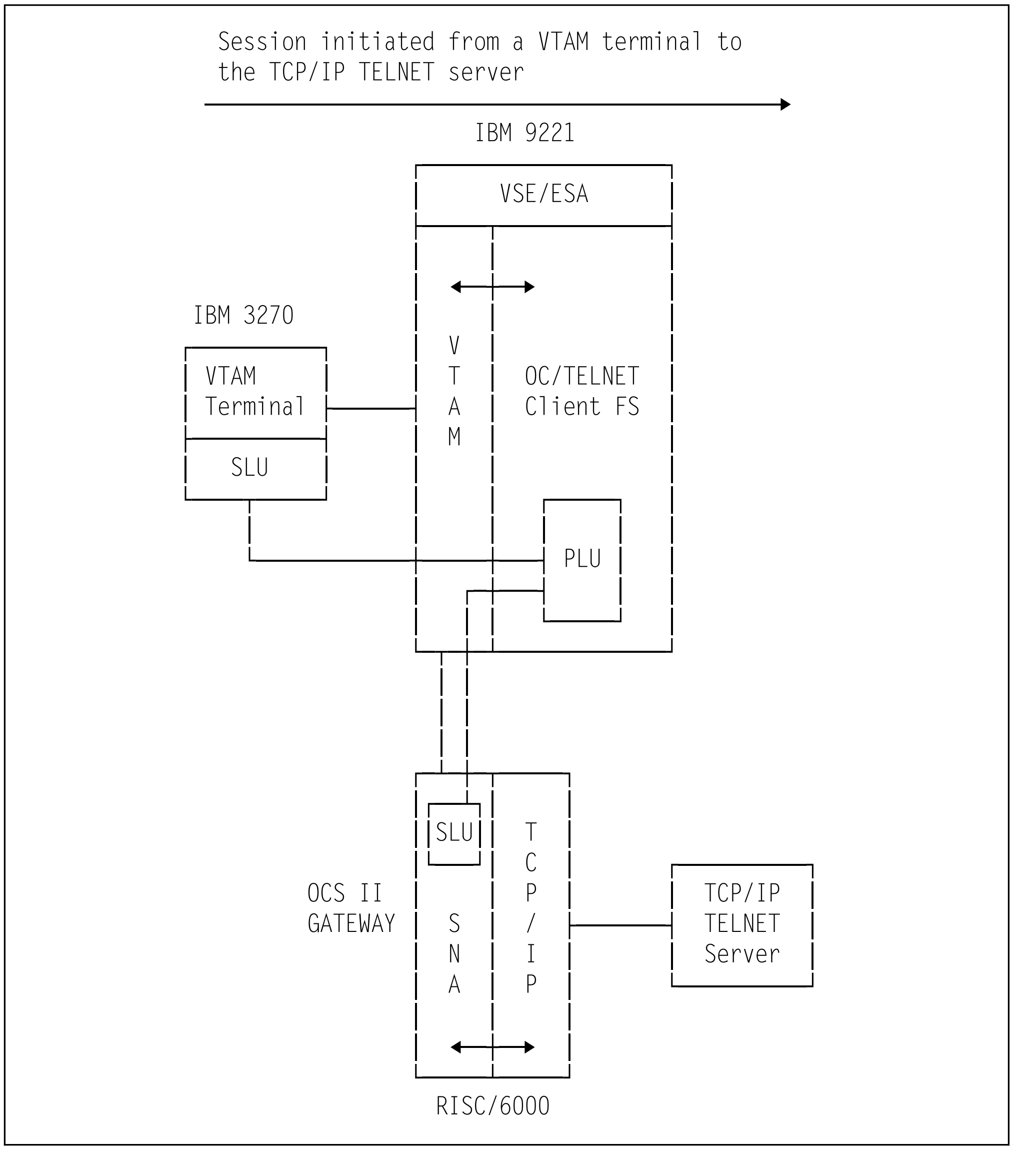


Figure 10. OC/TELNET FS Functions

Figure 10 provides a simplified illustration of how OC/TELNET FS functions:

- After VTAM, OC/TELNET FS and OCS II Gateway are initialized, the system is ready to accept requests from the VSE/ESA TELNET clients
- When a VSE/ESA TELNET client logs on to OC/TELNET FS from a VTAM terminal, an LU-LU session is established between OC/TELNET FS and the VTAM terminal LU. OC/TELNET FS acts as the PLU, and the VTAM terminal LU acts as the SLU
- The VSE/ESA TELNET client may then request a TELNET session to the TCP/IP TELNET server by the OPEN option. OC/TELNET FS creates an LU-LU session with the LU in the OCS II Gateway. OC/TELNET FS acts as the PLU, and the OCS II Gateway LU acts as the SLU. The VTAM terminal LU is then ′connected′ to the OCS II Gateway LU
- The gateway function in OCS II Gateway translates the session request and routes it to the target TCP/IP TELNET server

- After the TCP/IP TELNET server accepts the session request, the VSE/ESA TELNET client may use the VTAM terminal as a local terminal attached to the TCP/IP TELNET server
- The VSE/ESA TELNET client may terminate the TELNET session by the CLOSE option. This also terminates the LU-LU session between OC/TELNET FS and OCS II Gateway
- The VSE/ESA client may log off from OC/TELNET FS by the QUIT option. This also terminates the LU-LU session between OC/TELNET FS and the VTAM terminal LU

3.7 OC/SAM Functional Overview

OC/SAM is a program development toolkit that provides a socket application interface for VSE/ESA to enable the development of client/server applications between VSE/ESA and TCP/IP systems. OCS/SAM allows a VSE application to communicate with an application residing on a TCP/IP system connected through an OpenConnect server. For a VSE/ESA system, OC/SAM implements the functions of the Berkley socket application programming interface available for TCP/IP systems. OC/SAM provides an API for communication programs using either of two transport protocols, TCP (Transmission Control Protocol) or UDP (User Datagram Protocol).

OC/SAM provides two programming environments:

1. The OC/SAM function library
   The base library to be used by programming languages that conform to IBM standard calling conventions.
2. The Berkley Software Distribution (BSD) 4.3 library
   The BSD function library, available to C and ASM/370 programming environments. These functions provide the basis for the socket implementations used by UNIX and IBM/PC platforms.

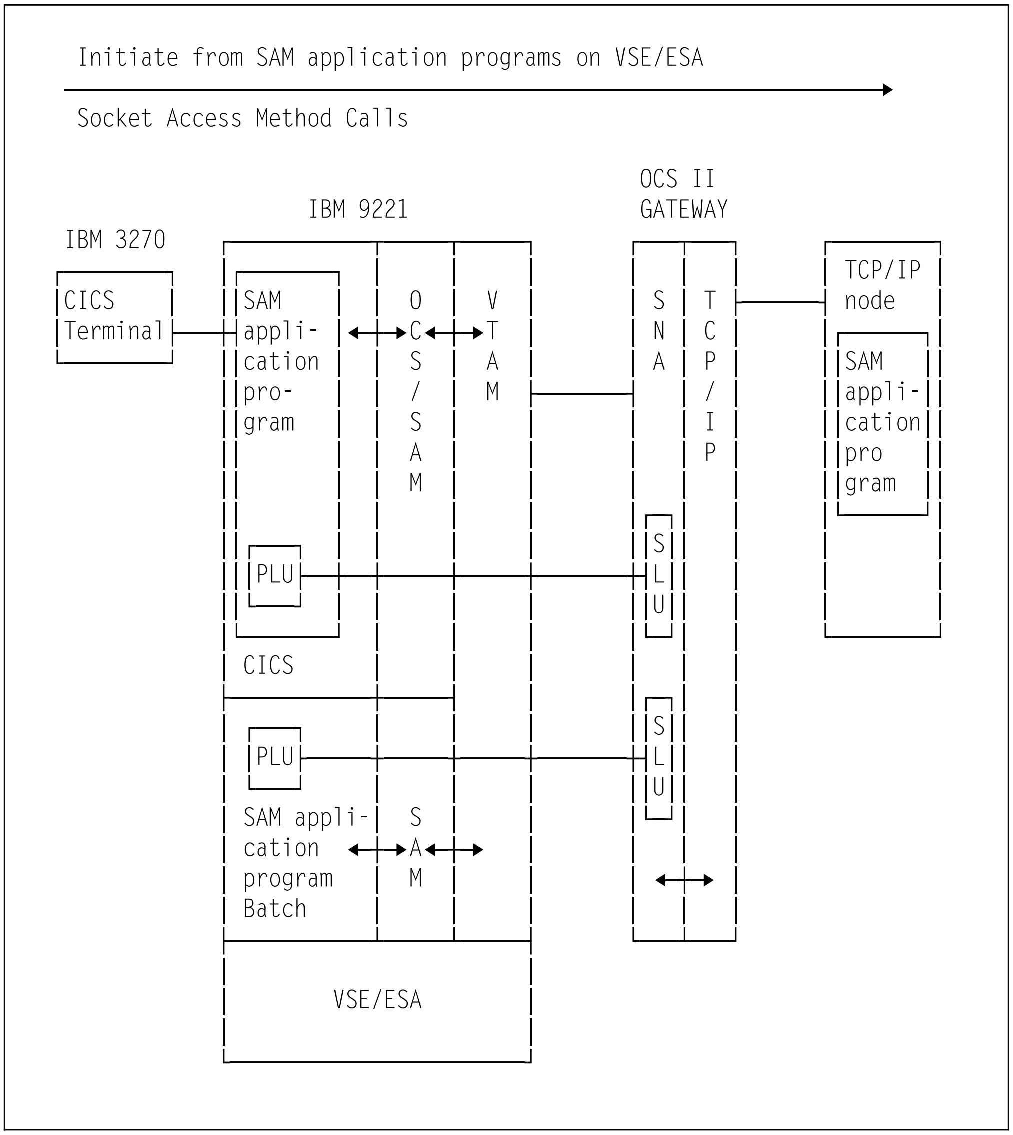


Figure 11. OC/SAM Functions

The figure above provides a simplified view of how OC/SAM functions:

- After VTAM and OCS II Gateway are initialized, the system is ready to accept requests
- A VSE SAM application program can be either the server for (one or more) client programs on TCP/IP nodes, or a client program asking for services from a TCP/IP server program
- The VSE SAM application program first initializes the interface to the OpenConnect Server
- Then it sends messages to the application program on the TCP/IP node and receives responses from it

The VSE/ESA OC/SAM application programs communicate with remote TCP/IP programs using either the TCP or UDP API. The interface between the mainframe application and the OpenConnect server is determined by the ACF/VTAM APPL and LU definitions. OC/SAM can be used from any available programming language that provides access to S/370 assembler calls.

3.8 OCS Software Functional Overview

The figure below shows a simplified overview of the different TCP/IP programs and how they interact with the OCS II Gateway in the RISC/6000 and the different VSE/ESA programs in the IBM 9221. The figure does **NOT** show all details, it is just included to give an overall feeling of how the different parts interact with each other.

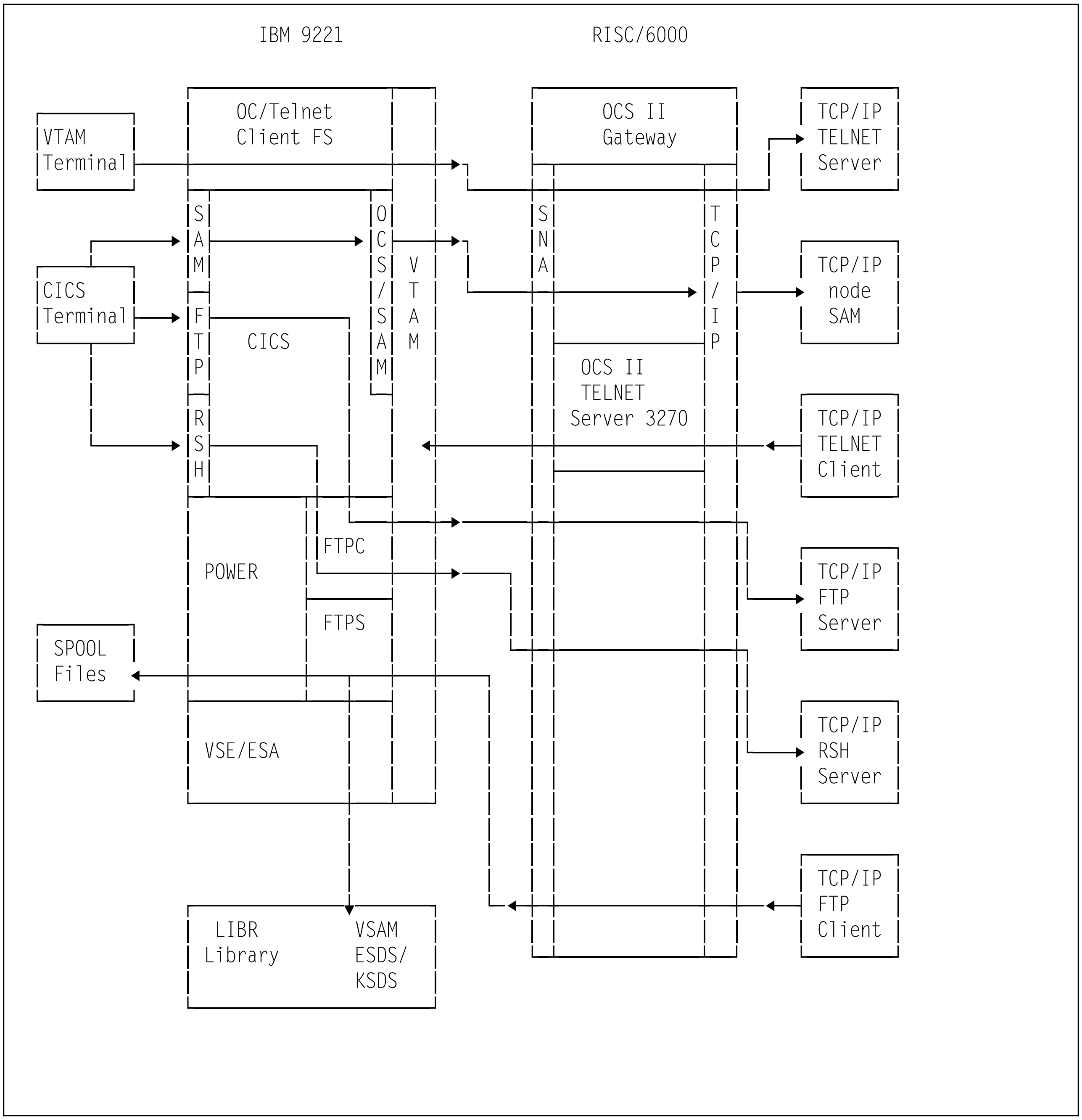


Figure 12. OCS Software Functional Overview

Chapter 4. Sample Environment

Our test environment consists of a working TCP/IP network, and the components that we added to allow VSE/ESA to participate in the network. The resulting network is illustrated in Figure 13 on page 34, and it consists of the following hardware and software.

4.1 Hardware

- An IBM 9221 Model 421 as the host with:
  - 128MB main memory
  - Parallel channel connecting to the IBM 3174
  - Token-Ring adapter connecting to the IBM Token-Ring LAN
- An IBM 3174 Model 11L with:
  - 6MB of real memory
  - Configuration support C release 3
  - Token-Ring adapter
- A RISC/6000 Model 320 as the OCS II Gateway with:
  - 32MB main memory
  - Token-Ring adapter connecting to the IBM Token-Ring Backbone LAN
- An IBM 8209 IBM Token-Ring LAN Bridge as a bridge between two IBM Token-Ring LANs
- A RISC/6000 Model 520 as a router between the IBM Token-Ring Backbone LAN and Ethernet LAN
  - 32MB main memory
  - Token-Ring adapter connecting to the IBM Token-Ring Backbone LAN
  - Ethernet LAN adapter connecting to the Ethernet LAN
- An IBM PS/2 Model 70 as a router between two IBM Token-Ring LANs. This PS/2 also acts as a domain name server as well as an OS/2 Client
  - 10MB main memory
  - Token-Ring adapter connecting to the IBM Backbone Token-Ring LAN
  - Token-Ring adapter connecting to another IBM Token-Ring LAN
- An IBM PS/2 Model 70 attached to the IBM Token-Ring Backbone LAN as a PC DOS Client
  - 2MB main memory
  - Token-Ring adapter connecting to the IBM Backbone Token-Ring LAN
- An IBM PC 350-P100 attached to the IBM Token-Ring Backbone LAN as an OS/2 Client (not part of Figure 13 on page 34)
  - 32MB main memory
  - Token-Ring adapter connecting to the IBM Backbone Token-Ring LAN

- A RISC/6000 Model 560 as an AIX UNIX Client from another IBM Token-Ring LAN
  - 32MB main memory
  - Token-Ring adapter connecting to the IBM Token-Ring LAN
- A SUN Sparc workstation attached to the Ethernet LAN as a SunOS UNIX Client

4.2 Software

- IBM 9221 Host
  - VM/ESA R2.1 with GCS and ACF/VTAM V3.4.1 in the IBM 9221
  - VSE/ESA V2.1.2 guest under VM in the IBM 9221 with
    - ACF/VTAM V4.2.0
    - OC/FTP Server V2.2.4.1, OC/FTP Client V2.2.4.1, OC/TELNET FS V4.11, OC/LPD V1.1.0, OC/SAM V2.1, OC/RSH V2.1.5
- RS/6000-320 OCS II Gateway
  - AIX 3.2.5 preloaded with the following also installed:
    - bosext2.dlctoken.obj 3.2.0, bosnet.tcpip.obj 3.2.0
  - OpenConnect Server II 16 LU with Telnet Server V3.7.5
  - OpenConnect/Print Server for AIX V1.2
- RS/6000-520 AIX Router
  - AIX 3.2.5 preloaded with the following also installed:
    - bosext2.dlcether.obj 3.2.0, bosext2.dlctoken.obj 3.2.0
    - bosnet.tcpip.obj 3.2.0
  - Other Software Applications
- PS/2-70 PC Router, Name Server & OS/2 Client
  - OS/2 V2.10 CSD Level XRU2010
  - Lan Adapter and Protocol Support (LAPS) V2.13 CSD Level WR06000
  - Comm Manager/2 V1.0
  - TCP/IP for OS/2 2.0 & 2.1 Base CSD Level UN64092
  - Domain Name Server Kit V2.0 without CSD
- PS/2-70 PC DOS Client
  - PC DOS 6.1
  - Microsoft Windows 3.1
  - TCP/IP for DOS 2.1.1 without CSD, Token-Ring Driver, NDIS Interface
- PC 350-P100 OS/2 Client (not part of Figure 13 on page 34)
  - OS/2 Warp Verion 3 with DOS and Windows Application Support
  - Multi Protocol Transport Services (MPTS)
  - TCP/IP for OS/2 Version 3
- RS/6000-560 AIX UNIX Client

- AIX 3.2.5 preloaded with the following also installed:
  - bosext2.dlctoken.obj 3.2.0, bosnet.tcpip.obj
- Other Software Applications

- SUN Sparc Workstation SunOS UNIX Client
  - SunOS 4.3 with TCP/IP & Ethernet driver installed
  - Other Software Applications

4.3 Network Configuration

Figure 13 on page 34 illustrates the network used in our implementation.

Our TCP/IP Network mainly consists of three networks, two routers and seven hosts:

- A Token-Ring Backbone Network bridging with a small Token-Ring Network through 8209 Token-Ring Bridge device. This TCP/IP Network uses a Class C address 192.61.100 with a Subnet Mask 255.255.255.0 and has the following TCP/IP nodes attached to it:
  1. VM/ESA and VSE/ESA in the IBM 9221 with:
     - VM-owned IBM 3174 connecting to the IBM Token-Ring LAN
     - VSE-owned IBM 9221 Token-Ring adapter connecting to the IBM Token-Ring LAN
     - OCS software installed in VSE/ESA
  2. A RISC/6000-320 running AIX with OCS II Gateway Software installed
  3. An OS/2 Router also acting as an OS/2 Client
  4. A PC/DOS Windows Client
  5. A RISC/6000-520 Router
- A small Token-Ring Network connecting to the Token-Ring Backbone Network through an OS/2 Router. This TCP/IP Network uses a Class A address 9.164.182 with a Subnet Mask 255.255.255.0 and has the following TCP/IP node attached:
  - RISC/6000 AIX UNIX Client
- An Ethernet Network connecting to the Token-Ring Backbone Network through a RISC/6000 Router. This TCP/IP Network uses a Class C address 192.61.200 with a Subnet Mask 255.255.255.0 and has the following TCP/IP node attached:
  - A SUN Sparc SunOS UNIX Client

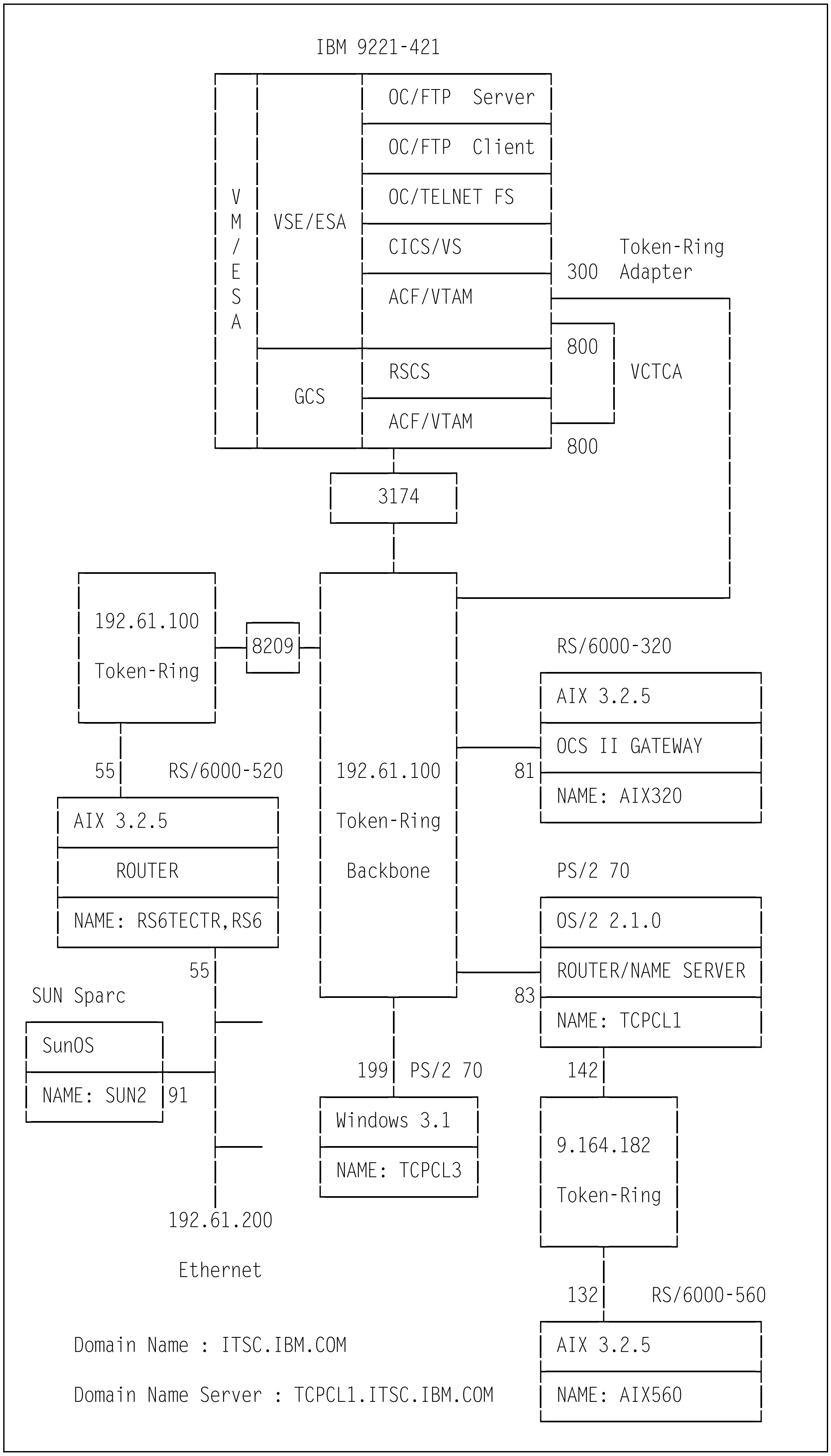


Figure 13. OCS TCP/IP Network Diagram

4.3.1 OCS II Gateway Attachment Configurations

The OCS II Gateway may be connected to the host using an SDLC line, an IBM Token-Ring LAN, or the OpenConnection for Channel box. There are many different options available, such as attachment through the IBM 37x5 communication controller, IBM 3172, IBM 3174, and IBM 9221 integrated communication adapters (Multi-Protocol Communication Subsystem and IBM Token-Ring LAN Subsystem).

The following three attachment configurations are selected, to illustrate the implementations of OCS TCP/IP for VSE in the VM/VSE and native VSE environments.

- Attachment of OCS II Gateway via the IBM 3174

  This attachment represents a VM/VSE environment where OCS II Gateway is owned by and defined in VM/ESA. The OCS II Gateway LUs are defined as cross domain resources in VSE/ESA.

  Virtual Channel to Channel Adapter (VCTCA) is used to connect VSE VTAM and VM VTAM.

  Details of this configuration are covered in 5.1, "VM/VSE with OCS II Gateway Attached via the IBM 3174" on page 41.

- Attachment of OCS II Gateway via the IBM 9221 Token-Ring adapter

  This attachment represents a native VSE/ESA environment where OCS II Gateway is owned by and defined in VSE/ESA.

  Details of this configuration are covered in 5.2, "VSE/ESA with OCS II Gateway Attached via the Token-Ring Adapter" on page 50.

- Attachment of OCS II Gateway via the OpenConnection for Channel

  This attachment represents a native VSE/ESA environment where OCS II Gateway is owned by and defined in VSE/ESA. Details can be found in 5.3, "VSE/ESA with OSCII Gateway Attached via the OCC" on page 56.

4.3.2 IP Address and Name Assignments

As illustrated in Figure 13 on page 34, the configuration consists of three networks connecting by two routers RISC/6000-520 and OS/2 2.1. There also are two types of Client PC (DOS & OS/2) and UNIX (AIX & SunOS). Table 3 on page 36 summarizes the IP address and name assignments for our network:

Network Node	IP Address	IP Node Name
RISC/6000-320 OCS II Gateway[1]	192.61.100.81	AIX320
RISC/6000-520 LAN router	192.61.100.55	RS6TECTR
	192.61.200.55	RS6
OS/2 2.1 LAN router[2]	192.61.100.83	TCPCL1
	9.164.182.142	
PS/2 Model 70 Windows[3]	192.61.100.199	TCPCL3
RISC/6000-560	9.164.182.132	AIX560
SUN Sparc Workstation	192.61.200.91	SUN2

Note:

1. The OCS II Gateway IP address and name apply to the OCS applications in VSE/ESA and OCS II Gateway. VSE/ESA is **not** a TCP/IP node and does not have an IP address. The OCS applications must communicate with other TCP/IP nodes through the OCS II Gateway
2. The OS/2 2.1 also functions as a client for OCS applications in VSE/ESA
3. This PC/DOS Client also has TCP/IP Windows applications installed

Table 3. IP Address and Name Assignment for the OCS Implementation

4.4 Router Functions

In a TCP/IP environment, a host connects to a network, and a network connects to other networks through routers.

In our configuration, we have employed two routers, one on the RS/6000-520 and another one on a PS/2 Model 70 to connect one Ethernet and one Token-Ring network to the Backbone Token-Ring Network, (see Figure 13 on page 34). This has been set up to illustrate a situation which can be found at many customer sites. If your VSE host wants to connect to remote TCP/IP hosts or nodes on another network (of the same or different IP class) via one or more routers, the routing tables in each TCP/IP node need to be set up accordingly.

For more information on the host network member customization, please refer to 7.2.4, "FTP Client Customization" on page 92.

There are three types of route that can be established:

- Implicit route - is defined when the adapter is configured
- Explicit route - also referred to as ′static route′ which has to be coded explicitly
- Dynamic route - employs a daemon process to periodically update the routing table (routed or gated daemon)

The IP routing algorithm will only consider the IP network address part of the destination address. By looking through the routing table, it performs sequentially:

1. Direct routing - direct network attach
2. Indirect routing - specific route to a remote host via a gateway

3. Default routing - default route to a remote host via a gateway; this is used when the destination IP network address is not found in the direct or indirect routing entries

When default routing is applied and the destination network still cannot be reached, the message ″No route to host″ is issued. Figure 14 shows the logical view of our network and in Table 4 the corresponding routing table entries for each of our TCP/IP nodes are listed.

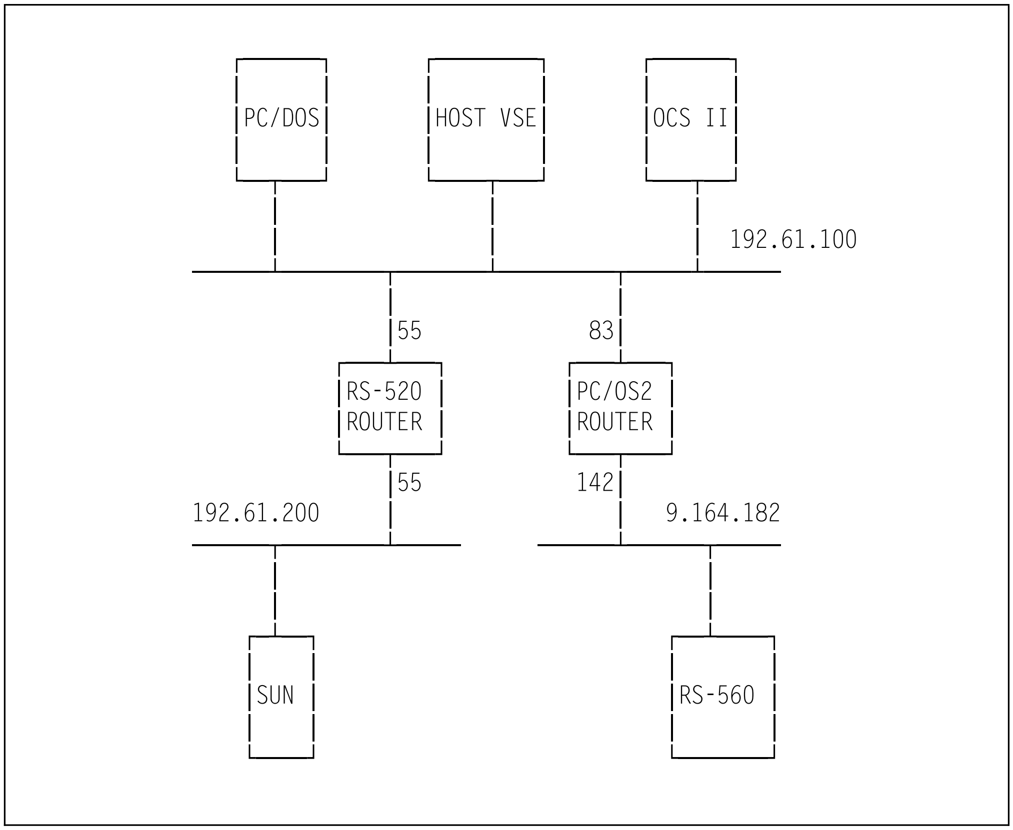


Figure 14. OCS TCP/IP Network Logical View

TCP/IP Node	Route Type	Destination	Router
PC/DOS Windows Client	Net	192.61.200	192.61.100.55
	Default		192.61.100.83
RISC/6000-320 OCS II	Net	192.61.200	192.61.100.55
	Default		192.61.100.83
RISC/6000-520 Router	Net[1]	9.164.182	192.61.100.83
	Default		192.61.100.83
OS/2 2.1 Router	Net[1]	192.61.200	192.61.100.55
	Default		192.61.100.55
SUN UNIX Client	Default		192.61.200.55
RISC/6000-560 UNIX Client	Default		9.164.182.142

Note:

1. If this route type is omitted, Default routing will be used instead of Indirect routing.

Table 4. Routing Table for TCP/IP Nodes

4.5 PC/DOS Windows Client in our Test Environment

Many of the standard TCP/IP applications (Telnet, FTP and so on) are supported on UNIX and PC platforms. In addition to the standard TCP/IP applications FTP, Telnet and RSH the OCS II Gateway supports also the Socket Access Method for VSE/ESA to provide for TCP/IP network applications.

We have also looked into the capabilities of the PC/DOS Windows Client to participate in our test environment. Figure 15 shows the IBM TCP/IP for DOS V.2.11 Windows applications that are contained in the base package.

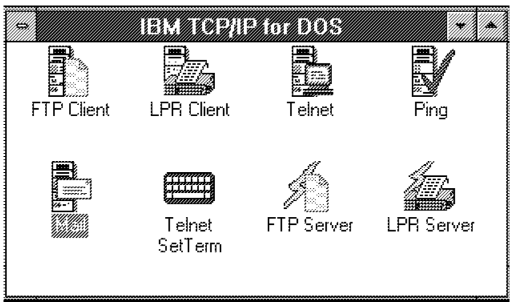


Figure 15. IBM TCP/IP Windows Applications

There are three applications which we used quite often in our environment. They all have a graphical interface and provide very useful functions:

1. TELNET Client - wtelnet.exe

   In this application, we can define TCP/IP hosts to which we want to log on. A click on the host′s icon gets us a telnet session to that host. Figure 16 shows the GUI of this Telnet application.

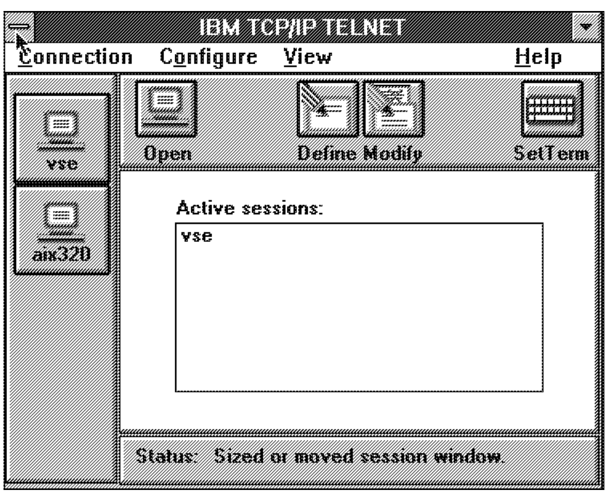


Figure 16. Telnet Windows Application on PC/DOS

2. FTP Client - wftp.exe

   This application reduces the FTP operation to a ′point and click′ interface. Your local and remote file systems will be represented in the upper and lower box. To copy or move files, just select files and click on the up or down arrows.

Problem

We were unable to get this application to work with the OC/FTP Server. The system type this application supports must be OS/2, MVS, VM, AS/400, UNIX, SUN, FTP Inc., NetManage, DOS or Negotiate, that is, VSE/ESA is not supported. The error message says that the remote host doesn't return the system type when trying with 'Negotiate'.

Figure 17 shows the GUI of the FTP application in session with an AIX host.

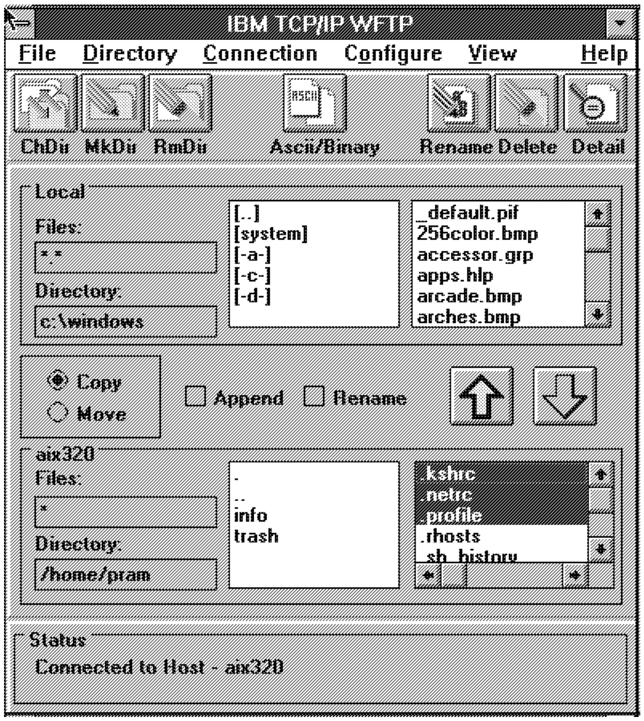


Figure 17. FTP Windows Applications on PC/DOS

3. SMTP mail client and server

   SMTP stands for Simple Mail Transfer Protocol. On the PC/DOS Windows, you can implement your mail as SMTP ONLY or POP. POP stands for Post Office Protocol, in this case the PC has to retrieve the mail from a POP Server in certain intervals of time. We have set up our PC/DOS to use SMTP ONLY for the mail function. Figure 18 on page 40 shows the GUI of the mail application:

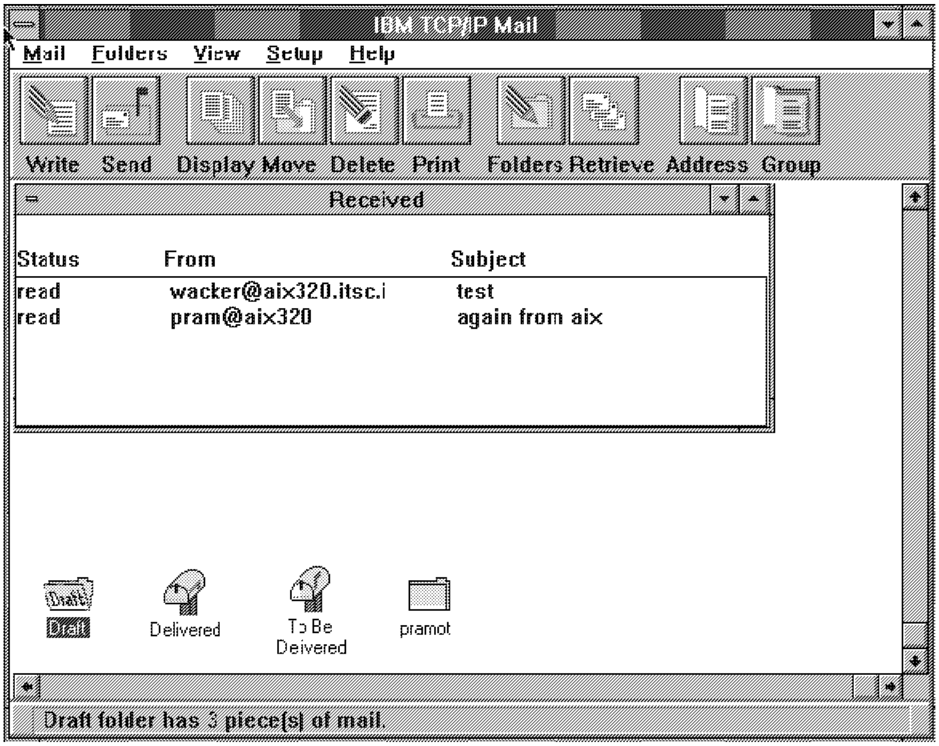


Figure 18. Mail Windows Application on PC/DOS

Chapter 5. Defining the OCS II Gateway to the Host

This chapter covers the definitions in VM, VSE and VTAM to implement the two selected attachments for the OCS II Gateway.

Independent of the OCS II Gateway attachment configuration, OCS II Gateway is defined to the owning VTAM as Physical Unit (PU) type 2. PU name IPFP2209 with LU names ranging from IPFT2S9A to IPFT2S9P are used. The LU names assigned to the OCS applications are:

- IPFT2S9A to OC/FTP Server
- IPFT2S9B - IPFT2S9H to OC/FTP Client
- IPFT2S9I to OC/LPD
- IPFT2S9J - IPFT2S9L to the OC/TELNET FS out-bound terminal emulation sessions
- IPFT2S9M to the terminal printer session
- IPFT2S9N - IPFT2S9O to the OC/TELNET FS in-bound terminal emulation sessions
- IPFT2S9P to OC/SAM sample programs

The LUs are defined in VM VTAM, VSE VTAM, OCS II Gateway and the OCS application customization. Details of the LU parameters are covered in the OCS II Gateway and OCS application customization chapters.

To automatically start up the path table and major nodes during VTAM initialization, their names are included in the VTAM configuration list.

Definition of the PS/2 workstations

No SNA definition is required for TCP/IP nodes that communicate with VSE/ESA via the OCS II Gateway. In our network implementation, the PS/2 workstations participate in the network **only** through TCP/IP. SNA definition is therefore **not** required.

5.1 VM/VSE with OCS II Gateway Attached via the IBM 3174

In this configuration, VM VTAM owns the IBM 3174 and OCS II Gateway. Both of these controllers are defined in the VTAM Local SNA major node. VM VTAM communicates with VSE VTAM through VCTCA, using line address 800 on both sides.

VSE/ESA runs as a guest (user ID V132A80K) under VM/ESA. The LUs of OCS II Gateway are defined in the cross domain resources major node.

To prepare for this connection configuration, the following environments are customized:

- VM/ESA
- VM VTAM

- IBM 9221
- VSE/ESA
- VSE VTAM

The definitions in both VM/ESA and VSE/ESA are closely related, and certain definition parameter values must match. Figure 19 on page 43 provides a summary of the attachment configuration and VTAM definitions. Pointers in the diagram illustrate the relationships of the definitions and where parameter values must match. This diagram should be referred to while studying the sample definitions provided in this document.

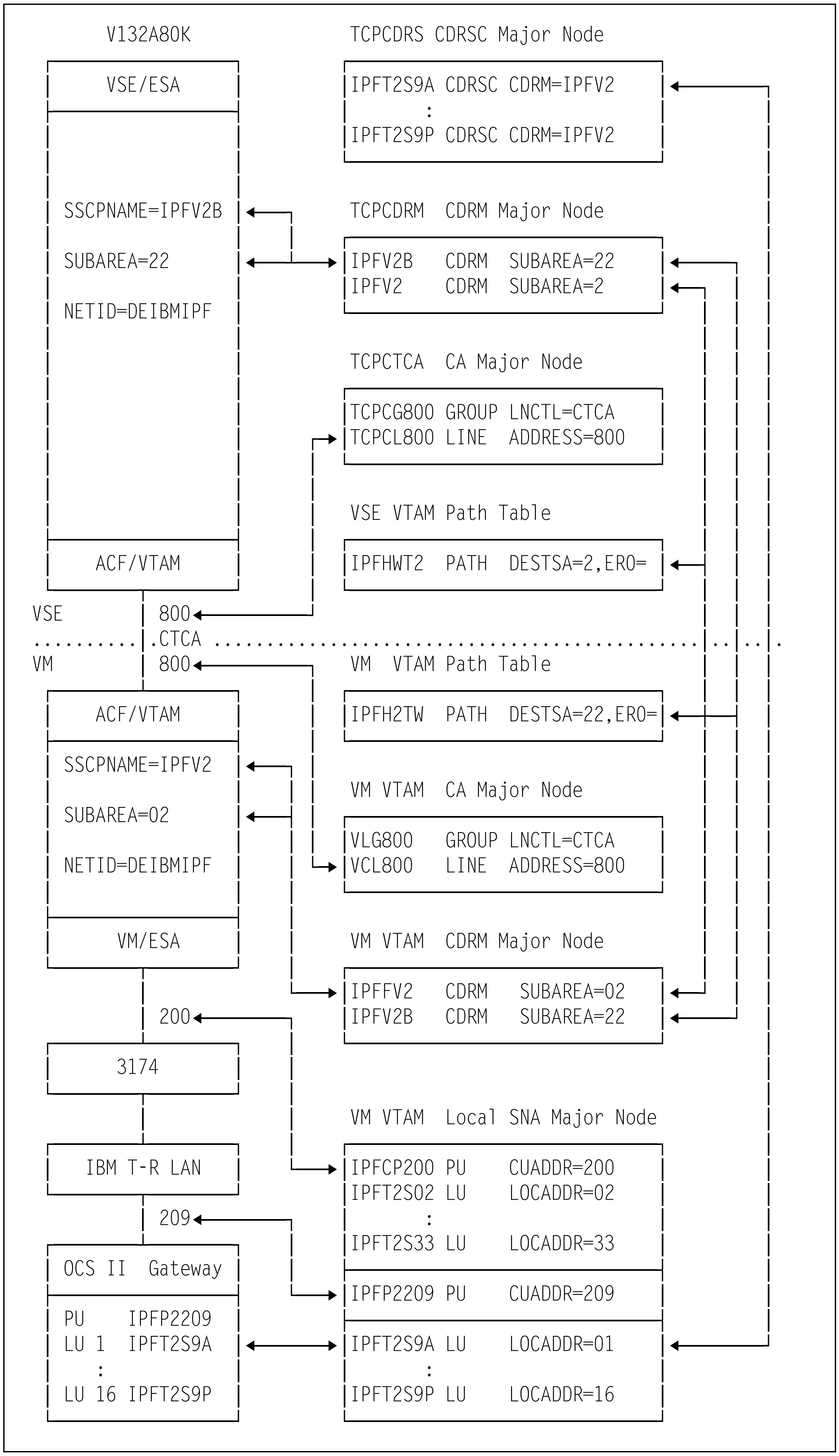


Figure 19. Configuration and Definition Summary for 3174 Connection

5.1.1 Customization in VM/ESA

The following tasks are performed in VM/ESA:

- Define the IBM 3174 and RISC/6000 controllers to VM/ESA
- Tailor the VTAM user directory to support VCTCA, IBM 3174 and RISC/6000
- Tailor the VSE/ESA user directory (V132A80K) to support VCTCA
- Tailor the PROFILE EXEC of VTAM or VSE/ESA guest to couple the VCTCA connections

5.1.1.1 Device Definitions

VM/ESA R2.1 uses dynamic configuration and automatic device sensing. Static device definition is not required. The following statements are included in the VM SYSTEM CONFIG file, to activate and auto-sense device addresses 0000-FFFF during system initialization.

```
Devices ,
  Online_at_IPL   0000-FFFF,
  Sensed          0000-FFFF
```

Figure 20. Device Address Definitions in SYSTEM CONFIG File

5.1.1.2 VTAM User Directory

The following commands are included in the VTAM user directory:

- ***SPECIAL 800 CTCA V132A80K.*** Defines the VCTCA 800 attachment to the VSE/ESA guest V132A80K
- ***DEDICATE 200 2000.*** Dedicates the IBM 3174 controller to VTAM
- ***DEDICATE 209 2009.*** Dedicates the RISC/6000 controller to VTAM

5.1.1.3 VSE/ESA User Directory

The following command is included in the VSE/ESA user directory (V132A80K), to define the VCTCA 800 connection to VM VTAM:

SPECIAL 800 CTCA VTAM

5.1.1.4 PROFILE EXEC

To automatically connect VCTCA between VTAM and the VSE/ESA guest, one of the following commands is used:

- ***CP COUPLE 800 V132A80K 800*** in the PROFILE EXEC of VM VTAM **or**
- ***CP COUPLE 800 VTAM 800*** in the PROFILE EXEC of the VSE/ESA guest

5.1.2 Customization in VM VTAM

The following VTAM definitions in VM VTAM are required to implement our network configuration:

- Path table for the cross domain communication with the VSE/ESA guest
- CA major node to define VCTCA
- CDRM major node to define the cross domain resource managers
- LOCAL SNA major node to define the IBM 3174 and RISC/6000 PUs and LUs

5.1.2.1 VTAM Path Table

A path is required to enable the cross domain communication from VM VTAM to VSE VTAM using VCTCA. The following path is defined to the VTAM path table:

```
IPFH2TW  PATH  DESTSA=22,ER0=(22,1),ER1=(22,1),ER2=(22,1),ER3=(22,1), C
               ER4=(22,1),ER5=(22,1),ER6=(22,1),ER7=(22,1),          C
               VR0=0,VR1=1,VR2=2,VR3=3,VR4=4,VR5=5,VR6=6,VR7=7
```

Figure 21. VM VTAM Path Table

The destination and adjacent subareas must match the host subarea of the VSE/ESA guest. The transmission group, explicit and virtual routes must correspond to the definitions in the VSE/ESA guest's path table.

5.1.2.2 VTAM CTCA Major Node

A CA major node is created to define VCTCA to VTAM. Listed in Figure 22 is the definition we used:

```
         VBUILD TYPE=CA
VLG800   GROUP LNCTL=CTCA,                  ** CTCA TO V132A80K       **C
               DELAY=.020,                  ** NORMAL=DEFAULT, MAX=30  **C
               MAXBFRU=(2),                 ** NORMAL=DEFAULT, MAX=30  **C
               MIH=YES,                     ** NORMAL=DEFAULT, MAX=30  **C
               REPLYTO=25.5,                ** DEFAULT                 **C
               ISTATUS=ACTIVE               ** DEFAULT                 **
VCL800   LINE  ADDRESS=800,                 ** DEFINE C-T-C LINK ADDR. **C
               ISTATUS=ACTIVE               ** DEFAULT                 **
VPU800   PU    PUTYPE=4,                    ** DEFINE C-T-C LINK STA.  **C
               ISTATUS=ACTIVE               ** DEFAULT                 **
```

Figure 22. VM VTAM CA Major Node for VCTCA

The line address (800) must match the address defined in the VM VTAM and VSE guest user directories. PU type 4 is defined for the VCTCA connection.

5.1.2.3 VTAM CDRM Major Node

CDRM major node is used to identify the cross domain resource manager functions of the local and remote VTAMs.

Figure 23 lists the definitions we used. The first statement defines the local VTAM CDRM and the second statement defines the remote VTAM CDRM. The CDRM name is the same as the SSCPNAME in the owning VTAM start options list.

A VTAM start options list example is provided in A.2, "VSE VTAM Start Options List" on page 341.

```
         VBUILD TYPE=CDRM
IPFV2    CDRM   SUBAREA=02,CDRDYN=YES  ** SA=02 BOEVMIS2
IPFV2B   CDRM   SUBAREA=22,CDRSC=OPT   ** SA=22 V132A80K
```

Figure 23. VM VTAM CDRM Major Node

5.1.2.4 VTAM Local SNA Major Node

The IBM 3174 and OCS II Gateway PUs and LUs are defined in the Local SNA major node. Illustrated in Figure 24 on page 47 are the definitions for the controllers.

CUADDR must match the addresses defined in the VTAM user directory for these two controllers. The key parameters of the OCS II Gateway PU are discussed in 5.2.3, "Customization in VSE VTAM" on page 53.

```
         VBUILD TYPE=LOCAL
*-----------------------------------------------------------------------
*        DEFINE THE GATEWAY 3174 CONTROLLER PU
*-----------------------------------------------------------------------
IPFCP200 PU    CUADDR=200,                                             C
               DELAY=0.2,                                              C
               ISTATUS=ACTIVE,                                         C
               MAXBFRU=29,                                             C
               PUTYPE=2,                                               C
               XID=YES,                   FOR APPN SUPPORT             C
               DYNLU=YES,                                              C
               USSTAB=ISTSNA,MODETAB=ISTINCLM,DLOGMOD=D4A32782
*-----------------------------------------------------------------------
*        DEFINE COAX-ATTACHED (NON-GATEWAY) TERMINAL LUS
*-----------------------------------------------------------------------
IPFT2S02 LU    LOCADDR=02,LOGAPPL=IPFA2VSC
IPFT2S03 LU    LOCADDR=03,LOGAPPL=IPFA2VSC
    :
    :
IPFT2S32 LU    LOCADDR=32,LOGAPPL=IPFA2VSC
IPFT2S33 LU    LOCADDR=33,LOGAPPL=IPFA2VSC
*
*-----------------------------------------------------------------------
*        DEFINE THE DOWNSTREAM  RS/6000 PU ATTACHED TO THE GATEWAY
*-----------------------------------------------------------------------
IPFP2209 PU    CUADDR=209,                                             C
               DELAY=0.2,                                              C
               ISTATUS=ACTIVE,                                         C
               MAXBFRU=7,                                              C
               PUTYPE=2,                                               C
               XID=NO,                                                 C
               USSTAB=ISTSNA,MODETAB=OCSBIND,DLOGMOD=OCSLU0,           C
               PACING=0,VPACING=0,                                     C
               SECNET=YES
IPFT2S9A LU    LOCADDR=01,LOGAPPL=OCSFTPS
IPFT2S9B LU    LOCADDR=02
IPFT2S9C LU    LOCADDR=03
IPFT2S9D LU    LOCADDR=04
IPFT2S9E LU    LOCADDR=05
IPFT2S9F LU    LOCADDR=06
IPFT2S9G LU    LOCADDR=07
IPFT2S9H LU    LOCADDR=08
IPFT2S9I LU    LOCADDR=09
IPFT2S9J LU    LOCADDR=10,DLOGMOD=OCSLU0,PACING=1,VPACING=2
IPFT2S9K LU    LOCADDR=11,DLOGMOD=OCSLU0,PACING=1,VPACING=2
IPFT2S9L LU    LOCADDR=12,DLOGMOD=OCSLU0,PACING=1,VPACING=2
IPFT2S9M LU    LOCADDR=13,DLOGMOD=OCSDSC,PACING=1,VPACING=2
IPFT2S9N LU    LOCADDR=14,DLOGMOD=OCSMD2,PACING=1,VPACING=2
IPFT2S9O LU    LOCADDR=15,DLOGMOD=OCSMD2,PACING=1,VPACING=2
IPFT2S9P LU    LOCADDR=16
*
```

Figure 24. VM VTAM LOCAL SNA Major Node for IBM 3174 and RISC/6000

5.1.3 Customization in VSE/ESA

VSE/ESA is installed with the standard Environment 6 partition layout. An ADD statement is included in the IPL procedure to define VCTCA:

ADD 800,CTCA,EML

5.1.4 Customization in VSE VTAM

The following VTAM definitions in VSE VTAM are required to implement our network configuration:

- Path table for the cross domain communication with VM VTAM
- CA major node to define VCTCA
- CDRM major node to define the cross domain resource managers
- CDRSC major node to define the OCS II Gateway LUs

5.1.4.1 VTAM Path Table

A path is required to enable the cross domain communication from VSE VTAM to VM VTAM using VCTCA. The sample job in Figure 25 is used to define and catalog the VTAM path table.

```
* $$ JOB JNM=TCPPATH,DISP=D,PRI=3,NTFY=YES,DEST=*,CLASS=0
// JOB TCPPATH  CATALOG VTAM BOOK
// EXEC LIBR,PARM=¢MSHP¢
ACCESS SUBLIB=PRD2.CONFIG
CATALOG TCPPATH.B                          REPLACE=YES
*  PATH TO CONNECT TO VM/ESA
IPFHWT2  PATH  DESTSA=2,                                              C
               ER0=(2,1),ER1=(2,1),ER2=(2,1),ER3=(2,1),               C
               ER4=(2,1),ER5=(2,1),ER6=(2,1),ER7=(2,1),               C
               VR0=0,VR1=1,VR2=2,VR3=3,VR4=4,VR5=5,VR6=6,VR7=7
/+
/*
/&
* $$ EOJ
```

Figure 25. VSE VTAM Path Table

The destination and adjacent subareas must match the host subarea of VM VTAM. The transmission group, explicit and virtual routes must correspond to the definitions in VM VTAM's path table.

5.1.4.2 VTAM CTCA Major Node

The sample job in Figure 26 on page 49 is used to define and catalog VCTCA with address 800 to VTAM.

```
* $$ JOB JNM=TCPCTCA,DISP=D,PRI=3,NTFY=YES,LDEST=*,CLASS=0
// JOB TCPCTCA  CATALOG VTAM CTCA MAJOR NODE
// EXEC LIBR,PARM=¢MSHP¢
ACCESS SUBLIB=PRD2.CONFIG
CATALOG     TCPCTCA.B          REPLACE=YES
TCPCTCA  VBUILD TYPE=CA
TCPCG800 GROUP LNCTL=CTCA
TCPCL800 LINE  ADDRESS=800,ISTATUS=ACTIVE
TCPCP800 PU    PUTYPE=4,TGN=1
/+
/*
/&
* $$ EOJ
```

Figure 26. VSE VTAM CA Major Node VCTCA

5.1.4.3 VTAM CDRM Major Nodes

The sample job in Figure 27 is used to define and catalog the CDRM major node.

```
* $$ JOB JNM=TCPCDRM,DISP=D,PRI=3,NTFY=YES,LDEST=*,CLASS=0
// JOB TCPCDRM  CATALOG VTAM BOOK
// EXEC LIBR,PARM=¢MSHP¢
ACCESS SUBLIB=PRD2.CONFIG
CATALOG     TCPCDRM.B          REPLACE=YES
TCPCDRM  VBUILD TYPE=CDRM
IPFV2B   CDRM  SUBAREA=22,CDRDYN=YES,ISTATUS=ACTIVE
IPFV2    CDRM  SUBAREA=2,CDRDYN=NO,ISTATUS=ACTIVE
/+
/*
/&
* $$ EOJ
```

Figure 27. VSE VTAM CDRM Major Node

5.1.4.4 VTAM CDRSC Major Nodes

Listed in Figure 28 on page 50 is the sample job to define the OCS II Gateway LUs as cross domain resources. The CDRSC entry names must match the LU names defined in the VM VTAM Local SNA major node.

```
* $$ JOB JNM=TCPCDRS,DISP=D,PRI=3,NTFY=YES,LDEST=*,CLASS=0
// JOB TCPCDRS  CATALOG VTAM BOOK
// EXEC LIBR,PARM=¢MSHP¢
ACCESS SUBLIB=PRD2.CONFIG
CATALOG     TCPCDRS.B          REPLACE=YES
TCPCDRS  VBUILD TYPE=CDRSC
IPFT2S9A CDRSC CDRM=IPFV2      FOR OCS FTP SERVER
IPFT2S9B CDRSC CDRM=IPFV2      FOR OCS FTP CLIENT
IPFT2S9C CDRSC CDRM=IPFV2      FOR OCS FTP CLIENT
IPFT2S9D CDRSC CDRM=IPFV2      FOR OCS FTP CLIENT
IPFT2S9E CDRSC CDRM=IPFV2      FOR OCS FTP CLIENT
IPFT2S9F CDRSC CDRM=IPFV2      FOR OCS FTP CLIENT
IPFT2S9G CDRSC CDRM=IPFV2      FOR OCS FTP CLIENT
IPFT2S9H CDRSC CDRM=IPFV2      FOR OCS FTP CLIENT
IPFT2S9I CDRSC CDRM=IPFV2      FOR OCS FTP CLIENT
IPFT2S9J CDRSC CDRM=IPFV2      FOR OCS TNFS OUTBOUND
IPFT2S9K CDRSC CDRM=IPFV2      FOR OCS TNFS OUTBOUND
IPFT2S9L CDRSC CDRM=IPFV2      FOR OCS TNFS OUTBOUND
IPFT2S9M CDRSC CDRM=IPFV2      FOR OCS TNFS INBOUND
IPFT2S9N CDRSC CDRM=IPFV2      FOR OCS TNFS INBOUND
IPFT2S9O CDRSC CDRM=IPFV2      FOR OCS TNFS INBOUND
IPFT2S9P CDRSC CDRM=IPFV2      FOR OCS SAM APPLICATIONS
/+
/*
/&
* $$ EOJ
```

Figure 28. VSE VTAM CDRSC Major Node for OCS II Gateway

5.2 VSE/ESA with OCS II Gateway Attached via the Token-Ring Adapter

In this configuration, VSE/ESA communicates directly with OCS II Gateway via the IBM 9221′s Token-Ring adapter. To prepare for this connection configuration, the following environments are customized:

- IBM 9221 Token-Ring adapter
- VSE/ESA
- VSE VTAM

The following tasks must be performed before the definitions of this configuration can be tested:

- Ensure that the IBM 9221 Token-Ring adapter and OCS II Gateway are properly connected to the same IBM Token-Ring LAN
- V NET,INACT the OCS II Gateway PU in VM VTAM
- V NET,INACT the CDRSC major node in VSE VTAM

Figure 29 on page 51 provides a summary of the attachment configuration and VTAM definitions. Pointers in the diagram illustrate the relationships of the definitions and where parameter values must match. This diagram should be referred to while studying the sample definitions provided in this document.

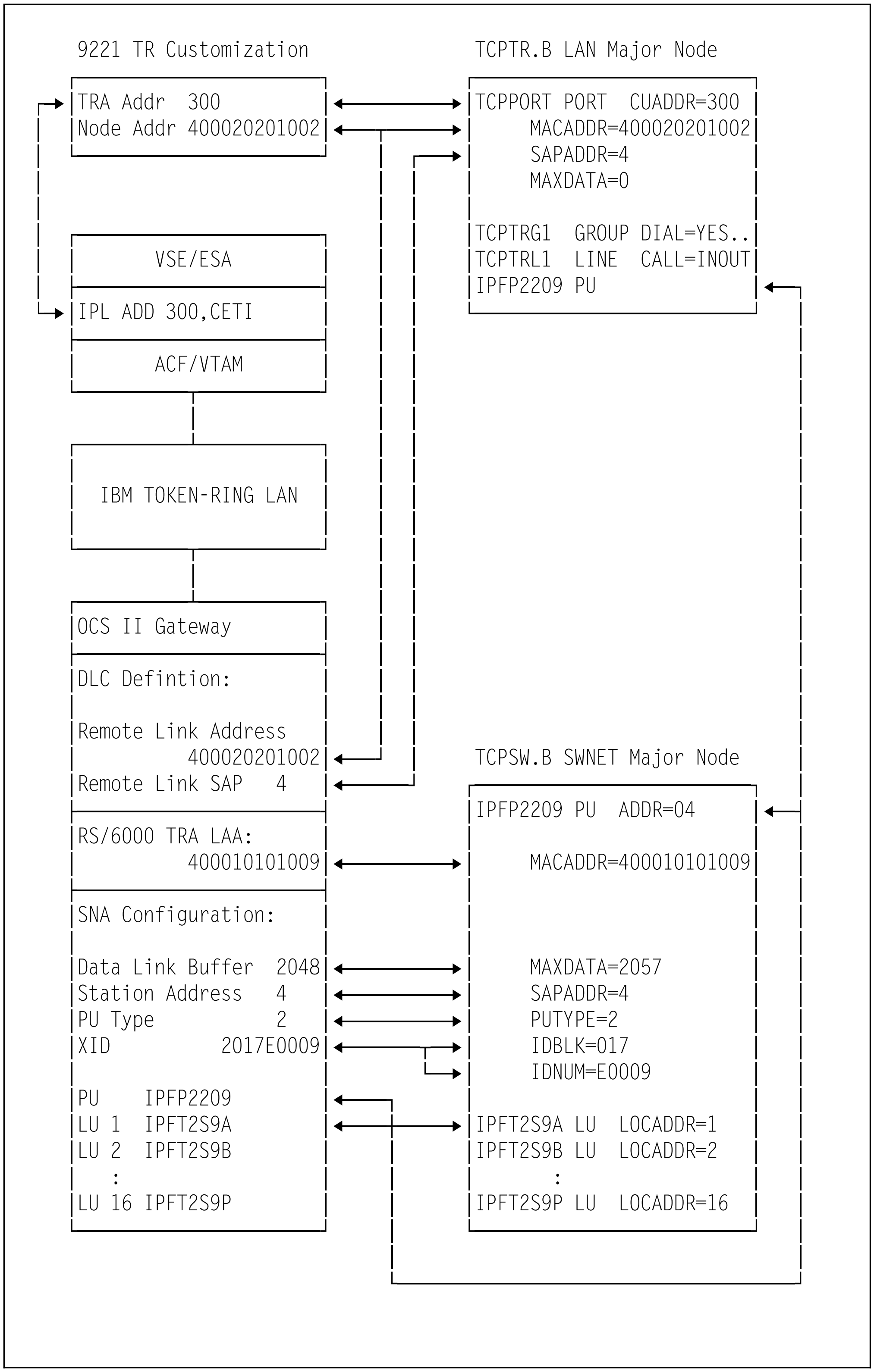


Figure 29. Configuration and Definition Summary for the TRA Connection

5.2.1 Customization of IBM 9221 Token-Ring Adapter

The unit and node address of the Token-Ring adapter are customized at the IBM 9221 processor console, using the following steps:

1. Log on as system programmer (SYSPROG)
2. Select '*Customize*' from the action bar
3. Select '*I/O Controller*'
4. Select '*Change Customization*' from the 'Action' option
5. Specify '00' for the unit address
6. Specify '400020201002' for the node address
7. Specify the correct ring speed

After the customization, the Token-Ring adapter must be re-initialized with the new data. This is accomplished by the '*Reconfigure the Adapter*' option from the '*Customize I/O Controller*' function.

5.2.2 Customization in VSE/ESA

The customization in VSE/ESA involves two tasks:

- Tailor the VSE/ESA user directory (for the VM/VSE environment)
- Tailor the IPL procedure to define the Token-Ring adapter

5.2.2.1 VSE/ESA User Directory

Since VSE/ESA is running under VM in our lab environment, the following commands are required in the VSE/ESA user directory:

- ***DEDICATE 300 0300.*** Dedicates the CETI group address to VSE/ESA
- ***DEDICATE 301 0301.*** Dedicates the CETI group address to VSE/ESA
- ***DEDICATE 302 0302.*** Dedicates the CETI group address to VSE/ESA
- ***DEDICATE 303 0303.*** Dedicates the CETI group address to VSE/ESA

This customization step is not required if VSE/ESA is running in native mode.

5.2.2.2 VSE/ESA IPL Procedure

An ADD statement is included in the IPL procedure to define the Token-Ring adapter:

ADD 300,CETI

5.2.3 Customization in VSE VTAM

To support the attachment configuration, the following VTAM definitions are required in VSE VTAM:

- A LAN major node to define the IBM Token-Ring LAN
- A Switched major node to define the OCS II Gateway

5.2.3.1 VTAM LAN Major Node

The LAN major node in Figure 30 is used to define the Token-Ring adapter and OCS II Gateway PU attached on the IBM Token-Ring LAN.

```
* $$ JOB JNM=TCPTR,DISP=D,PRI=3,NTFY=YES,LDEST=(,RSCS),CLASS=0
* $$ LST CLASS=A
// JOB TCPTR
// EXEC LIBR,PARM=¢MSHP¢
 ACCESS SUBLIB=PRD2.CONFIG
 CATALOG   TCPTR.B    REPLACE=YES
**********************************************************************
*        LAN MAJOR NODE                                              *
**********************************************************************
*
TCPTR    VBUILD TYPE=LAN
*
* DEFINITION FOR 9221 TRA
TCPPORT  PORT  CUADDR=300,            PORT ADDRESS                    C
               MACADDR=400020201002,  LAA FOR 9221 TRA                C
               LANCON=(02.0,2),       TIMER AND RETRY COUNT           C
               MAXDATA=0,             SET TO 0 FOR MAX FRAME SIZE     C
               MAXSTN=60,                                             C
               SAPADDR=4              SERVICE ACCESS POINT ADDRESS
*
* DEFINITION FOR RS/6000
*
TCPTRG1  GROUP DIAL=YES,LNCTL=SDLC
TCPTRL1  LINE  ISTATUS=ACTIVE,CALL=INOUT
IPFP2209 PU    ISTATUS=ACTIVE
/+
/*
/&
* $$ EOJ
```

Figure 30. VSE VTAM LAN Major Node

The key parameters are:

- ***CUADDR of the PORT macro.*** Must match the hardware address defined in the IBM 9221 and VSE ADD statement.
- ***MACADDR of the PORT macro.*** The medium access control (MAC) address of the IBM 9221 Token-Ring adapter. It must match the Node Address defined in the IBM 9221, and the Remote Link Address defined in the OCS II Gateway customization.
- ***MAXDATA of the PORT macro.*** The maximum length of the information field of an LPDU on the LAN. We set this value to 0, to allow for the maximum frame size supported by the IBM 9221 Token-Ring adapter.

- ***SAPADDR of the PORT macro.*** The service access point (SAP) address of this PU. The value must be a multiple of four and must match the remote link SAP in the OCS II Gateway customization.
- ***PU Name IPFP2209.*** The name of this PU. It must match the PU name in the Switched major node and OCS II Gateway.

5.2.3.2 VTAM Switched Major Node

A Switched major node is used to define the OCS II Gateway PU and LUs attached on the IBM Token-Ring LAN. Listed in Figure 31 on page 55 is an extract from the definition job, to highlight the key PU parameters. The complete Switched major node definition job is listed in A.3, "VSE TCPSW.B Switched Major Node" on page 342.

The LU parameter values are determined by the OCS applications. They are discussed in the following sections:

- 6.2, "OC/FTP Server Customization" on page 64
- 7.2, "OC/FTP Client Customization" on page 82
- 10.2, "OC/TELNET FS Customization" on page 134

```
TCPSW          VBUILD TYPE=SWNET
*
*  PU AND LU DEFINITION FOR OCS II GATEWAY
*
IPFP2209 PU    ADDR=04,MACADDR=400010101009,  LAA FOR RS/6000        C
               IDBLK=017,             IDENTIFICATION BLOCK           C
               IDNUM=E0009,           IDNUM SET BY ITSC CONVENTION   C
               PACING=0,VPACING=0,    NO PACING FOR OCS FTP          C
               PUTYPE=2,              PU-2 REQUIRED BY OCS GATEWAY   C
               MAXDATA=2057,          FROM 265 TO 2057               C
                   :
                   :
               SAPADDR=4              SERVICE ACCESS POINT ADDRESS
*   LU DEFINITION FOR RS/6000 GATEWAY
*
*   9A FOR FTP SERVER, 9B-I FOR FTP CLIENT
*   9J-L FOR TELNET CLIENT, 9M-O FOR TELNET SERVER
*   9P FOR SAM SAMPLE PROGRAMS
*
IPFT2S9A LU    LOCADDR=1,DLOGMOD=OCSLU0,LOGAPPL=OCSFTPS,             C
               ISTATUS=ACTIVE,MODETAB=OCSBIND,                       C
               MDLTAB=VTMMDL,MDLENT=VSELU2A,                         C
               SSCPFM=USSSCS,USSTAB=VTMUSSTR
IPFT2S9B LU    LOCADDR=2,DLOGMOD=OCSLU0,                             C
               ISTATUS=ACTIVE,MODETAB=OCSBIND,                       C
               MDLTAB=VTMMDL,MDLENT=VSELU2A,                         C
               SSCPFM=USSSCS,USSTAB=VTMUSSTR
                 :
                 :
IPFT2S9O LU    LOCADDR=15,DLOGMOD=OCSMD2,                            C
               ISTATUS=ACTIVE,MODETAB=OCSBIND,                       C
               MDLTAB=VTMMDL,MDLENT=VSELU2A,                         C
               PACING=1,VPACING=2,                                   C
               SSCPFM=USSSCS,USSTAB=VTMUSSTR
IPFT2S9P LU    LOCADDR=16,DLOGMOD=OCSLU0,                            C
               ISTATUS=ACTIVE,MODETAB=OCSBIND,                       C
               MDLTAB=VTMMDL,MDLENT=VSELU2A,                         C
               SSCPFM=USSSCS,USSTAB=VTMUSSTR
```

Figure 31. VSE VTAM Switched Major Node Definition

The key PU parameters are:

- ***PU name.*** The name of the PU. It must match the name specified in the LAN major node and OCS II Gateway customization.
- ***MACADDR.*** The MAC address of the RISC/6000 Token-Ring adapter. It must match the Token-Ring adapter address defined in the RISC/6000.
- ***IDBLK.*** A three-digit hexadecimal number to identify the device. A value of ′017′ is required by OCS II Gateway.
- ***IDNUM.*** A user assigned five-digit hexadecimal number to identify the device. It must match the IDNUM specification in the ′*PU Definition*′ panel of the OCS II Gateway customization.
  The PU type, IDBLK and IDNUM make up the XID field in the OCS II Gateway configuration. Please refer to Figure 136 on page 179 for details.
- ***PACING and VPACING.*** LU pacing parameters to control the VTAM pacing values between LUs. They are placed in the PU to allow the specifications to sift-down to the LUs. Both parameters are set to 0 for the FTP LUs. These values are overridden in the OC/TELNET FS LUs definition.
- ***PUTYPE.*** The PU type of this controller. PU type 2 is required by OCS II Gateway.
- ***MAXDATA.*** The maximum PIU size that this PU can receive. The value is the maximum RU size plus nine bytes (for the TH and RH). The maximum RU size must match the data link buffer size in the OCS II Gateway configuration listing. Please refer to 12.4.6, "Generate a Configuration Listing" on page 178 for the configuration listing.
- ***SAPADDR.*** The SAP address of this PU. It must match the station address in the OCS II Gateway customization.

5.3 VSE/ESA with OSCII Gateway Attached via the OCC

In this configuration, VSE/ESA communicates directly with OCS II Gateway via the OpenConnection for Channel adapter. To prepare for this connection configuration, the following environments are customized:

- OpenConnection for Channel
- VSE/ESA
- VSE VTAM

Figure 32 on page 57 provides a summary of the attachment configuration and VTAM definitions. This diagram should be referred to while studying the sample definitions provided in this document.

```
        9221
┌────────────────────────┐
│                        │
│                        │
├────────────────────────┤
│ Channel Address 2201   │
├────────────────────────┤
│                        │
│                        │
├────────────────────────┤
│      VSE/ESA           │
├────────────────────────┤
│ IPL ADD 901,3791L      │
│                        │
├────────────────────────┤
│      ACF/VTAM          │
│                        │
│  IPFT2901 PU CUADDR=901│
│  IPFT2S9A LU LOCADDR=1 │
│   .                    │
│   .                    │
│  IPFT2S9P LU LOCADDR=16│
│                        │
└────────────────────────┘
          │  │ Channel
          │  │
┌────────────────────────┐
│        OCC             │
│                        │
└────────────────────────┘
      SCSI   │
             │
┌────────────────────────┐
│   OCS II Gateway       │
├────────────────────────┤
│   SCA Definitions      │
│                        │
│   PU IPFP2901          │
│   LU  1 IPFT2S8A       │
│    .                   │
│    .                   │
│   LU 16 IPFT2S9P       │
├────────────────────────┤
│   Data Link Buffer 265 │
│   Station Address 4    │
│   PU Type 2            │
└────────────────────────┘
```

Figure 32. Configuration and Definition Summary for the OCC Connection

5.3.1 Customization of OpenConnection for Channel

The OpenConnection for Channel adapter configuration consists of the following steps:

- Configure the OCC adapter by selecting the OCC SCSI address (4) using the OCC front panel menu.
- Configure OCC Hardware in the AIX operating system using SMIT.

For detailed information on the OCC hardware configuration please see 16.1.3, "Configuring OCC Hardware" on page 210.

5.3.2 Customization in VSE/ESA

The customization in VSE/ESA involves two tasks:

- Tailor the VSE/ESA user directory (for the VM/VSE environment)
- Tailor the IPL procedure to define the OpenConnection for Channel

5.3.2.1 VSE/ESA User Directory

Since VSE/ESA is running under VM in our lab environment, the following command is required in the VSE/ESA user directory:

- ***DEDICATE 901 2201.*** Dedicates the device address to VSE/ESA

This customization step is not required if VSE/ESA is running in native mode.

5.3.2.2 VSE/ESA IPL Procedure

An ADD statement is included in the IPL procedure to define OpenConnection for Channel:

ADD 901,3791L

5.3.3 Customization in VSE VTAM

To support the attachment configuration, the following VTAM definition is required in VSE VTAM:

1. A PU definition to define the channel.

The PU in Figure 33 on page 59 is used to define the OpenConnection for Channel.

```
OCSOCC   VBUILD TYPE=LOCAL
**********************************************************************
*        DEFINE THE GATEWAY OCC CONTROLLER PU FOR OCS/TCP
**********************************************************************
IPFP2901 PU    CUADDR=901,                                            C
               ISTATUS=ACTIVE,                                        C
               MAXBFRU=1,                                             C
               PUTYPE=2,                                              C
               PACING=0,VPACING=0
*
```

Figure 33. PU Definition for the OCC Connection

Chapter 6. OC/FTP Server Installation and Customization

This chapter covers the installation and customization of OC/FTP Server. It consists of two steps:

1. Installation - Describes the preparation and procedures to install OC/FTP Server in VSE/ESA

2. Customization - Provides a detailed explanation of the product customization

6.1 OC/FTP Server Installation

There are two major steps involved to install the product:

1. Prepare the installation environment for the product

2. Install the product from tape to the VSE/ESA sublibrary

6.1.1 Prepare the Installation

The installation preparation for OC/FTP Server involves the definitions of:

- OCS II Gateway to VSE/ESA

  Please refer to Chapter 5, “Defining the OCS II Gateway to the Host” on page 41 for the setup instructions.

- The VSE/ESA library and sublibraries for the product installation

  Library TCPOCS is defined using the VSE/ESA Interactive Interface (II) ′*Define a Library*′ panel.

  Sublibrary TCPOCS.FTPS is defined using the LIBR DEF SUB command in ICCF. This sublibrary is used to store the OC/FTP Server product, and the FTP server customization.

- VSE/ESA user sublibraries for the FTP clients

  This is an optional step. The numbers and names of the sublibraries to be used depend on the installation′s requirements. You may define as many user sublibraries as needed, and assign them to the FTP clients. The default assignment for the FTP clients is discussed in 6.2.3.2, “SITE Member” on page 69.

  We defined the following user sublibraries for our FTP clients: TCPOCS.USER, TCPOCS.USR1, TCPOCS.USR2, TCPOCS.USR3, TCPOCS.USR4, TCPOCS.USR5, TCPOCS.USR6, TCPOCS.USR7, TCPOCS.USR8, TCPOCS.USR9, and TCPOCS.USR10.

  Figure 34 on page 62 illustrates the TCPOCS library information that we defined. The space utilization of the OCS products may vary from release to release, and you should not use it for your library space estimation.

- Changes

  Any necessary/optional changes to your system are marked by ″$CHGOPT$″ for optional, and ″$CHGREQ$″ for required modifications. These changes are for system configurations, and **MUST** be completed, where mandatory.

```
O LD L=TCPOCS O=ST
BG 0000
BG 0000 STATUS  DISPLAY          LIBRARY=TCPOCS              DATE: 96-07-04
BG 0000                                                      TIME: 11:54
BG 0000 --------------------------------------------------------------------
BG 0000 FILE-ID        : VSE.TCPOCS.LIBRARY
BG 0000 CREATION DATE  : 96-05-29   13:13
BG 0000 SUBLIBRARIES   :       26
BG 0000
BG 0000 LOCATION (VSAM): DEVICE=9336  VOLID=SYSWK1 BLK# =  242176 -   272895
BG 0000                  DEVICE=9336  VOLID=SYSWK1 BLK# =  281472 -   292031
BG 0000                  DEVICE=9336  VOLID=SYSWK1 BLK# =  292032 -   302591
BG 0000
BG 0000 LIBRARY BLOCK  : SIZE=  1024 BYTES   DATA SPACE=   988 BYTES
BG 0000
BG 0000 TOTAL   SPACE  :   25920 LIBRARY BLOCKS   (100 %)
BG 0000 USED    SPACE  :   10214 LIBRARY BLOCKS   ( 39 %)
BG 0000 DELAYED SPACE  :     364 LIBRARY BLOCKS   (  1 %)
BG 0000 FREE    SPACE  :   15342 LIBRARY BLOCKS   ( 60 %)
BG 0000
BG 0000 --------------------------------------------------------------------
BG 0000 SUBLIBRARY   CREATION   SPACE     NO. OF    USED   DELAYED   % LIBR.
BG 0000              DATE       REUSAGE   MEMBERS   LB¢S   LB¢S      SPACE
BG 0000 --------------------------------------------------------------------
BG 0000 FTPC         96-05-29   AUTO        95      1446        0       6 %
BG 0000 FTPS         96-05-29   AUTO       100      1299        2       5 %
BG 0000 HUGO         96-05-29   IMMED        1         2        0       0 %
BG 0000 LPD          96-05-29   AUTO        37       547        0       2 %
BG 0000 PARMLIB      96-06-17   AUTO        11       185      362       2 %
BG 0000 RSH          96-05-29   AUTO        36       478        0       2 %
BG 0000 SAMMAC       96-06-24   AUTO       176       527        0       2 %
BG 0000 SAMOBJ       96-06-24   AUTO       101       818        0       3 %
BG 0000 SAMSMP       96-06-24   AUTO        96       647        0       2 %
BG 0000 TNFS         96-05-29   AUTO       622      3504        0      14 %
BG 0000 USER         96-05-29   AUTO        17       735        0       3 %
BG 0000 USR1         96-05-29   AUTO         2         8        0       0 %
BG 0000 USR10        96-05-29   AUTO         0         1        0       0 %
BG 0000 USR2         96-05-29   AUTO         0         1        0       0 %
BG 0000 USR3         96-05-29   AUTO         0         1        0       0 %
BG 0000 USR4         96-05-29   AUTO         0         1        0       0 %
BG 0000 USR5         96-05-29   AUTO         0         1        0       0 %
BG 0000 USR6         96-05-29   AUTO         0         1        0       0 %
BG 0000 USR7         96-05-29   AUTO         0         1        0       0 %
BG 0000 USR7         96-05-29   AUTO         0         1        0       0 %
BG 0000 USR8         96-05-29   AUTO         0         1        0       0 %
BG 0000 USR9         96-05-29   AUTO         0         1        0       0 %
```

Figure 34. VSE/ESA Library and Sublibraries for OCS Products

- VSAM catalog and space definitions for the FTP clients

  A VSAM catalog named TCPCAT is defined using the VSE/ESA II ′*Define a New User Catalog*′ panel. VSAM space is defined to the catalog using the VSE/ESA II ′*Display or Process a Catalog, Space*′ panel.

  This is an optional step. We defined the catalog to hold user VSAM data sets for the FTP clients. You may define a new or use an existing catalog for the FTP clients, depending on the installation′s requirements. You may assign the catalog to the FTP clients as a default catalog. The default assignment for the FTP clients is discussed in 6.2.3.2, "SITE Member" on page 69.

6.1.2 Installation Procedures

Installation of the product from tape to the product sublibrary involves the following steps:

- Use the VSE/ESA II '*Prepare for Installation (Stacked Tapes Only)*' panel to generate the job to scan the product tape, as illustrated in Figure 35.

```
* $$ JOB JNM=OCSPRE,DISP=H,PRI=3,NTFY=YES,LDEST=*,CLASS=0
// JOB OCSPRE  SCAN OPTIONAL PRODUCT TAPE
// LIBDEF PHASE,SEARCH=(PRD1.BASE,IJSYSRS.SYSLIB)
* *
* *     PREPARE ADDITIONAL PROGRAM INSTALLATION
* *     -  SCAN PROGRAM TAPE
* *
// ASSGN SYS006,184
// EXEC DTRIPRE,PARM=¢ADDR=184¢
/*
/&
* $$ EOJ
```

Figure 35. Sample Job to Scan Product Tapes

- Submit the job to VSE/ESA and review the library space requirements of the product.
- Use the VSE/ESA II '*Install Product(s) from Tape*' panel to generate the job to install the product into sublibrary TCPOCS.FTPS.

  Note

  With OC/FTP software Version 2.2 and later, both OC/FTP Client and OC/FTP Server can be loaded into the same *library.sublibrary*

  The VSE/ESA II panel also displays the minimum library space requirements for the product.
- Submit the job to VSE/ESA to install the product. Our installation step took less than 10 minutes.

There is no installation verification job provided by the software. You may use the sample job in Figure 36 on page 64, to verify that the product is recorded by MSHP. Testing of the product can only be performed after the product customization. Please refer to Chapter 18, "OC/FTP Server Operation and Examples" on page 219 for the testing and operation procedures.

```
* $$ JOB JNM=RETRACE,DISP=D,PRI=9,NTFY=YES,LDEST=(,RSCS),CLASS=0
* $$ LST CLASS=A
// JOB RETRACE
* -------------------------------------------------------------------
* RETRACE PRODUCTS         ****   FROM SYSTEM HISTORY FILE
* -------------------------------------------------------------------
// ASSGN SYS018,DISK,VOL=DOSRES,SHR
// DLBL IJSYSHF,¢VSE.SYSTEM.HISTORY.FILE¢,99/365,SD
// EXTENT SYS018,DOSRES,1,0,6135,75
// EXEC MSHP
   RETRACE PRODUCTS
/*
/&
* $$ EOJ
```

Figure 36. Sample Retrace Product Job

6.2 OC/FTP Server Customization

The customization is based on the samples in PRODBLDS.PROC provided by OC/FTP Server. After the product is installed in the product sublibrary, PRODBLDS.PROC is punched into an ICCF library, using the following command from your VSE/ICCF command mode screen:

LIBRP TCPOCS.FTPS PRODBLDS.PROC PRODBLDS

We then edited it into individual jobs to tailor the OC/FTP Server environment.

The OC/FTP Server customization consists of four steps:

- VSAM customization
- VTAM customization
- FTP Server customization
- Startup customization

6.2.1 VSAM Customization

When an FTP client wants to put a file into a VSAM file, OC/FTP Server requests VSAM to implicit-define the file in the catalog assigned. VSAM implicit-define requires a model in the catalog to define the data set. It will use the respective ESDS, SAM or KSDS model for the definition according to the file type needed.

The job in Figure 37 on page 65 defines models to the catalog TCPIP.USER.CATALOG.

```
* $$ JOB JNM=DEFMDL,DISP=D,CLASS=A
// JOB DEFMDL - DEFINE DEFAULT MODEL FOR ESDS, SAM & KSDS
// EXEC PGM=IDCAMS,SIZE=IDCAMS
      DEFINE CLUSTER(NAME(DEFAULT.MODEL.ESDS) +
        RECORDSIZE(8 8) +
        RECORDS(8 8) +
        VOLUMES(SYSWK4)   /* $CHGOPT$ */ +
      NOALLOCATION NERAS NIXD RUS SHR(2 4) SPEED) +
      DATA(NAME(DEFAULT.MODEL.ESDS.DATA)) +
      CATALOG(TCPIP.USER.CATALOG)
 /*  */
      DEFINE CLUSTER(NAME(DEFAULT.MODEL.ESDS.SAM) +
        RECORDFORMAT(U) +
        RECORDSIZE(8 8) +
        RECORDS(8 8) +
        VOLUMES(SYSWK4)   /* $CHGOPT$ */ +
      NOALLOCATION NERAS NIXD RUS SHR(2 4) SPEED) +
      DATA(NAME(DEFAULT.MODEL.ESDS.SAM.DATA)) +
      CATALOG(TCPIP.USER.CATALOG)
 /*  */
      DEFINE CLUSTER(NAME(DEFAULT.MODEL.KSDS) +
        RECORDSIZE(8 8) +
        RECORDS(8 8) +
        VOLUMES(SYSWK4)   /* $CHGOPT$ */ +
      KEYS(1 0) +
      NOALLOCATION NERAS IXD RUS SHR(2 4) SPEED) +
      DATA(NAME(DEFAULT.MODEL.KSDS.DATA)) +
      INDEX(NAME(DEFAULT.MODEL.KSDS.INDEX)) +
      CATALOG(TCPIP.USER.CATALOG)
/*
/&
* $$ EOJ
```

Figure 37. ESDS, SAM and KSDS Model Definitions for OC/FTP Server

The key parameters to tailor are:

- ***RECORDSIZE.*** The most commonly used record length.
- ***RECORDS.*** The primary and secondary space allocation.
- ***RECORDFORMAT.*** The record format of the VSAM SAM file.
- ***KEYS.*** The keylength and key start position of the VSAM KSDS file.
- ***CATALOG.*** The user catalog name assigned.

The VSAM data set definition parameters may be overridden by the SITE member and command. Please refer to 6.2.3.2, “SITE Member” on page 69 and Chapter 18, “OC/FTP Server Operation and Examples” on page 219 for details.

If you are using the VSAM.USER.CATALOG, do **not** delete/define the VSAM ESDS.SAM model cluster. It is already defined during VSE/ESA installation and used as workfile.

6.2.2 VTAM Customization

The following customization tasks are required in VTAM:

- Define the logmode table and entry for the application and LU
- Specify the APPL definition for OC/FTP Server
- Add the LU definition for the OCS II Gateway LU

6.2.2.1 Logmode Table OCSBIND

The sample job listed in A.6, "OCSBIND MODETAB for OCS Products" on page 351 creates the OCSBIND logmode table. There are three entries in this table used by the OCS products. OC/FTP Server uses the entry listed in Figure 38:

```
OCSLU0   MODEENT  LOGMODE=OCSLU0,                                C
                  FMPROF=X¢06¢,                                  C
                  TSPROF=X¢03¢,                                  C
                  PRIPROT=X¢00¢,                                 C
                  SECPROT=X¢00¢,                                 C
                  COMPROT=X¢0000¢,                               C
                  RUSIZES=X¢F8F8¢,                               C
                  PSERVIC=X¢000000000000000000000000¢
```

Figure 38. MODETAB Entry for OC/FTP Server

6.2.2.2 VTAM APPL Definition

The completed VTAM APPL major node definition is listed in A.4, "VTAM APPL Major Node for OCS Products" on page 344.

The APPL entry below is used by OC/FTP Server:

```
OCSFTPS  APPL  MODETAB=OCSBIND,DLOGMOD=OCSLU0,AUTH=(NVPACE,ACQ),EAS=2
```

Figure 39. VTAM APPL Statement for OC/FTP Server

Key parameters are:

- ***MODETAB and DLOGMOD.*** Identify the appropriate logmode table entry.
- ***NVPACE in AUTH parameter.*** Turns off VTAM pacing for FTP.
- ***ACQ in AUTH parameter.*** Allows OC/FTP Server to acquire LUs.
- ***EAS=2.*** Allows two concurrent sessions for OC/FTP Server.

6.2.2.3 VTAM LU Definition

The completed OCS II Gateway LUs definition is listed in Figure 24 on page 47 and A.3, "VSE TCPSW.B Switched Major Node" on page 342.

Figure 40 on page 67 lists the LU definitions used by OC/FTP Server:

```
IPFT2S9A LU    LOCADDR=1,DLOGMOD=OCSLU0,LOGAPPL=OCSFTPS,             C
               ISTATUS=ACTIVE,MODETAB=OCSBIND,                       C
               MDLTAB=VTMMDL,MDLENT=VSELU2A,                         C
               SSCPFM=USSSCS,USSTAB=VTMUSSTR
```

Figure 40. Logical Unit Definition for OC/FTP Server

The key parameters are:

- ***LU name.*** Must match the LU name in the OCS II Gateway customization.
- ***LOCADDR.*** Must match the LU number in the OCS II Gateway customization.
- ***MODETAB and DLOGMOD.*** Identify the appropriate logmode table entry.
- ***LOGAPPL.*** Logs on to OC/FTP Server when the LU is activated.

6.2.3 FTP Server Customization

The following members are tailored for the FTP server function of OC/FTP Server:

- ***CONFIG.*** Relates the FTP server function to VTAM, and defines the TCP/IP port used by OC/FTP Server.
- ***SITE.*** Defines the default FTP file values.
- ***PASSWD.*** Defines the FTP user IDs and default working environments.

These three members are filed in sublibrary TCPOCS.FTPS.

The alternative to using the product library is to define a **Parameter Library** as a VSE librarian sublibrary. We defined TCPOCS.PARMLIB to catalog the OpenConnect products′ configuration members. The OpenConnect Applications, such as OC/TELNET FS, OC/FTP Server, OC/FTP Client, OC/LPD, OC/SAM, and OC/RSH are using the same members for the site configuration. Defining a single parameter library will reduce installation and maintenance effort, as only the necessary entries for the application programs need to be added. The following DLBL statement specifies the parameter library, used with FTP Client and OC/LPD:

```
// DLBL name,¢Library.sublibrary.membertype¢
// DLBL OCSPLIB,¢TCPOCS.PARMLIB.FTP¢
```

Figure 41. Parameter Library DLBL

6.2.3.1 CONFIG Member

The CONFIG member is used to define the VTAM ACB name, TCP/IP FTP port number and the pacing values.

Figure 42 on page 68 lists the job we tailored:

```
* $$ JOB JNM=SFCONFIG,DISP=D,PRI=3,NTFY=YES,LDEST=(,RSCS),CLASS=0
* $$ LST CLASS=A
// JOB SFCONFIG
// EXEC LIBR,PARM=¢MSHP¢
ACCESS SUBLIB=TCPOCS.FTPS
CATALOG CONFIG.TCPIP EOD=$$ REPLACE=Y
ACBNAME      OCSFTPS          DEFAULT VTAM ACB NAME
ETRACE       NO               INITIAL TRACE VALUE
PORT         21               COMMUNICATION PORT FOR FTP SERVER
PACEIN       50               INBOUND PACING
PACEOUT      50               OUTBOUND PACING
$$
/*
/&
* $$ EOJ
```

Figure 42. CONFIG Member for OC/FTP Server

The key parameters are:

- ***ACBNAME.*** Relates the FTP server function to VTAM. It must match the name defined for OC/FTP Server in the VTAM APPL statement.
- ***ETRACE.*** Trace of packet option. You can also turn on the trace from the FTP session using the FTP client command - ′trace on′. All the packet trace output will be appended to the output of the OC/FTP Server startup job.
- ***PORT.*** Specifies the OC/FTP Server listen port. We have used the TCP/IP standard FTP listen port 21 for the VSE OC/FTP Server. If you also want to access the AIX′s FTP Server, you need to change the listen port number of the AIX FTP Server. How to set up the AIX FTP Server to use another listen port number is described in 12.6, "Set up Alternative Ports for AIX TELNET and FTP Servers" on page 186.

  If you set up this port number to be a number greater than 2000, the AIX FTP Server port number can be left unchanged with the standard FTP listen port 21. However, from an FTP Client, you need to specify the port number when you open an FTP session to the VSE OC/FTP Server.

  So, keep in mind that the port number for the VSE and AIX FTP Servers have to be different. In our case, we preferred to connect to the VSE FTP Server without specifying a port number.
- ***PACEIN and PACEOUT.*** These two parameters are used to implement application pacing between the OC/FTP Server programs in the host and OCS II Gateway control unit.

  Pacing is the mechanism to control and limit the traffic flow between senders and receivers. It is important for some network functions that have the potential to use a lot of network resources, if uncontrolled. FTP may transfer large files over the network, and is considered a candidate for pacing control.

  The OCS pacing implementation is conceptually similar to the VTAM one-stage pacing between LUs. Note that it is implemented at the application level, and does not affect VTAM pacing. It allows the user to limit the traffic between OC/FTP Server and OCS II Gateway, **in addition** to the VTAM LU-LU pacing.

  – PACEIN

Defines the number of RUs OCS II Gateway may send to OC/FTP Server, before waiting for a pacing response from OC/FTP Server.

- PACEOUT

  Defines the number of RUs OC/FTP Server may send to OCS II Gateway, before waiting for a pacing response from OCS II Gateway. OCS II Gateway will not return the pacing response until the data is sent to the TCP/IP end node.

In general, it is more important to control the outbound pacing for the following reasons:

- The resources (for example, cycles and data buffers) are limited in OCS II Gateway
- Usually the data flow between OC/FTP Server and OCS II Gateway is faster than within the TCP/IP network

• ***FTPSXIT1.*** Specify YES or NO for the user′s Access Control Exit. OC/FTP Server library comes with a sample program ′FTPSXIT1.SAMPLE′. You can modify this program for example to:

  - Send messages to specific users when they log on
  - Restrict a specific user(s) from logon
  - Compute the logon duration of user(s)
  - Restrict access to VSAM, Libraries and POWER queues

  If YES is specified this phase will be loaded at the start of the FTP session and deleted when the session ended.

  The default for FTPSXIT1 is NO.

6.2.3.2 SITE Member

The SITE member is used to override the OC/FTP Server default values for the VSAM, VSE/ESA library and POWER environment. The default values are used for the VSAM implicit-define and file transfer, if they are not overridden by the TCP/IP FTP clients using the QUOTE SITE command. Please refer to 18.1.2, "TCP/IP FTP Clients" on page 220 for the explanation of the QUOTE SITE command.

The parameters, descriptions and default values provided by OC/FTP Server are listed in Table 5 on page 70.

Figure 43 on page 70 lists the job we tailored for our environment.

Error in SITE member

We were unable to specify the new VSAM-KSDS and VSAM-SAM parameters. When we specified parameters such as CISIZEDATA or KEYLENGTH in the SITE member, we received a SITE parameter error during OC/FTP Server startup.

Note

If errors are found in the SITE member, the entire member is ignored and the default values are used.

Parameter	Description	Default
CATALOG	Default VSAM catalog	VSESPUC
PALLOC	The primary record allocation for VSAM implicit-define ESDS	512
SALLOC	The secondary record allocation for VSAM implicit-define ESDS	256
VSAMRECL	The maximum VSAM record length	256
CISIZE	The VSAM CI size used for ESDS	4096
MBRTYPE	The default LIBR member type	TCPIP
RDRCLASS	The default POWER job class	A
LSTCLASS	The default POWER LST class	A
PUNCLASS	The default POWER PUN class	A
LSTFORM	The default POWER LST form number	
PUNFORM	The default POWER PUN form number	
RDRPRI	The default POWER job priority	3
LSTPRI	The default POWER LST priority	3
PUNPRI	The default POWER PUN priority	3
SYSID	The default SYSID for POWER	N
UPCASE	Data translated to upper case for POWER	YES
RESTART	The default restart specification	OFF

Table 5. SITE Member Parameters and Default Values

```
* $$ JOB JNM=SFSITE,DISP=D,PRI=3,NTFY=YES,LDEST=(,RSCS),CLASS=0
* $$ LST CLASS=A
// JOB SFSITE
// EXEC LIBR,PARM=¢MSHP¢
ACCESS SUBLIB=TCPOCS.FTPS
CATALOG SITE.TCPIP EOD=$$ REPLACE=Y
CATALOG    TCPCAT        DEFAULT VSAM CATALOG
PALLOC     1024          DEFAULT PRIMARY ALLOCATION IN RECORD
SALLOC     512           DEFAULT SECONDARY ALLOCATION IN RECORD
VSAMRECL   132           DEFAULT VSAM RECORD LENGTH
CISIZE     4096          DEFAULT VSAM CISIZE FOR DATA
$$
/*
/&
* $$ EOJ
```

Figure 43. SITE Member for OC/FTP Server

Key parameters we tailored are:

- ***CATALOG.*** Defines the user catalog in our environment.
- ***PALLOC.*** Increases the VSAM primary allocation to 1000 records.

- ***SALLOC.*** Increases the VSAM secondary allocation to 500 records.
- ***VSAMRECL.*** Sets the maximum file record length as 132.
- ***CISIZE.*** Reduces the VSAM CI size to 4K.

6.2.3.3 PASSWD Member

The PASSWD member is used to define the FTP client user IDs and their default working directories. The command in the PASSWD member has the following syntax:

Userid Password [LIBR [library[.sublibrary]]]

Userid Password [VSAM [catalog [PREFIX node[.node]]]]

Userid and Password are required parameters to which you assign user names and passwords. The maximum length for both parameters is 32 characters. The other parameters are optional.

LIBR and VSAM are access modes for the user:

- ***LIBR for VSE/ESA library access.***

  It may be followed by an optional library or library.sublibrary name.

  If library or library.sublibrary name is specified, it becomes the default working directory when this user logs on to OC/FTP Server.

- ***VSAM for VSAM files access.***

  It may be followed by an optional catalog name and/or PREFIX nodes.

  If a catalog name is specified, it becomes the default catalog for the user.

  Keyword PREFIX specifies the file node prefix to be used as the default working directory for this user. For example, if we have the following files in the catalog:

  TESTING.FILE1

  TESTING.FILE2

  TESTING.FILE3

  TESTING.FILE4

  TESTING.FILE5

  We can use the PREFIX keyword followed by ′TESTING′. This specification defines TESTING as the working directory when users log on to OC/FTP Server.

Error in PASSWD member

We specified access mode VSAMKSD for a user ID; when this user logged on to OC/FTP Server, he got access mode VSAMSAM instead. This is also true for access mode VSESAM. Therefore, the user profile in the PASSWD member should still adhere to the old version′s format. The new VSAM features have not been added.

The user IDs we defined are listed in Figure 44 on page 72. The user IDs and passwords should not normally be the same. It is set this way in our environment for ease of operation.

- User ID USER has library TCPOCS as the default library
- User IDs USR1-USR7 use library access and have their own sublibraries assigned
- User ID USR8 uses VSAM access, and has no default working directory assigned
- User ID USR9 uses VSAM access. The default catalog is TCPCAT, and the working directory assigned is VTMBK
- User ID USR10 uses VSAM access and has VSESPUC as a default catalog
- User IDs JYIU, KAMI, MARU, PRAM and WACK use library access and have TCPOCS as the default library
- User ID NEWUSER ANYPSWD VSAM VSESPUC will allow VSE/Interactive Interface user additions without the need to synchronize the PASSWD member.

```
* $$ JOB JNM=SFPASSW,DISP=D,PRI=3,NTFY=YES,LDEST=(,RSCS),CLASS=0
* $$ LST CLASS=A
// JOB SFPASSW
// EXEC LIBR,PARM=¢MSHP¢
ACCESS SUBLIB=TCPOCS.FTPS
CATALOG PASSWD.TCPIP EOD=$$ REPLACE=Y
*
* NAME PASSWORD (LIBR LIB.SUBLIB ] VSAM CATALOG)
*
USER USER LIBR TCPOCS
USR1 USR1 LIBR TCPOCS.USR1
USR2 USR2 LIBR TCPOCS.USR2
USR3 USR3 LIBR TCPOCS.USR3
USR4 USR4 LIBR TCPOCS.USR4
USR5 USR5 LIBR TCPOCS.USR5
USR6 USR6 LIBR TCPOCS.USR6
USR7 USR7 LIBR TCPOCS.USR7
USR8 USR8 VSAM
USR9 USR9 VSAM TCPCAT PREFIX VTMBK
USR10 USR10 VSAM VSESPUC
JYIU BYIU LIBR TCPOCS
KAMI KAMI LIBR TCPOCS
MARU MARU LIBR TCPOCS
PRAM PRAM LIBR TCPOCS
WACK WACK LIBR TCPOCS
NEWUSER ANYPSWD VSAM VSESPUC
$$
/*
/&
* $$ EOJ
```

Figure 44. PASSWD Member for OC/FTP Server

Security Note

Note that the PASSWD member cannot have encrypted passwords. The member is a plain text file with all the user entries.

OCS supports the standard VSE/ACF interface, you can restrict the user from accessing unauthorized VSAM and Library files during his or her FTP session. To be able to use the VSE/ACF interface, you have to code DTSECTAB table and SET SEC=YES during the IPL of VSE. For more information on implementing the VSE/ACF interface, please refer to the appropriate VSE documentation.

On the other hand, if you have modified the user′s Access Control program ′FTPSXIT1′ to reflect your specific environment, you can easily turn on this type of security from the CONFIG member. Please refer to 6.2.3.1, "CONFIG Member" on page 67.

6.2.4 Startup Customization

The following preparation tasks are required to bring up OC/FTP Server:

- Build the executable phases
- Run the site protection modification job
- Create the FTP Server startup job

6.2.4.1 Build the Executable Phase

Figure 45 on page 74 lists the job we used to assemble and link-edit the OC/FTP Server phases into the TCPOCS.FTPS sublibrary. The source of this job is obtained from PRODBLDS.PROC.

```
* $$ JOB JNM=SFASM1,CLASS=A,DISP=D
// JOB SFASM1
*  ---------------------------------------- *
*  BUILDING COMPONENT EXECUTION MODULE(S).   *
*  ---------------------------------------- *
// LIBDEF SOURCE,SEARCH=TCPOCS.FTPS,TEMP
// LIBDEF OBJ,SEARCH=TCPOCS.FTPS,TEMP
// LIBDEF PHASE,SEARCH=TCPOCS.FTPS,CATALOG=TCPOCS.FTPS,TEMP
// OPTION CATAL,NOXREF
/*
   PHASE FTPSMAIN,*,NOAUTO
    INCLUDE SVRTVT
    INCLUDE FTPSMAIN
    INCLUDE SVRTRNS
    INCLUDE SVRSNAP
    INCLUDE SVRSCAN
    INCLUDE SVRNDF
    INCLUDE SVRCPU
    INCLUDE SVRIOS
    INCLUDE OCSSMH
    INCLUDE MTKMSG
    INCLUDE MTKCWD
    INCLUDE MTKPORT
    INCLUDE MTKRETR
    INCLUDE MTKSTOR
    INCLUDE MTKLIST
    INCLUDE OCSPOWER
    INCLUDE OCSPARSL
    INCLUDE MTKPWD
    INCLUDE MTKTYPE
    INCLUDE MTKSITE
    INCLUDE MTKDELE
    INCLUDE MTKHELP
    INCLUDE MTKUSER
    INCLUDE MTKPASS
    INCLUDE MTKQUIT
    INCLUDE MTKSTRU
    INCLUDE MTKMODE
    INCLUDE MTKRNFR
    INCLUDE MTKRNTO
    INCLUDE MTKNOOP
    INCLUDE MTKXMKD
    INCLUDE MTKXRMD
    INCLUDE MTKSTOP
    INCLUDE MTKTRAC
    INCLUDE MTKALLO
```

Figure 45 (Part 1 of 2). Build the Executable Phases for OC/FTP Server

```
    INCLUDE MTKCONF
    INCLUDE MTKOPER
    INCLUDE OCSLIH
    INCLUDE OCSDRH
    INCLUDE OCSIRH
    INCLUDE OCSDAH
    INCLUDE OCSPLH
    INCLUDE OCSNPL
    INCLUDE OCSVPL
    INCLUDE OCSGSV
/*
// EXEC PGM=ASSEMBLY
   PRINT ON,NOGEN,NODATA
   PRMOD RECFORM=FIXUNB,CTLCHR=ASA,WORKA=YES
   END
/*
   ENTRY FTPSMAIN
// EXEC PGM=LNKEDT,PARM=¢MSHP¢
/*
// EXEC PGM=LIBR,PARM=¢MSHP;ACCESS SUBLIB=TCPOCS.FTPS¢
  DELETE MTKSAM.PHASE
/*
   PHASE SVRSAM,*
    INCLUDE SVRSAM
    INCLUDE MTKEXITS
/*
   ENTRY SVRSAM
/*
// EXEC PGM=LNKEDT,PARM=¢MSHP¢
/*
   PHASE FTPSTASK,*,NOAUTO
    INCLUDE FTPSTASK
    INCLUDE OCSLIH
    INCLUDE OCSIRH
    INCLUDE OCSPLH
    INCLUDE OCSDAH
    INCLUDE OCSNPL
    INCLUDE OCSGSV
    INCLUDE OCSGEH
    INCLUDE OCSDRH
    INCLUDE OCSVPL
    INCLUDE OCSSMH
/*
   ENTRY FTPSTASK
/*
// EXEC PGM=LNKEDT,PARM=¢MSHP¢
/*
/&
* $$ EOJ
```

Figure 45 (Part 2 of 2). Build the Executable Phases for OC/FTP Server

6.2.4.2 Run the Site Protection Modification Job

Figure 46 lists the job we used for the OCS license protection. The source of this job is obtained from PRODBLDS.PROC.

```
* $$ JOB JNM=SFASM2,DISP=D,PRI=9,                                            C
* $$ NTFY=YES,                                                               C
* $$ LDEST=*,                                                                C
* $$ CLASS=0
*  -----------------------------------------------------  *
*  APPLYING LICENSE SITE PROTECTION MODIFICATION.          *
*  -----------------------------------------------------  *
// EXEC PGM=MSHP
   REMOVE  5758-PC-115-224 APAR=SP11501
   CORRECT 5758-PC-115-224:SP11501 IRREVOKABLE
     AFFECTS PHASE=FTPSMAIN
       ALTER 0088 /16/AA:DA24D678DE745738C1DC60E04C517412
       ALTER 0000 /16/AA:A35EF6929F756CB8496C7B334507AA27
       ALTER 0044 /16/AA:2CC7C589EC4CC491D190EAF899CFCDC4
     RESOLVES ¢LICENSE SITE PROTECTION¢
/*
/&
* $$ EOJ
```

Figure 46. Site Protection Job for OC/FTP Server

6.2.4.3 Create the FTP Server Startup Job

The source of this job is obtained from PRODBLDS.PROC. We added the necessary JCL to start the program in a dynamic partition. Figure 47 lists the job we used.

```
* $$ JOB JNM=FTPS,DISP=L,PRI=3,
* $$ NTFY=YES,
* $$ LDEST=(,RSCS),
* $$ CLASS=Z
* $$ LST CLASS=A
// JOB FTPS
// OPTION LOG,NOSYSDUMP,JCANCEL
// LIBDEF *,SEARCH=(TCPOCS.FTPS,PRD2.CONFIG),TEMP
// EXEC FTPSMAIN,SIZE=(FTPSMAIN,256K),PARM=¢CONFIG¢
/*
/&
* $$ EOJ
```

Figure 47. OC/FTP Server Startup Job

This job is submitted to the POWER RDR queue and is ready to start. The operation and testing of OC/FTP Server are discussed in Chapter 18, "OC/FTP Server Operation and Examples" on page 219.

6.2.5 Customizing the Translation Tables

This step is optional. The OC/FTP Server comes with a default data translation table ′FTPSXLTE′. If, during the FTP session, no user specific table is specified via the SITE command (′QUOTE SITE XLATE *newxlte*′), the default translation table will be used.

The customization of the OC/FTP Server translation table involves three steps:

1. Copy the sample ′FTPSXLTE′ member from the OC/FTP Server production library. Simply issue the following command at your VSE/ICCF command mode screen:

   LIBRP TCPOCS.FTPS FTPSXLTE.SAMPLE FTPSXLTE

2. Change the appropriate hexadecimal codes in the EBCDIC to ASCII table and vice versa.

   As an example, assume you transfer data from a PC system to VSE and you want to preserve the left/right bracket symbols (x′5B′ and x′5D′). You need to change the ASCII to EBCDIC table as follows:

   The original value at x′5B′ was x′4A′, this means that the ASCII x′5B′ left bracket will be translated to an EBCDIC x′4A′ which is a cent sign. Since the EBCDIC value for the left bracket is x′80′, we should change x′4A′ to x′80′ at this ASCII x′5B′ location.

   Do the same for the ASCII x′5D′ right bracket, and change the value from an EBCDIC x′4F′ at that location to x′90′.

Now if you want to have the same translation from EBCDIC to ASCII, you will have to modify the EBCDIC to ASCII table:

1. Locate the EBCDIC x′80′ left bracket, and change the value from an ASCII x′00′ at that location to x′5B′.

2. Locate the EBCDIC x′90′ left bracket, and change the value from an ASCII x′00′ at that location to x′5D′.

3. Assemble and link-edit the job to produce a user specific translation phase. Figure 48 on page 78 lists the job we used to assemble and link-edit the translation table to become a new phase ′NEWXLTE′. The table has been customized for the preservation of the left/right brackets.

   The translation table is not part of the startup phase ′FTPSMAIN′, therefore it is not required to rebuild this phase.

```
* $$ JOB JNM=NEWXLTE,CLASS=A
// JOB NEWXLTE - USER SPECIFIC TRANSLATION TABLE
// LIBDEF SOURCE,SEARCH=TCPOCS.FTPS
// LIBDEF PHASE,CATALOG=TCPOCS.FTPS
// OPTION CATAL
   PHASE NEWXLTE,*
// EXEC PGM=ASSEMBLY
         TITLE ¢TRANSLATION TABLES¢
NEWXLTE  CSECT
*----------------------------------------------------------------------*
*                  TABLE FOR EBCDIC TO ASCII
*----------------------------------------------------------------------*
*                  0 1 2 3 4 5 6 7 8 9 A B C D E F    EBCDIC
         DC    X¢000102030009007F0000000B0C0D0E0F¢   00-0F
         DC    X¢10111200000A0800181900001C1D1E1F¢   10-1F
         DC    X¢00000000000171B0000000000050607¢   20-2F
         DC    X¢0000160000000004000000001415001A¢   30-3F
         DC    X¢2000000000000000000005B2E3C282B5D¢   40-4F
         DC    X¢26000000000000000000021242A293B5E¢   50-5F
         DC    X¢2D2F0000000000000007C2C255F3E3F¢   60-6F
         DC    X¢000000000000000000603A2340273D22¢   70-7F
         DC    X¢5B616263646566676869000000000000¢   80-8F
         DC    X¢5D6A6B6C6D6E6F707172000000000000¢   90-9F
         DC    X¢007E737475767778797A000000000000¢   A0-AF
         DC    X¢00000000000000000000000000000000¢   B0-BF
         DC    X¢7B414243444546474849000000000000¢   C0-CF
         DC    X¢7D4A4B4C4D4E4F505152000000000000¢   D0-DF
         DC    X¢5C00535455565758595A000000000000¢   E0-EF
         DC    X¢303132333435363738390000000000FF¢   F0-FF
*----------------------------------------------------------------------*
*                  TABLE FOR ASCII TO EBCDIC
*----------------------------------------------------------------------*
*                  0 1 2 3 4 5 6 7 8 9 A B C D E F    ASCII
         DC    X¢00010203372D2E2F1605150B0C0D0E0F¢   00-0F
         DC    X¢101112003C3D322618193F271C1D1E1F¢   10-1F
         DC    X¢405A7F7B5B6C507D4D5D5C4E6B604B61¢   20-2F
         DC    X¢F0F1F2F3F4F5F6F7F8F97A5E4C7E6E6F¢   30-3F
         DC    X¢7CC1C2C3C4C5C6C7C8C9D1D2D3D4D5D6¢   40-4F
         DC    X¢D7D8D9E2E3E4E5E6E7E8E980E0905F6D¢   50-5F
         DC    X¢79818283848586878889919293949596¢   60-6F
         DC    X¢979899A2A3A4A5A6A7A8A9C06AD0A107¢   70-7F
         DC    X¢00000000000000000000000000000000¢   80-8F
         DC    X¢00000000000000000000000000000000¢   90-9F
         DC    X¢00000000000000000000000000000000¢   A0-AF
         DC    X¢00000000000000000000000000000000¢   B0-BF
         DC    X¢00000000000000000000000000000000¢   C0-CF
         DC    X¢00000000000000000000000000000000¢   D0-DF
         DC    X¢00000000000000000000000000000000¢   E0-EF
         DC    X¢F0F1F2F3F4F5F6F7F8F9FAFBFCFDFEFF¢   F0-FF
         END
/*
// EXEC PGM=LNKEDT
/*
/&
* $$ EOJ
```

Figure 48. Assemble and Link-edit NEWXLTE Translation Table

6.2.6 User Exits

There are two user exits supported by OC/FTP Server:

1. Access Control Exit.

   This is the exit program ′FTPSXIT1′ which provides some security and management control of the OC/FTP Server. Please refer to 6.2.3.1, “CONFIG Member” on page 67 for a description of the FTPSXIT1 parameter.

2. I/O Exit.

   This exit program can provide other access methods for usage by OC/FTP Server. The exit is activated through the SITE command:

   quote site user ioexit *exitpgm*

   This defines that the exit routine *exitpgm* implements the access method to be used by the OC/FTP Server.

Chapter 7. OC/FTP Client Installation and Customization

This chapter covers the installation and customization of OC/FTP Client. It consists of two steps:

1. Installation - Describes the preparation and procedures to install OC/FTP Client in VSE/ESA

2. Customization - Provides a detailed explanation of the product customization

7.1 OC/FTP Client Installation

There are two major steps involved to install the product:

1. Prepare the installation environment for the product

2. Install the product from tape to the VSE/ESA sublibrary

7.1.1 Installation Preparation

To prepare for installation of the OC/FTP Client the following definitions are required:

- OCS II Gateway to VSE/ESA

  Please refer to Chapter 5, "Defining the OCS II Gateway to the Host" on page 41 for the setup instructions.

- The VSE/ESA library and sublibraries for the product installation

  Library TCPOCS definition is discussed and defined in 6.1, "OC/FTP Server Installation" on page 61.

  Sublibrary TCPOCS.FTPC is defined using the LIBR DEF SUB command in ICCF. This sublibrary is used to store the OC/FTP Client product, and the FTP client customization.

- VSE/ESA user sublibraries for the VSE/ESA FTP clients

  This is an optional step. The numbers and names of the sublibraries to be used depend on the installation's requirements. You may define as many user sublibraries as needed. We used the sublibraries defined in 6.1, "OC/FTP Server Installation" on page 61. Default library and sublibrary assignment for the FTP clients is discussed in 7.2.4.1, "DYNALL Member" on page 92.

- VSAM catalog and space definitions for the VSE/ESA FTP clients

  We used the same VSAM catalog TCPCAT defined in 6.1, "OC/FTP Server Installation" on page 61 for the VSE/ESA FTP clients.

  This is an optional step. You may define a new or use an existing catalog for the VSE/ESA FTP clients, depending on the installation's requirements. The default VSAM catalog assignment for the FTP client is discussed in 7.2.4.1, "DYNALL Member" on page 92.

7.1.2 Installation Procedures

Installation of the product from tape to the product sublibrary involves the following steps:

- Use the VSE/ESA II Installation dialog to generate the installation jobstreams
- Scan the product tape by submitting the job in Figure 35 on page 63 to VSE/ESA
- Submit the generated job to install OC/FTP Client into the TCPOCS.FTPC sublibrary

There is no installation verification job provided by the software. You may use the sample job in Figure 36 on page 64, to verify that the product is recorded by MSHP. Testing of the product can only be performed after the product customization. Please refer to Chapter 19, "OC/FTP Client Operation and Examples" on page 239 for the testing and operation procedures.

7.2 OC/FTP Client Customization

The customization is based on the samples in PRODBLDC.PROC provided by OC/FTP Client. After the product is installed in the product sublibrary, PRODBLDC.PROC is punched into an ICCF library, using the following command from your VSE/ICCF command mode screen:

LIBRP TCPOCS.FTPC PRODBLDC.PROD PRODBLDC

We then edited it into individual jobs to tailor the environment for OC/FTP Client.

The OC/FTP Client customization consists of the following steps:

- VSAM customization
- VTAM customization
- CICS customization
- FTP client customization
- Startup customization

7.2.1 VSAM Customization

When an FTP client requests a file from a TCP/IP node (via FTP GET) as VSAM files (ESDS, KSDS or SAM), OC/FTP Client requests VSAM to implicit-define the file to the assigned catalog. As discussed in 6.2.1, "VSAM Customization" on page 64, VSAM implicit-define requires a model in the catalog to define the data set. Please refer to the job in Figure 37 on page 65 for details.

The VSAM data set definition parameters may be overridden by the DYNALL member and DEFAULT command. Please refer to 7.2.4.1, "DYNALL Member" on page 92, and Chapter 19, "OC/FTP Client Operation and Examples" on page 239 for details.

7.2.2 VTAM Customization

The following customization tasks are required in VTAM:

- Define the logmode table and entries for the applications and LUs
- Specify the APPL definitions for the OC/FTP Client
- Add the LU definitions for the OCS II Gateway LUs

7.2.2.1 Logmode Table OCSBIND

The sample job listed in A.6, "OCSBIND MODETAB for OCS Products" on page 351 is used to create the OCSBIND logmode table. OC/FTP Client uses the following entries:

- OCSLU62P for the OC/FTP Client program that communicates and interfaces with CICS via LU 6.2 sessions
- OCSLU0 for the programs used by OC/FTP Client
- OCSLU0 for the OCS II Gateway LUs used by OC/FTP Client

Figure 49 lists the two logmode entries used by OC/FTP Client.

```
OCSLU62P MODEENT  LOGMODE=OCSLU62P,                                  C
               TYPE=X¢00¢,                                           C
               FMPROF=X¢13¢,                                         C
               TSPROF=X¢07¢,                                         C
               PRIPROT=X¢B0¢,                                        C
               SECPROT=X¢B0¢,                                        C
               COMPROT=X¢50A1¢,                                      C
               RUSIZES=X¢8585¢,                                      C
               SSNDPAC=X¢07¢,                                        C
               SRCVPAC=X¢07¢,                                        C
               PSNDPAC=X¢07¢,                                        C
               PSERVIC=X¢060200000000000000002300¢
OCSLU0   MODEENT  LOGMODE=OCSLU0,                                    C
               FMPROF=X¢06¢,                                         C
               TSPROF=X¢03¢,                                         C
               PRIPROT=X¢00¢,                                        C
               SECPROT=X¢00¢,                                        C
               COMPROT=X¢0000¢,                                      C
               RUSIZES=X¢F8F8¢,                                      C
               PSERVIC=X¢000000000000000000000000¢
```

Figure 49. MODETAB Entries for OC/FTP Client

7.2.2.2 VTAM APPL Definitions

Before we discuss the VTAM APPL definitions, it is important to understand how OC/FTP Client establishes sessions with the LUs in OCS II Gateway.

Figure 50 on page 85 provides a summary of the OC/FTP Client sessions and the relationships between APPL and LU definitions.

- VTAM, CICS and OCS II Gateway are initialized, and OCS II Gateway is connected to VTAM
- When OC/FTP Client is started in a dynamic partition, the OC/FTP Client program OCSFTP62 automatically establishes the parallel sessions with ISC modules in the CICS partition. The number of concurrent sessions is determined in the OCSFTP62 APPL statement. OC/FTP Client is now ready to accept FTP transactions from CICS terminals
- When the FTP transaction is entered from a CICS terminal, the OC/FTP Client online interface is then initialized
- When OC/FTP Client is invoked from a batch job, OC/FTP Client runs in the batch dynamic partition and initializes the offline interface
- After the online or offline interface is started, OC/FTP Client is ready to accept the OPEN command from the VSE/ESA FTP clients
- For each session request, OC/FTP Client dispatches a program with the name OCSFTP**nn**, where **nn** ranges from 01 to the maximum sessions allowed. OCSFTPnn creates an LU-LU session with the first available LU in OCS II Gateway
- OCSFTPnn is a VTAM application and must be defined in the VTAM APPL major node. The number of OCSFTPnn APPL entries depends on the maximum number of concurrent FTP sessions required
- After the OCSFTPnn application establishes an LU-LU session with the OCS II Gateway LU, the system is ready for FTP requests such as PUT and GET from the VSE/ESA FTP clients

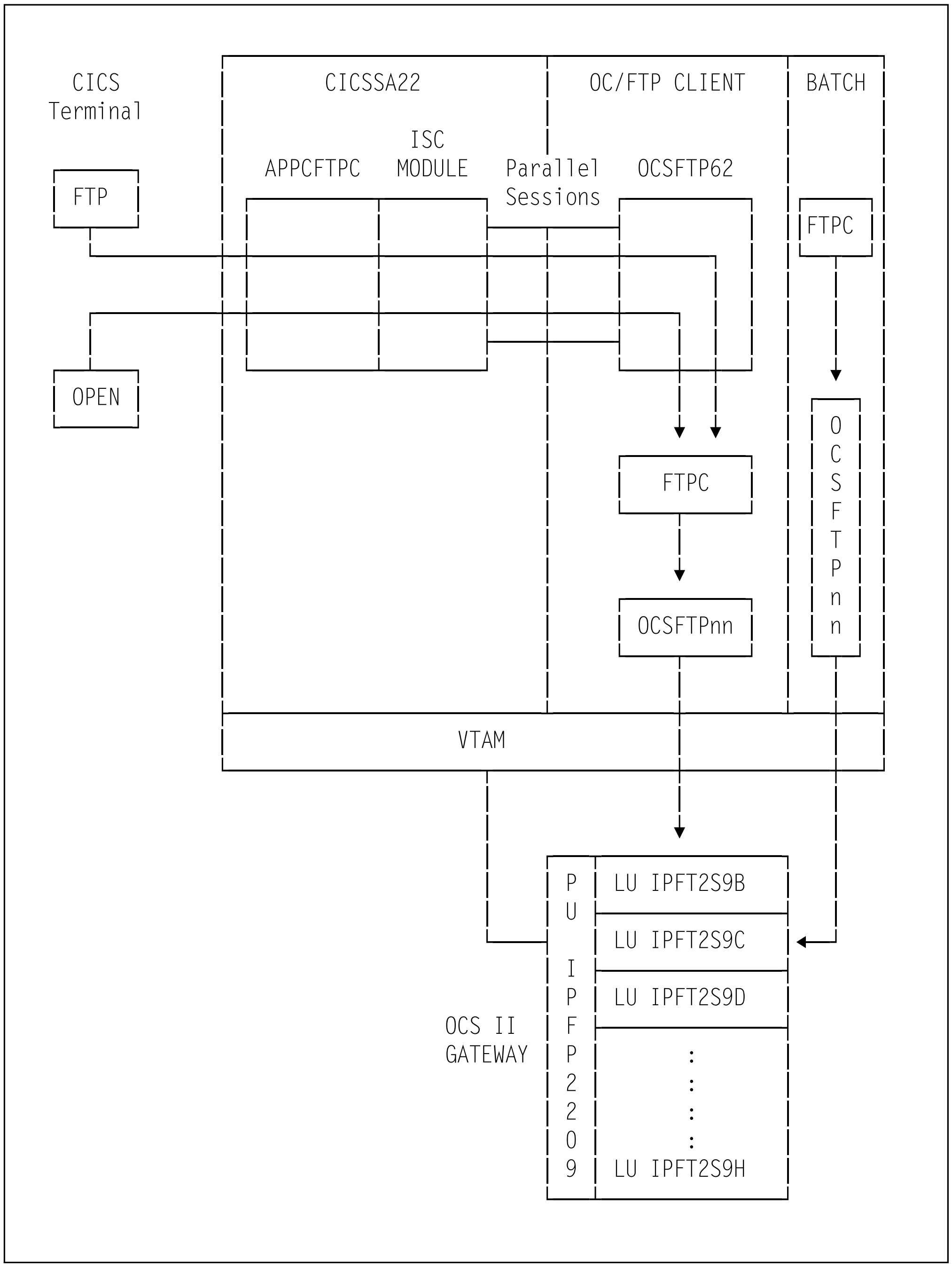


Figure 50. OC/FTP Client Applications and LU Relationships

The VTAM APPL entries for OC/FTP Client are listed in Figure 51 on page 86. The completed VTAM APPL major node definition is listed in A.4, "VTAM APPL Major Node for OCS Products" on page 344.

```
 CICSSA22 APPL  AUTH=(PASS,ACQ,VPACE),PARSESS=YES,ACBNAME=CICSSA22,    C
                EAS=4000,MODETAB=CICSIPMT,APPC=NO,                      C
                SONSCIP=YES,VPACING=5
 OCSFTP62 APPL  MODETAB=OCSBIND,DLOGMOD=OCSLU62P,AUTH=(ACQ),APPC=YES,  C
                PARSESS=YES,DSESLIM=10,DMINWNL=0,DMINWNR=10
 OCSFTP01 APPL  MODETAB=OCSBIND,DLOGMOD=OCSLU0,AUTH=(NVPACE,ACQ),EAS=2
 OCSFTP02 APPL  MODETAB=OCSBIND,DLOGMOD=OCSLU0,AUTH=(NVPACE,ACQ),EAS=2
 OCSFTP03 APPL  MODETAB=OCSBIND,DLOGMOD=OCSLU0,AUTH=(NVPACE,ACQ),EAS=2
 OCSFTP04 APPL  MODETAB=OCSBIND,DLOGMOD=OCSLU0,AUTH=(NVPACE,ACQ),EAS=2
 OCSFTP05 APPL  MODETAB=OCSBIND,DLOGMOD=OCSLU0,AUTH=(NVPACE,ACQ),EAS=2
 OCSFTP06 APPL  MODETAB=OCSBIND,DLOGMOD=OCSLU0,AUTH=(NVPACE,ACQ),EAS=2
 OCSFTP07 APPL  MODETAB=OCSBIND,DLOGMOD=OCSLU0,AUTH=(NVPACE,ACQ),EAS=2
 OCSFTP08 APPL  MODETAB=OCSBIND,DLOGMOD=OCSLU0,AUTH=(NVPACE,ACQ),EAS=2
 OCSFTP09 APPL  MODETAB=OCSBIND,DLOGMOD=OCSLU0,AUTH=(NVPACE,ACQ),EAS=2
 OCSFTP10 APPL  MODETAB=OCSBIND,DLOGMOD=OCSLU0,AUTH=(NVPACE,ACQ),EAS=2
```

Figure 51. VTAM APPL Statements for OC/FTP Client

The key definitions and parameters are:

- ***PARSESS=YES in the CICS APPL statement.*** This parameter allows CICS to have parallel sessions with the OC/FTP Client application OCSFTP62. Details of the CICS parallel session definitions are discussed in 7.2.3, “CICS Customization” on page 88.
- ***APPL statement for OCSFTP62.*** The definition is based on the sample from PRODBLDC.PROC. This program is responsible to set up multiple sessions and to interface with CICS. The key parameters are:
  - MODETAB and DLOGMOD. Identify the appropriate logmode table entry.
  - ACQ in AUTH parameter. Allows OCSFPT62 to acquire LUs.
  - APPC. Allows OCSFTP62 to participate in LU 6.2 sessions with CICS.
  - PARSESS. Allows OCSFTP62 to have parallel sessions with CICS.
  - DSESLIM. Defines the maximum number of sessions between this application and CICS. This parameter is one of the factors governing the maximum number of online FTP clients
  - DMINWNL and DMINWNR. Determine that CICS is the contention winner for the parallel sessions.
- ***APPL statements for OCSFTP*nn.** Another factor governing the maximum number of online FTP clients, is the number of OCSFTPnn APPL statements. **nn** consists of values ranging from 01 to the maximum number of online FTP clients required.

  You should define as many OCSFTPnn APPL statements as needed. The number of OCSFTPnn APPL statements should match the values specified in the DSESLIM parameter of the OCSFTP62 APPL statement. The names of the APPL statements must also match the entries in 7.2.4.2, “FAPPL Member” on page 94.

  The key parameters are:
  - MODETAB and DLOGMOD. Identify the appropriate logmode table entry.
  - NVPACE in AUTH parameter. Turns off pacing for FTP.
  - ACQ in AUTH parameter. Allows this application to acquire LU in OCS II Gateway.

– EAS=2. Allows two concurrent sessions for this application.

7.2.2.3 VTAM LU Definition

The completed OCS II Gateway LU definitions are listed in Figure 24 on page 47, and A.3, "VSE TCPSW.B Switched Major Node" on page 342.

Figure 52 lists the LU definitions used by OC/FTP Client. We defined seven LUs for the online FTP clients. The number of LUs depends on the installation requirements.

```
IPFT2S9B LU    LOCADDR=2,DLOGMOD=OCSLU0,                          C
               ISTATUS=ACTIVE,MODETAB=OCSBIND,                    C
               MDLTAB=VTMMDL,MDLENT=VSELU2A,                      C
               SSCPFM=USSSCS,USSTAB=VTMUSSTR
IPFT2S9C LU    LOCADDR=3,DLOGMOD=OCSLU0,                          C
               ISTATUS=ACTIVE,MODETAB=OCSBIND,                    C
               MDLTAB=VTMMDL,MDLENT=VSELU2A,                      C
               SSCPFM=USSSCS,USSTAB=VTMUSSTR
IPFT2S9D LU    LOCADDR=4,DLOGMOD=OCSLU0,                          C
               ISTATUS=ACTIVE,MODETAB=OCSBIND,                    C
               MDLTAB=VTMMDL,MDLENT=VSELU2A,                      C
               SSCPFM=USSSCS,USSTAB=VTMUSSTR
IPFT2S9E LU    LOCADDR=5,DLOGMOD=OCSLU0,                          C
               ISTATUS=ACTIVE,MODETAB=OCSBIND,                    C
               MDLTAB=VTMMDL,MDLENT=VSELU2A,                      C
               SSCPFM=USSSCS,USSTAB=VTMUSSTR
IPFT2S9F LU    LOCADDR=6,DLOGMOD=OCSLU0,                          C
               ISTATUS=ACTIVE,MODETAB=OCSBIND,                    C
               MDLTAB=VTMMDL,MDLENT=VSELU2A,                      C
               SSCPFM=USSSCS,USSTAB=VTMUSSTR
IPFT2S9G LU    LOCADDR=7,DLOGMOD=OCSLU0,                          C
               ISTATUS=ACTIVE,MODETAB=OCSBIND,                    C
               MDLTAB=VTMMDL,MDLENT=VSELU2A,                      C
               SSCPFM=USSSCS,USSTAB=VTMUSSTR
IPFT2S9H LU    LOCADDR=8,DLOGMOD=OCSLU0,                          C
               ISTATUS=ACTIVE,MODETAB=OCSBIND,                    C
               MDLTAB=VTMMDL,MDLENT=VSELU2A,                      C
               SSCPFM=USSSCS,USSTAB=VTMUSSTR
```

Figure 52. Logical Unit Definitions for OC/FTP Client

The key parameters are:

- ***The LU name.*** Must match the:
  - LU name in the OCS II Gateway customization
  - LU name in the Network LU List Member(s). Please refer to 7.2.4.4, "NETWORKS LU List Member for IPFNET NETWORKS Member" on page 97, 7.2.4.5, "NETWORKS LU List Member for SUNNET NETWORKS Member" on page 97 and 7.2.4.4, "NETWORKS LU List Member for IPFNET NETWORKS Member" on page 97 for the explanation of Network LU List members.
- ***LOCADDR.*** Must match the LU number in the OCS II Gateway customization.
- ***MODETAB and DLOGMOD.*** Identify the appropriate logmode table entry.

7.2.3 CICS Customization

This section covers the definitions in CICS to enable VSE/ESA users to use OC/FTP Client online. Our definition is based on the samples in PRODBLDC.PROC provided by OC/FTP Client. We used the CEDA transaction to define the following resources to CICS:

- Connection VSE
- Session VSESESS
- Transaction FTP
- Program APPCFTPC

We filed these resources under the group VSEGROUP. After the resources are defined, we used:

- CEDA INSTALL GROUP(VSEGROUP) to activate VSEGROUP
- CEDA ADD GROUP(VSEGROUP) to make VSEGROUP permanent

Screen images of the CEDA sessions are reproduced in this section, to help you with the definitions in your installation.

7.2.3.1 Connection VSE

A connection is used to define a VTAM application communicating with CICS through ISC or MRO.

As illustrated in Figure 50 on page 85, OC/FTP Client application OCSFTP62 communicates with CICS via the ISC links. OCSFTP62 is defined to CICS using CEDA DEF CONN with the connection name VSE. Listed below are the specifications of the VSE connection.

```
CEDA  DEFINE
 Connection     : VSE
 Group          : VSEGROUP
CONNECTION IDENTIFIERS
 Netname        : OCSFTP62
 INDsys         :
REMOTE ATTRIBUTES
 REMOTESystem   :
 REMOTEName     :
CONNECTION PROPERTIES
 ACcessmethod   : Vtam              Vtam | IRc | INdirect
 Protocol       : Appc              Appc | Lu61
 SInglesess     : No                No | Yes
 Datastream     : User              User | 3270 | SCs | STrfield | Lms
 RECordformat   : U                 U | Vb
OPERATIONAL PROPERTIES
 AUtoconnect    : All               No | Yes | All
 INService      : Yes               Yes | No
SECURITY
 SEcurityname   :
 ATtachsec      : Local             Local | Identify | Verify
 Bindpassword   :                   PASSWORD NOT SPECIFIED
```

The key parameters are:

- ***CONNECTION.*** The name of the connection. It must match the CONNECTION name in the session definition.
- ***GROUP.*** The name of the group to which this definition belongs.
- ***NETNAME.*** The VTAM APPLID of the OC/FTP Client application OCSFTP62. Please refer to Figure 51 on page 86 for the OCSFTP62 APPL statement.
- ***ACCESSMETHOD.*** VTAM as the access method.
- ***PROTOCOL.*** APPC to be used for CICS ISC.

7.2.3.2 Session VSESESS

A session is used to define the logical links and the session characteristics between CICS and OCSFTP62. The session definitions are closely related to the APPL statement of OCSFTP62. Listed below are the specifications of the VSESESS session.

```
CEDA  DEFINE
 Sessions       : VSESESS
 Group          : VSEGROUP
SESSION IDENTIFIERS
 Connection     : VSE
 SESSName       :
 NETnameq       :
 MOdename       : OCSLU62P
SESSION PROPERTIES
 Protocol       : Appc                Appc | Lu61
 MAximum        : 00010 , 00010       0-32767
 RECEIVEPfx     :
 RECEIVECount   : No                  No | 1-999
 SENDPfx        :
 SENDCount      : No                  No | 1-999
 SENDSize       : 04096               1-30720
 RECEIVESize    : 04096               1-30720
OPERATOR DEFAULTS
 OPERId         :
 OPERPriority   : 000                 0-255
 OPERRsl        : 0
 OPERSecurity   : 1
 USERId         :
SESSION USAGES
 Transaction    :
 SESSPriority   : 000                 0-255
OPERATIONAL PROPERTIES
 Autoconnect    : All                 No | Yes | All
 INservice      :                     No | Yes
 Buildchain     : Yes                 Yes | No
 USERArealen    : 000                 0-255
 IOarealen      : 00000 , 00000       0-32767
 RELreq         : No                  No | Yes
 Discreq        : No                  No | Yes
 NEPclass       : 000                 0-255
RECOVERY
  RECOvoption    : Sysdefault          Sysdefault | None
```

The key parameters are:

- ***SESSION.*** The name of the session.
- ***GROUP.*** The name of the group to which this definition belongs.
- ***CONNECTION.*** The name of the connection associated with this session. It must match the connection name defined in the CONNECTION definition.
- ***MODENAME.*** The logmode entry for OCSFTP62. It must match the DLOGMOD value of the OCSFTP62 APPL statement.
- ***PROTOCOL.*** APPC is required for ISC.
- ***MAXIMUM.*** This parameter has two values. The first value defines the maximum number of sessions supported. It should match the DSESLIM parameter of the OCSFTP62 APPL statement. The second value defines the maximum number of sessions as contention winner. It should match the DMINWNR parameter of the OCSFTP62 APPL statement.
- ***SENDSIZE.*** The RU size for send.
- ***RECEIVESIZE.*** The RU size for receive.
- ***AUTOCONNECT ALL.*** Automatic session connection between CICS and session partner OCSFTP62 during CICS initialization.

7.2.3.3 Define the ″FTP″ Transaction

CEDA DEF TRAN(FTP) is used to define the OC/FTP Client transaction FTP to CICS. Listed below are the specifications of the FTP transaction.

```
CEDA  DEFINE
 TRansaction    : FTP
 Group          : VSEGROUP
 PROGram        : APPCFTPC
 TWasize        : 00256              0-32767
 PROFile        : DFHCICST
 PArtitionset   :
 STatus         : Enabled            Enabled | Disabled
 PRIMedsize     : 00000              0-65520
REMOTE ATTRIBUTES
 DYnamic        : No                 No | Yes
 REMOTESystem   :
 REMOTEName     :
 TRProf         :
 Localq         :                    No | Yes
SCHEDULING
 PRIOrity       : 001                0-255
 TClass         : No                 No | 1-10
ALIASES
 Alias          :
 TAskreq        :
 Xtranid        :
RECOVERY
 DTimout        : No                 No | 1-7000
 Indoubt        : Backout            Backout | Commit | Wait
 REStart        : No                 No | Yes
 SPurge         : No                 No | Yes
 TPurge         : No                 No | Yes
 DUmp           : Yes                Yes | No
 TRACe          : Yes                Yes | No
SECURITY
 Extsec         : No                 No | Yes
 TRANsec        : 01                 1-64
 RSL            : Public             0-24 | Public
 RSLC           : No                 No | Yes | External
```

The key parameters are:

- ***TRANSACTION.*** The name of the OC/FTP Client transaction.
- ***GROUP.*** The name of the group to which this definition belongs.
- ***PROGRAM.*** The first CICS program that this transaction invokes.
- ***TWASIZE.*** The transaction work area. 256 bytes is recommended by OCS.

7.2.3.4 Program APPCFTPC

CEDA DEF PRO(APPCFTPC) is used to define the OC/FTP Client program APPCFTPC to CICS. Listed below are the specifications of the APPCFTPC program.

```
CEDA  DEFINE
 PROGram        : APPCFTPC
 Group          : VSEGROUP
 Language       : Assembler         CObol | Assembler | C | Pli | Rpg
 RELoad         : No                No | Yes
 RESident       : No                No | Yes
 RSl            : 00                0-24 | Public
 Status         : Enabled           Enabled | Disabled
REMOTE ATTRIBUTES
 REMOTESystem   :
 REMOTEName     :
 Transid        :
 Executionset   : Fullapi           Fullapi | Dplsubset
```

The key parameters are:

- ***PROGRAM.*** The name of the program associated with transaction FTP.
- ***GROUP.*** The name of the group to which this definition belongs.
- ***LANGUAGE.*** The programming language of this program is ASSEMBLER.

7.2.4 FTP Client Customization

This section covers the definitions required by the FTP function of OC/FTP Client. The following members are discussed:

- ***DYNALL.*** Defines the default VSAM and LIBR environments for the VSE/ESA FTP clients.
- ***FAPPL.*** Defines the programs used by OC/FTP Client.
- ***NETWORKS.*** Defines IP network names and addresses.
- ***NETWORKS LU List "IPFNET".*** Allocates LUs to the networks defined in the NETWORKS member.
- ***HOSTS.*** Defines TCP/IP IP addresses and host names.
- ***CACHE.*** Defines the most commonly used IP addresses.
- ***PACING.*** Defines the FTP inbound and outbound pacing values.
- ***NETMASK.*** Defines the IP subnet mask.
- ***RESOLVER.*** Defines the IP domain name server.

All these members are filed in sublibrary TCPOCS.FTPC.

7.2.4.1 DYNALL Member

The DYNALL member is used to define the default VSAM and VSE/ESA library environments for the VSE/ESA FTP clients.

The default values provided by OC/FTP Client may not apply to your installation. You should review and tailor the sample specifications provided in PRODBLDC.PROC.

The specification in the DYNALL member can be overridden by the DEFAULTS command. Please refer to the description of the DEFAULT command in Chapter 19, "OC/FTP Client Operation and Examples" on page 239 for details.

Figure 53 on page 93 lists the default environment we tailored.

```
 * $$ JOB JNM=CFDYNALL,DISP=D,PRI=3,NTFY=YES,LDEST=(,RSCS),CLASS=0
 * $$ LST CLASS=A
 // JOB CFDYNALL
 // EXEC LIBR,PARM=¢MSHP¢
 ACCESS SUBLIB=TCPOCS.FTPC
    CATALOG DYNALL.FTP EOD=$$ REPLACE=Y
VSAMCAT     TCPCAT          DEFAULT VSAM CATALOG
LRECL       00132           DEFAULT LIBR/VSAM RECORD LENGTH
PALLOC      0000512         DEFAULT VSAM PRIMARY ALLOCATION
SALLOC      0000256         DEFAULT VSAM SECONDY ALLOCATION
CISIZE      04096           DEFAULT VSAM DATA CISIZE
CISIZEDATA  02048           ALTERNATE DATA CISIZE KEYWORD
BUFND       020             DEFAULT VSAM DATA BUFND=
BUFNDATA    010             ALTERNATE DATA BUFND= KEYWORD
BUFSP       0000000         DEFAULT VSAM CLUSTER BUFSP=
OPNDSP      NEW             DEFAULT VSAM OPEN DISPOSITION
CLSDSP      KEEP            DEFAULT VSAM CLOSE DISPOSITION
RETAIN      0007            DEFAULT VSAM CLUSTER RETENTION
MODE        LIBR            DEFAULT OPERATION MODE
DDSN        OCS.FTP.TESTING DEFAULT VSAM DATASET NAME
DLIB        TCPOCS          DEFAULT LIBRARIAN LIBRARY
DSLIB       USR1            DEFAULT LIBRARIAN SUBLIBRARY
DMBRN       FTPOBJ          DEFAULT LIBRARIAN MEMBER NAME
DMBRT       TCPIP           DEFAULT LIBRARIAN MEMBER TYPE
RECFM       RECORD          DEFAULT LIBR/VSAM RECORD FORMAT
BUFNINDEX   004             DEFAULT VSAM KSDS BUFNI=
BLKFACTOR   001             DEFAULT VSAM BLOCK FACTOR
KEYLOCATION 000             DEFAULT VSAM KSDS KEY LOCATION
KEYLENGTH   010             DEFAULT VSAM KSDS KEY LENGTH
CAFREESPACE 000             DEFAULT VSAM KSDS CA FREESPACE
CIFREESPACE 000             DEFAULT VSAM KSDS CI FREESPACE
CISIZEINDEX 000             DEFAULT VSAM KSDS CISIZE
MODELENTRY  NULL            DEFAULT VSAM MODEL ENTRY
SHARE       2               DEFAULT VSAM SHAREOPTION
UNITS       REC             DEFAULT VSAM ALLOCATIONS UNITS
$$
/*
/&
* $$ EOJ
```

Figure 53. DYNALL Member for OC/FTP Client

Listed below is a brief description of the parameters:

- ***VSAMCAT.*** The default VSAM master/user catalog DLBL name used for VSAM file transfer. This will be used when no CWD specified via VSECD command.
- ***LRECL.*** The maximum VSAM logical record length.
- ***PALLOC.*** The primary record allocation for the VSAM implicit-define. The value is changed from 256 to 512 records.
- ***SALLOC.*** The secondary record allocation for the VSAM implicit-define. The default is 256.
- ***CISIZE.*** The VSAM CI size. We specified a 4K CI size.
- ***BUFND.*** The number of CI buffers to be allocated. We specified 20 CI buffers.
- ***BUFSP.*** The buffer space to be allocated. We let it default to 0.
- ***OPNDSP and CLSDSP.*** The open and close dispositions of the data set.
- ***RETAIN.*** The number of retention days of the data set. The default value is 7.
- ***MODE.*** The access mode for the VSE/ESA FTP clients. The value can be VSAM or LIBR.
- ***DDSN.*** The default VSAM file node name to be searched in the working directory, if not specified in the MPUT command. A value of ′*′ marks all the files in the working directory.

  This parameter is especially important for the MPUT command. If the FTP clients want to put multiple VSAM files with the same file node name to the TCP/IP FTP server, the node name should be specified in the:

  - DDSN parameter of the DYNALL member **or**
  - DSN parameter of the DEFAULTS command

  For example, If you want to PUT the following files to the FTP server:

  - TEST.FILE1
  - TEST.FILE2
  - TEST.FILE3

  You should override the DSN with the node name ′TEST′ using the DEFAULTS command, then issue the ′MPUT *′ command to put all three files to the FTP server.
- ***DLIB and DSLIB.*** The names of the default library and sublibrary.
- ***DMBRN and DMBRT.*** The default member name and type.

7.2.4.2 FAPPL Member

The FAPPL member is used to define the VTAM ACB names used by OC/FTP Client. The names must match the OCSFTPnn APPL entries defined in the VTAM APPL major node.

Figure 54 on page 95 lists the job we used.

```
* $$ JOB JNM=CFAPPLS,DISP=D,PRI=3,NTFY=YES,LDEST=(,RSCS),CLASS=0
* $$ LST CLASS=A
// JOB CSAPPLS
// EXEC LIBR,PARM=¢MSHP¢
ACCESS SUBLIB=TCPOCS.FTPC
   CATALOG FAPPLS.FTP EOD=$$ REPLACE=Y
OCSFTP01
OCSFTP02
OCSFTP03
OCSFTP04
OCSFTP05
OCSFTP06
OCSFTP07
OCSFTP08
OCSFTP09
OCSFTP10
$$
/*
/&
* $$ EOJ
```

Figure 54. FAPPL Member for OC/FTP Client

7.2.4.3 NETWORKS Member

The NETWORKS and NETWORK LU List members are used to define to OC/FTP Client, the:

- IP network names and addresses
- OCS II Gateway LU assignments to specific networks

The 32-bit IP address consists of the network address and host address. It is usually represented in a four-part dotted decimal format, for example 192.61.100.81.

The length of the network and host address is determined by the IP address class used. There are four IP address classes: A, B, C and D. Please refer to *TCP/IP Tutorial and Technical Overview, GG24-3376* for the description of IP addresses and classes.

We used both IP address class C and class A in our TCP/IP network.

For IP address class C, we use a subnet mask 255.255.255.0. This means there are 8 bits left for host addresses and we don't have a subnetwork. As usual, bits 1-3 of the IP address are the IP address class identifier. The next 21 bits are the network address, followed by 8 bits for the host address. In dotted decimal format, the first three parts form the network address and the last part forms the host address.

For IP address class A, we also use a subnet mask 255.255.255.0. Since class A only uses the first 8 bits for the network address, the next 16 bits are used for subnetworking, followed by the last 8 bits for host addresses. For this case, we have to specify a subnet mask in the NETMASK member. Please refer to 7.2.4.10, "NETMASK Member" on page 101. In dotted decimal format, the first part identifies the network address, the second and third parts form a subnetwork address, and the last part forms the host address.

All these IP addresses and subnet masks will only be used by the applications to determine the route to nodes within the TCP/IP network. The IP address class only limits the number of hosts in the network and the subnet mask only divides the existing network into two or more smaller networks allowing for more hosts to be attached.

Figure 55 illustrates the definitions and relationships between the NETWORKS and NETWORK LU List members for our configuration.

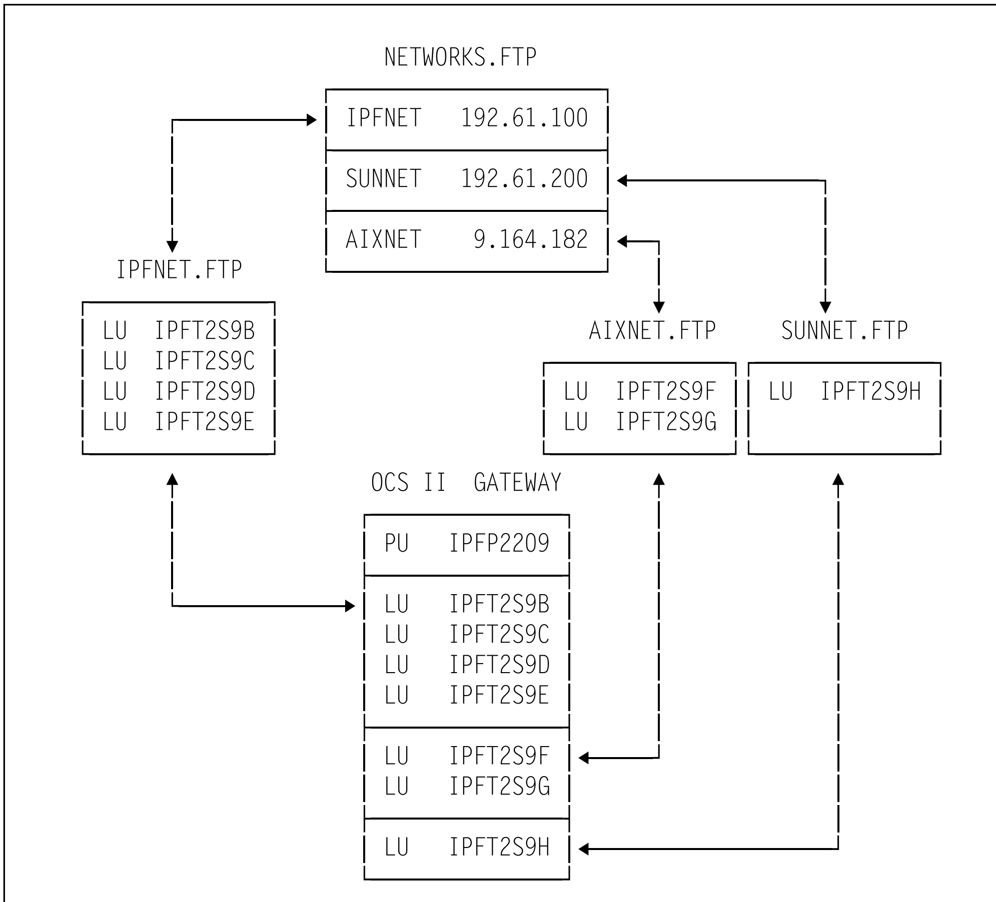


Figure 55. Definition and Relationships between NETWORKS and LU List Members

This definition is based on the network diagram in Figure 13 on page 34, and the IP address and name assignments in Table 3 on page 36. In the NETWORKS Member, we defined three networks with IPFNET as the default network:

- IPFNET with IP network address 192.61.100 (IBM Token-Ring LAN)
- SUNNET with IP network address 192.61.200 (Ethernet LAN)
- AIXNET with IP network address 9.164.182 (IBM Token-Ring LAN)

Figure 56 on page 97 shows the job we tailored.

```
* $$ JOB JNM=CFNETW,DISP=D,PRI=3,NTFY=YES,LDEST=(,RSCS),CLASS=0
* $$ LST CLASS=A
// JOB CFNETW
// EXEC LIBR,PARM=¢MSHP¢
ACCESS SUBLIB=TCPOCS.FTPC
   CATALOG NETWORKS.FTP EOD=$$ REPLACE=Y
IPFNET 192.61.100
SUNNET 192.61.200
AIXNET  9.164.182
IPFNET
$$
/*
/&
* $$ EOJ
```

Figure 56. NETWORKS Member for OC/FTP Client

7.2.4.4 NETWORKS LU List Member for IPFNET NETWORKS Member

We used the IPFNET.FTP member to assign four LUs to the IPFNET network. The LU list member name must correspond to the network name defined in the NETWORKS member. The LU names must correspond to the LU names defined for OCS II Gateway.

Figure 57 lists the job we tailored.

```
* $$ JOB JNM=CFLUVSE,DISP=D,PRI=3,NTFY=YES,LDEST=(,RSCS),CLASS=0
* $$ LST CLASS=A
// JOB CFLUVSE
// EXEC LIBR,PARM=¢MSHP¢
ACCESS SUBLIB=TCPOCS.FTPC
   CATALOG IPFNET.FTP EOD=$$ REPLACE=Y
GATEWAY BEGIN
LU IPFT2S9B
LU IPFT2S9C
LU IPFT2S9D
LU IPFT2S9E
GATEWAY END
$$
/*
/&
* $$ EOJ
```

Figure 57. NETWORK LU List Member for IPFNET NETWORKS Member

7.2.4.5 NETWORKS LU List Member for SUNNET NETWORKS Member

We used the SUNNET.FTP member to assign two LUs to the SUNNET network. The LU list member name must correspond to the network name defined in the NETWORKS member. The LU names must correspond to the LU names defined for OCS II Gateway.

Figure 58 on page 98 lists the job we tailored.

```
* $$ JOB JNM=CFLUVSE2,DISP=D,PRI=3,NTFY=YES,LDEST=(,RSCS),CLASS=0
* $$ LST CLASS=A
// JOB CFLUVSE2
// EXEC LIBR,PARM=¢MSHP¢
ACCESS SUBLIB=TCPOCS.FTPC
   CATALOG SUNNET.FTP EOD=$$ REPLACE=Y
GATEWAY BEGIN
LU IPFT2S9H
GATEWAY END
$$
/*
/&
* $$ EOJ
```

Figure 58. NETWORK LU List Member for SUNNET NETWORKS Member

7.2.4.6 NETWORKS LU List Member for AIXNET NETWORKS Member

We used the AIXNET.FTP member to assign two LUs to the AIXNET network which is a class A network and needs a subnet mask identification from the NETMASK member. The LU list member name must correspond to the network name defined in the NETWORKS member. The LU names must correspond to the LU names defined for OCS II Gateway.

Figure 59 lists the job we tailored.

```
* $$ JOB JNM=CFLUVSE3,DISP=D,PRI=3,NTFY=YES,LDEST=(,RSCS),CLASS=0
* $$ LST CLASS=A
// JOB CFLUVSE3
// EXEC LIBR,PARM=¢MSHP¢
ACCESS SUBLIB=TCPOCS.FTPC
   CATALOG AIXNET.FTP EOD=$$ REPLACE=Y
GATEWAY BEGIN
LU IPFT2S9F
LU IPFT2S9G
GATEWAY END
$$
/*
/&
* $$ EOJ
```

Figure 59. NETWORK LU List Member for AIXNET NETWORKS Member

7.2.4.7 HOSTS Member

This member is used to define the local and remote IP addresses and host names to OC/FTP Client. With the specifications in this member, the VSE/ESA FTP clients may use the host name for the OPEN command. The definitions in this member are based on the IP name and address assignments in Table 3 on page 36.

Figure 60 lists the job we tailored.

```
* $$ JOB JNM=CFHOSTS,DISP=D,PRI=3,NTFY=YES,LDEST=(,RSCS),CLASS=0
* $$ LST CLASS=A
// JOB CFHOSTS
// EXEC LIBR,PARM=¢MSHP¢
ACCESS SUBLIB=TCPOCS.FTPC
   CATALOG HOSTS.FTP EOD=$$ REPLACE=Y
192.61.100.84  TCPCL2
192.61.100.55  RS6TECTR
192.61.200.55  RS6
192.61.200.91  SUN2
9.164.182.132  AIX560
$$
/*
/&
* $$ EOJ
```

Figure 60. HOSTS Member for OC/FTP Client

Note

There are no host names AIX320, TCPCL1 and TCPCL3. These three hosts are using the domain name system. They go through the domain name server TCPCL1 to resolve unknown host names. Please refer to 7.2.4.11, "RESOLVER Member" on page 101 for more information on domain name server.

7.2.4.8 CACHE Member

This member is used to define the most commonly used IP addresses to the OC/FTP Client. They are usually the IP addresses defined in the HOSTS member. These IP addresses will be loaded into memory when the OC/FTP Client is started. This saves time searching for the IP addresses.

Figure 61 on page 100 lists the job we tailored.

```
* $$ JOB JNM=CFCACHE,DISP=D,PRI=3,NTFY=YES,LDEST=(,RSCS),CLASS=0
* $$ LST CLASS=A
// JOB CFCACHE
// EXEC LIBR,PARM=¢MSHP¢
ACCESS SUBLIB=TCPOCS.FTPC
   CATALOG CACHE.FTP EOD=$$ REPLACE=Y
192.61.100.84
$$
/*
/&
* $$ EOJ
```

Figure 61. CACHE Member for OC/FTP Client

7.2.4.9 PACING Member

This member is used to define the OCS application level pacing values between OC/FTP Client and the remote node. Please refer to the PACEIN/PACEOUT parameter in 6.2.3.1, "CONFIG Member" on page 67, for an explanation of OCS pacing.

You may include as many statements as required to define the OCS pacing values between OC/FTP Client and the remote nodes. The definition statement consists of three fields:

1. The first field is the inbound pacing

   It defines the number of RUs the remote FTP server may send to OC/FTP Client, before waiting for a pacing response from OC/FTP Client

2. The second field is the outbound pacing

   It defines the number of RUs OC/FTP Client may send to the remote FTP server, before waiting for a pacing response from the remote server

3. The third field defines the name of the remote FTP server

Figure 62 lists the job we tailored.

```
* $$ JOB JNM=CFPACING,DISP=D,PRI=3,NTFY=YES,LDEST=(,RSCS),CLASS=0
* $$ LST CLASS=A
// JOB CFPACING
// EXEC LIBR,PARM=¢MSHP¢
ACCESS SUBLIB=TCPOCS.FTPC
   CATALOG PACING.FTP EOD=$$ REPLACE=Y
10.10 TCPCL1
10.10 TCPCL2
10.10 TCPCL3
10.10 AIX320
10.10 AIX560
$$
/*
/&
* $$ EOJ
```

Figure 62. PACING Member for OC/FTP Client

7.2.4.10 NETMASK Member

Since we have a Token-Ring network 9.164.182 which has subnetworking, we have to define our subnet mask in this member.

This member is used only to specify the subnet masks for the implementation of IP subnets. Those networks which do not use subnetworking don′t have to be specified here. Please refer to *TCP/IP Tutorial and Technical Overview, GG24-3376* for the explanation of IP subnets. Figure 63 lists the job we tailored.

```
* $$ JOB JNM=NETMASK,CLASS=A,DISP=D
// JOB NETMASK - FOR SUBNET MASK
// EXEC LIBR,PARM=¢MSHP¢
  ACC S=TCPOCS.FTPC
   CATALOG NETMASK.FTP EOD=$$ REPLACE=Y
9.164.182  255.255.255.0
$$
/*
/&
* $$ EOJ
```

Figure 63. NETMASK Member for OC/FTP Client

7.2.4.11 RESOLVER Member

In our test environment, we have set up the OS/2 TCP/IP (TCPCL1) to be a domain name server for the following hosts:

- OS/2 itself named TCPCL1
- PC/DOS Windows named TCPCL3
- RISC/6000-320 OCS II named AIX320

These hosts don′t have a hosts file in their /etc directory. They resolve host names through the Domain Name Server (DNS) **TCPCL1.ITSC.IBM.COM**.

In the first place, these hosts will look into /etc directory to see if a resolver file exists. If it does and the domain name server is running, the host name is resolved through DNS. If the DNS cannot resolve the host name, then the host′s file is searched, if it exists.

In contrast, our VSE/ESA host will search its host′s file (HOSTS member) first. If the name is not found and the RESOLVER member exists, it will resolve the host name through the DNS.

We have cataloged this member into the product sublibrary **TCPOCS.FTPC**.

// LIBDEF *,SEARCH=TCPOCS.FTPC

Please refer to 7.2.5.3, “Create the FTP Client Startup Online Job” on page 104 for the OC/FTP Client startup job. Refer also to 19.2.1, “Get a File from PC/DOS to VSE Library” on page 245 for an example of resolving host names via a Domain Name Server. Figure 64 on page 102 lists the job we tailored.

```
* $$ JOB JNM=RESOLVER,CLASS=A,DISP=D
// JOB RESOLVER - FOR DOMAIN NAME RESOLVER
// EXEC LIBR,PARM=¢MSHP¢
  ACC S=TCPOCS.FTPC
  CATALOG RESOLVER.FTP EOD=$$ REPLACE=Y
DFNAME ITSC.IBM.COM
SERVER 192.61.100.83   TCPCL1.ITSC.IBM.COM
$$
/*
/&
* $$ EOJ
```

Figure 64. RESOLVER Member for OC/FTP Client

7.2.5 Startup Customization

The following preparation tasks are required to bring up OC/FTP Client:

- Build the executable phases
- Run the site protection modification job
- Create the FTP Client startup job
- Create the sample FTP client batch job

7.2.5.1 Build the Executable Phase

Figure 65 on page 103 lists the job we used to assemble and link-edit the OC/FTP Client phase into the TCPOCS.FTPC sublibrary. The source of this job is provided in PRODBLDC.PROC.

Note

We put OCSSMH into the member shipped.

```
* $$ JOB JNM=CFASM,CLASS=0,DISP=D,NTFY=YES
// JOB CFASM
*  ---------------------------------------------------  *
*  BUILDING COMPONENT EXECUTION MODULE(S).              *
*  ---------------------------------------------------  *
// OPTION CATAL,NOXREF
// LIBDEF OBJ,SEARCH=TCPOCS.FTPC,TEMP
// LIBDEF SOURCE,SEARCH=TCPOCS.FTPC,TEMP
// LIBDEF PHASE,SEARCH=TCPOCS.FTPC,CATALOG=TCPOCS.FTPC,TEMP
   PHASE FTPC,*,NOAUTO
    INCLUDE CLNTVT
    INCLUDE OCSCVT
    INCLUDE MTKFTPM
    INCLUDE OCSDAH
    INCLUDE OCSIRH
    INCLUDE OCSPLH
    INCLUDE OCSDRH
    INCLUDE OCSLIH
    INCLUDE OCSGEH
    INCLUDE OCSDTF
    INCLUDE MTKINIT
    INCLUDE MTKLOGON
    INCLUDE OCSNPL
    INCLUDE NDFSIH
    INCLUDE OCSVPL
    INCLUDE MTKTIO
    INCLUDE MTKPORTX
    INCLUDE MTKFTPW
    INCLUDE MTKXLATE
    INCLUDE OCSDMH
    INCLUDE MTKTIME
    INCLUDE CLNTRNS
    INCLUDE CLNSCAN
    INCLUDE CLNSAM
    INCLUDE CLNNDF
    INCLUDE CLNIOS
    INCLUDE CLNCPU
    INCLUDE OCSSMH
    INCLUDE MTKRESV
    INCLUDE OCSMLH
    INCLUDE MTKEXIT1
    INCLUDE OCSGSV
    INCLUDE
/*
// EXEC PGM=ASSEMBLY
   PRINT ON,NOGEN,NODATA
   CDMOD RECFORM=FIXUNB,WORKA=YES,TYPEFLE=INPUT
   PRMOD RECFORM=FIXUNB,WORKA=YES
   END
/*
   ENTRY OCSIOO
// EXEC PGM=LNKEDT,PARM=¢MSHP¢
/*
/&
* $$ EOJ
```

Figure 65. Build the Executable Phases for OC/FTP Client

7.2.5.2 Run the Site Protection Modification Job

Figure 66 lists the job we used for the OCS license protection. The source of this job is provided in PRODBLDC.PROC.

```
* $$ JOB JNM=CFASM2,CLASS=0,DISP=D,NTFY=YES
// JOB CFASM2
*  ----------------------------------------------------  *
*  APPLYING LICENSE SITE PROTECTION MODIFICATION.        *
*  ----------------------------------------------------  *
// EXEC PGM=MSHP
   REMOVE  5758-PC-014-224 APAR=SP01401
   CORRECT 5758-PC-014-224:SP01401 IRREVOKABLE
     AFFECTS PHASE=FTPC
       ALTER 000C /16/AA:DCE41639117EB548EFD06F750D973161
       ALTER 001C /16/AA:A30366608A7413E8BE67747583F0B5B4
       ALTER 002C /16/AA:2CC7C589EC4CC491D190EAF899CFCDC4
     RESOLVES ¢LICENSE SITE PROTECTION¢
/*
/&
* $$ EOJ
```

Figure 66. Site Protection Job for OC/FTP Client

7.2.5.3 Create the FTP Client Startup Online Job

The source of this job is provided in PRODBLDC.PROC. We added the necessary JCL to start OC/FTP Client in a dynamic partition.

Figure 67 lists the job we used.

```
* $$ JOB JNM=FTPC,DISP=L,PRI=3,NTFY=YES,LDEST=*,CLASS=Y
// JOB FTPC
*
// LIBDEF *,SEARCH=TCPOCS.FTPC,TEMP
// EXEC PGM=OCSFTPC,SIZE=(OCSFTPC,256K)
/*
/&
* $$ EOJ
```

Figure 67. OC/FTP Client Startup Job

This job is submitted to the POWER RDR queue and is ready to start. The operation and testing of OC/FTP Client are discussed in Chapter 19, "OC/FTP Client Operation and Examples" on page 239.

7.2.5.4 Create the Sample FTP Client Batch Job

The sample FTP client job in PRODBLDC.PROC may be customized, to start and execute OC/FTP Client in a batch partition. Figure 68 lists the job we tailored to test out various FTP functions.

```
* $$ JOB JNM=CFBATCH,DISP=L,PRI=3,NTFY=YES,LDEST=*,CLASS=Y
// JOB CFBATCH
*
// LIBDEF *,SEARCH=TCPOCS.FTPC,TEMP
// EXEC PGM=FTPC,SIZE=(FTPC,256K)
OPEN TCPCL1
pram
jack
VSECD
DEFAULTS MODE LIBR LIB TCPOCS
VSECD .USR1
PWD
MKDIR C:\TCPIP\FTPC
RMDIR C:\TCPIP\FTPC
MKVSEDIR TCPOCS.FTPB
RMVSEDIR TCPOCS.FTPB
STATUS
LDIR
BYE
/*
/&
* $$ EOJ
```

Figure 68. Sample OC/FTP Client Batch Job

7.2.6 Customizing the OC/FTP Client Translation Tables

This step is optional. Unlike the OC/FTP Server, OC/FTP Client uses an object module for the data translation table. This object module ′MTKXLATE′ has to be linked with other modules to become the OC/FTP Client startup phase - ′FTPC′. Refer to Figure 65 on page 103 for the inclusion of ′MTKXLATE′ during the assembly of the startup phase.

The customization of OC/FTP Client translation table consists of the following steps:

1. Change the translation table source code. Please refer to 6.2.5, "Customizing the Translation Tables" on page 77 for changing the EBCDIC and ASCII tables.
2. Assemble the translation table source code and catalog it as an object module.
3. Assemble and link-edit the ′FTPC′ phase to include the new translation table object module.
4. Restart the online OC/FTP Client.

Figure 69 on page 106 shows the translation source code in hexadecimal. Figure 70 on page 107 shows the job we used to assemble the table and catalog it as an object module. Lastly, we resubmit the job shown in Figure 65 on page 103 to rebuild the ′FTPC′ phase.

```
* $$ JOB JNM=CATXLATE,CLASS=A
// JOB CATXLATE - CATALOG SOURCE TRANSLATION TABLE
// EXEC LIBR,PARM=¢MSHP¢
   ACCESS S=TCPOCS.FTPC
   CATALOG MTKXLATE.A  REPLACE=YES
MTKXLATE CSECT
*-------------------------------------------------------------------*
*                 TABLE FOR EBCDIC TO ASCII
*-------------------------------------------------------------------*
         ENTRY MTKASCII
MTKASCII DS    0F
*                 0 1 2 3 4 5 6 7 8 9 A B C D E F    EBCDIC
         DC    X¢000102030009007F0000000B0C0D0E0F¢   00-0F
         DC    X¢10111200000A0800181900001C1D1E1F¢   10-1F
         DC    X¢000000000000171B0000000000050607¢   20-2F
         DC    X¢0000160000000004000000001415001A¢   30-3F
         DC    X¢200000000000000000005B2E3C282B5D¢   40-4F
         DC    X¢2600000000000000000021242A293B5E¢   50-5F
         DC    X¢2D2F00000000000000007C2C255F3E3F¢   60-6F
         DC    X¢000000000000000000603A2340273D22¢   70-7F
         DC    X¢5B616263646566676869000000000000¢   80-8F
         DC    X¢5D6A6B6C6D6E6F707172000000000000¢   90-9F
         DC    X¢007E737475767778797A000000000000¢   A0-AF
         DC    X¢00000000000000000000000000000000¢   B0-BF
         DC    X¢7B414243444546474849000000000000¢   C0-CF
         DC    X¢7D4A4B4C4D4E4F505152000000000000¢   D0-DF
         DC    X¢5C00535455565758595A000000000000¢   E0-EF
         DC    X¢303132333435363738390000000000FF¢   F0-FF
*-------------------------------------------------------------------*
*                 TABLE FOR ASCII TO EBCDIC
*-------------------------------------------------------------------*
         ENTRY MTKEBCDI
MTKEBCDI DS    0F
*                 0 1 2 3 4 5 6 7 8 9 A B C D E F    ASCII
         DC    X¢00010203372D2E2F1605150B0C0D0E0F¢   00-0F
         DC    X¢101112003C3D322618193F271C1D1E1F¢   10-1F
         DC    X¢405A7F7B5B6C507D4D5D5C4E6B604B61¢   20-2F
         DC    X¢F0F1F2F3F4F5F6F7F8F97A5E4C7E6E6F¢   30-3F
         DC    X¢7CC1C2C3C4C5C6C7C8C9D1D2D3D4D5D6¢   40-4F
         DC    X¢D7D8D9E2E3E4E5E6E7E8E980E0905F6D¢   50-5F
         DC    X¢79818283848586878889919293949596¢   60-6F
         DC    X¢979899A2A3A4A5A6A7A8A9C06AD0A107¢   70-7F
         DC    X¢00000000000000000000000000000000¢   80-8F
         DC    X¢00000000000000000000000000000000¢   90-9F
         DC    X¢00000000000000000000000000000000¢   A0-AF
         DC    X¢00000000000000000000000000000000¢   B0-BF
         DC    X¢00000000000000000000000000000000¢   C0-CF
         DC    X¢00000000000000000000000000000000¢   D0-DF
         DC    X¢00000000000000000000000000000000¢   E0-EF
         DC    X¢F0F1F2F3F4F5F6F7F8F9FAFBFCFDFEFF¢   F0-FF
/+
/*
/&
* $$ EOJ
```

Figure 69. Job to Catalog Translation Table Source Code

```
* $$ JOB JNM=MTKXLATE,CLASS=A
* $$ PUN DISP=I,CLASS=A
// JOB MTKXLATE
// OPTION DECK
// LIBDEF SOURCE,SEARCH=TCPOCS.FTPC
// EXEC ASSEMBLY
         PUNCH ¢// JOB CATOBJ¢
         PUNCH ¢// EXEC LIBR,PARM=¢¢MSHP;A S=TCPOCS.FTPC¢¢¢
         PUNCH ¢   CATALOG MTKXLATE.OBJ REPLACE=Y¢
         END
/*
// EXEC ASSEMBLY
         COPY  MTKXLATE
         END
/*
// EXEC ASSEMBLY
         PUNCH ¢/*¢
         PUNCH ¢/&&¢
         END
/*
/&
* $$ EOJ
```

Figure 70. Job to Assemble and Catalog the Translation Object Module

Chapter 8. OC/Line Printer Daemon Installation and Customization

This chapter covers the installation and customization of OC/Line Printer Daemon. It consists of two steps:

1. Installation - Describes the preparation and procedures to install OC/Line Printer Daemon in VSE/ESA
2. Customization - Provides a detailed explanation of the product customization

8.1 OC/Line Printer Daemon Installation

There are two major steps involved to install the product:

1. Prepare the installation environment for the product
2. Install the product from tape to the VSE/ESA sublibrary

8.1.1 Installation Preparation

To prepare for installation of the OC/Line Printer Daemon the following definitions are required:

- OCS II Gateway to VSE/ESA

  Please refer to Chapter 5, "Defining the OCS II Gateway to the Host" on page 41 for the setup instructions.

- The VSE/ESA library and sublibraries for the product installation

  Library TCPOCS definition is discussed and defined in 6.1, "OC/FTP Server Installation" on page 61.

  Sublibrary TCPOCS.LPD is defined using the LIBR DEF SUB command in ICCF. This sublibrary is used to store the OC/Line Printer Daemon product, and the Line Printer Daemon customization.

- VSAM catalog and space definitions for printing to a data set instead of a printer

  This is an optional step. It is only required if you intend to print to a VSAM file instead of a VSE/POWER queue.

The OC/Line Printer Daemon runs mutually exclusive with the AIX LPD on the OCS II Gateway. This means that:

- The AIX qdaemon subsystem must be stopped. This is done with the following command:

  stop -s qdaemon

  If the qdaemon subsystem is automatically started at AIX startup, the qdaemon entry must be removed from ′/etc/inittab′.

- If you are using printers attached to the OCS II Gateway from remote Line Printer Requestors (LPRs), the following steps must be performed:
  - The printers must be physically connected to an AIX system other than the OCS II Gateway.

– The local printer queues must be removed from the OCS II Gateway and defined in the other AIX system.

– The former local printer queues on OCS II Gateway must be defined as remote printer queues.

8.1.2 Installation Procedures

Installation of the product from tape to the product sublibrary involves the following steps:

- Scan the product tape by submitting the job in Figure 35 on page 63 to VSE/ESA.
- Use the VSE/ESA II *′Install Product(s) from Tape′* panel to generate the job to install the product into sublibrary TCPOCS.LPD.

  The VSE/ESA II panel also displays the minimum library space requirements for the product.
- Submit the job to VSE/ESA to install OC/Line Printer Daemon. Our installation step took less than 10 minutes.

There is no installation verification job provided by the software. You may use the sample job in Figure 36 on page 64, to verify that the product is recorded by MSHP. Testing of the product can only be performed after the product customization. Please refer to Chapter 20, "OC/Line Printer Daemon Operation and Examples" on page 273 for the testing and operation procedures.

8.2 OC/Line Printer Daemon Customization

The customization is based on the samples in PRODBLDL.PROC provided by OC/Line Printer Daemon. After the product is installed in the product sublibrary, PRODBLDL.PROC is punched into an ICCF library. It is then edited into individual jobs to tailor the environment for OC/Line Printer Daemon.

The OC/Line Printer Daemon customization consists of the following steps:

- VTAM customization
- LPD customization
- Queue definition in AIX
- Startup customization

8.2.1 VTAM Customization

The following customization tasks are required in VTAM:

- Define the logmode table and entries for the applications and LUs
- Specify the APPL definitions for the OC/Line Printer Daemon
- Add the LU definitions for the OCS II Gateway LUs

8.2.1.1 Logmode Table OCSBIND

The sample job listed in A.6, "OCSBIND MODETAB for OCS Products" on page 351 is used to create the OCSBIND logmode table. OC/Line Printer Daemon uses the following entries:

- OCSLU0 for the OCS II Gateway LU used by OC/Line Printer Daemon to receive incoming Line Printer Requests
- OCSLU0 for the OCS II Gateway LUs used to send E-mail notifications to the print job originators

8.2.1.2 VTAM APPL Definitions

The OC/Line Printer Daemon uses:

- One VTAM APPL to receive incoming line printer requests. This APPL definition is shown below

```
OCSLPD01 APPL  MODETAB=OCSBIND,DLOGMOD=OCSLU0,AUTH=(NVPACE,ACQ),EAS=2
```

Figure 71. VTAM APPL Statement for OC/Line Printer Daemon Receive Connection

  Key parameters are:

  - ***MODETAB and DLOGMOD.*** Identify the appropriate logmode table entry.
  - ***NVPACE in AUTH parameter.*** Turns off VTAM pacing for FTP.
  - ***ACQ in AUTH parameter.*** Allows OC/FTP Server to acquire LUs.
  - ***EAS=2.*** Allows two concurrent sessions for OC/FTP Server.

- Multiple VTAM APPLs to send E-mail notifications. The OC/Line Printer Daemon product allows the usage of the OC/FTP Client VTAM definitions as described in 7.2.2, "VTAM Customization" on page 83.

  By using the OC/FTP Client VTAM definitions the E-mail notifications will be sent using the set of connections available to the OC/FTP Client.

  Warning

  If all connections assigned to OC/FTP Client are in use, no E-mail notification will be sent.

  These APPL definitions are only required if you intend to use the E-mail notification feature described in 8.2.2.1, "LPDCONF Member" on page 112.

8.2.1.3 VTAM LU Definition

The completed OCS II Gateway LUs definition is listed in Figure 24 on page 47, and A.3, "VSE TCPSW.B Switched Major Node" on page 342.

Figure 72 on page 112 lists the LU definitions used by OC/Line Printer Daemon. We defined one LU for the Line Printer Daemon and used the LUs defined for the OC/FTP Client as shown in Figure 52 on page 87. The number of LUs depends on the installation requirements.

```
IPFT2S9I LU    LOCADDR=2,DLOGMOD=OCSLU0,                              C
               ISTATUS=ACTIVE,MODETAB=OCSBIND,                        C
               MDLTAB=VTMMDL,MDLENT=VSELU2A,                          C
               SSCPFM=USSSCS,USSTAB=VTMUSSTR
```

Figure 72. Logical Unit Definitions for OC/Line Printer Daemon

The key parameters are:

- ***The LU name.*** Must match the LU name in the OCS II Gateway customization
- ***LOCADDR.*** Must match the LU number in the OCS II Gateway customization.
- ***MODETAB and DLOGMOD.*** Identify the appropriate logmode table entry.

8.2.2 LPD Customization

This section covers the definitions required by OC/Line Printer Daemon.

8.2.2.1 LPDCONF Member

The LPDCONF member is used to:

- relate the LPD function to VTAM
- define the E-mail Gateway used by the E-mail notification feature
- define the printer queues available to remote Line Printer Requestors

Figure 73 on page 113 lists the job we tailored.

```
 * $$ JOB JNM=LPDCONF,DISP=D,PRI=3,NTFY=YES,LDEST=(,RSCS),CLASS=0
 * $$ LST CLASS=A
 // JOB LPDCONF
 // EXEC LIBR,PARM=¢MSHP¢
 ACCESS SUBLIB=TCPOCS.PARMLIB
    CATALOG LPDCONF.LPDPARM EOD=$$ REPLACE=Y
 *-------------------------------------------------------*/
 * THIS MEMBER WILL BE USED TO IDENTIFY VARIOUS INITIAL-   */
 * IZATION CHARACTERISTICS FOR ALL LPD SESSION USERS.      */
 * THIS MEMBER NAME MUST NOT BE CHANGED.                   */
 *-------------------------------------------------------*/
 *-------------------------------------------------------*/
 * THE GATEWAY STATEMENT IDENTIFIES THE GATEWAY LU CHOICE. */
 *-------------------------------------------------------*/
 GATEWAY IPFT2S9I
 *-------------------------------------------------------*/
 * THE MAIL STATEMENT IDENTIFIES THE MAIL IP ADDRESS.      */
 *-------------------------------------------------------*/
 MAIL 192.61.100.81
 *-------------------------------------------------------*/
 * THE ACB STATEMENT IDENTIFIES THE VTAM ACB CHOICE.       */
 *-------------------------------------------------------*/
 ACB OCSSMP01
 *-------------------------------------------------------*/
 * THE QUEUE STATEMENT(S) IDENTIFY REMOTE QUEUE ELEMENTS.  */
 *-------------------------------------------------------*/
 QUEUE LPDFILE DDNAME=LPDFILE
 QUEUE SYSOUT SYSOUT=A,H=NO
 QUEUE VSEPRT SYSOUT=A,MAIL=NO
 QUEUE VSEPRTM SYSOUT=A,MAIL=YES
$$
/*
/&
* $$ EOJ
```

Figure 73. LPDCONF Member for OC/Line Printer Daemon

Listed below is a brief description of the parameters:

- ***GATEWAY.*** Identifies the VTAM LU used by the OC/Line Printer Daemon. It must match the LU defined for the OC/Line Printer Daemon in the VTAM LU Definition.
- ***MAIL.*** The IP address of the Mail server. We used the mail server on the RISC/6000 used for the OpenConnect Server II Gateway.
- ***ACB.*** Relates the LPD function to VTAM. It must match the name defined for OC/Line Printer Daemon in the VTAM APPL statement.
- ***QUEUE.*** Each usage of the QUEUE parameter defines a Line Printer queue for remote Line Printer Requestors. The QUEUE parameter supports the following operands:
  - ***QUEUENAME*** The name of the printer queue to be specified by the remote Line Printer Requestors.
  - ***QUEUETYPE*** This operand is used to determine the destination of the print queue specified on the QUEUENAME operand. The operand is built using:
    - The DDNAME= operand if printing to a data set

- A combination of the SYSOUT=, D=, and F= operands if printing to the VSE/POWER LST queue. The default value for this operand is ***SYSOUT=A, F=STD***

– ***DDNAME=*** The standard label of the file into which Line Printer Requests for this queue are stored.

  This standard label must be defined using a DLBL statement.

  Note

  OC/Line Printer Daemon supports any VSE/VSAM SAM ESDS and most OEM DASD manager organisations. We only tested with VSE/VSAM SAM ESDS files with the print to file feature.

  This operand is mutually exclusive to the SYSOUT=, D=, and F= operands.

– ***SYSOUT=*** The Identifier of the VSE/POWER LST class used for printing Line Printer Requests spooled to this queue. This operand is mutually exclusive with the DDNAME operand.

– ***D=*** The VSE/POWER destination assigned to Line Printer Requests spooled to this queue.

  Note

  In contrast to the VSE/POWER ***DEST=(node,userid)***, node and userid on this operand are not separated by a comma but a dot and are not enclosed by brackets.

  D=node.userid

  This operand is mutually exclusive with the DDNAME operand.

– ***F=*** The VSE/POWER form assigned to Line Printer Requests spooled to this queue. This operand is mutually exclusive with the DDNAME operand. The default for this operand is F=STD.

- ***MAIL=*** specifies if the originator of the Line Printer Request should be notified by E-mail when his request is being queued. The default for this operand is MAIL=YES.

  Note

  To use this option, the VTAM definitions for the OC/Line Printer Daemon E-mail function must be performed. We used the OC/FTP Client definitions as defined in 7.2.2, "VTAM Customization" on page 83.

- ***H=*** specifies if the print output should be preceded by the LPD banner. For a more detailed description of the LPD banner, refer to 8.2.5, "Customizing the OC/Line Printer Daemon Print Banner" on page 119.

  The default for this operand is H=Y.

For our LPDCONF member as shown in Figure 73 on page 113 this means that there are four queues available to remote Line Printer Requestors.

Queue name	Line print requests to this queue will be ...
LPDFILE	... preceded by the LPD banner and then stored in the file with label LPDFILE. After the request has been successfully filed, the originator will be notified by E-mail
SYSOUT	... spooled to VSE/POWER LST queue with CLASS=A,FORM=STD
VSEPRT	... preceded with the LPD banner and added to VSE/POWER LST queue with CLASS=A,FORM=STD. After the request has been successfully spooled to the VSE/POWER LST queue, the originator will be notified by E-mail
VSEPRTM	... preceded with LDP Banner, added to VSE/POWER LST queue with CLASS=A, FORM=STD. After the request has been successfully spooled to the VSE/POWER LST queue, the originator will be notified by E-mail

Table 6. LPD Queue Definitions used in our Environment

8.2.2.2 Additional Configuration Members

In order to use the E-mail notification function of the OC/Line Printer Daemon the following members have to be supplied:

- ***FAPPL.*** Defines the programs used by E-mail notification
- ***NETWORKS.*** Defines IP network names and addresses
- ***NETWORKS LU List.*** Allocates LUs to the networks defined in the NETWORKS member.
- ***RESOLVER.*** Defines the IP domain name servers.
- ***NETMASK.*** Defines the IP subnet mask.

Since we are using the OC/FTP Client connections for the E-mail notification, we just refer to these members in the OC/Line Printer Daemon startup job as described in 8.2.3.2, “Create the LPD Startup Job” on page 117.

8.2.3 Startup Customization

The following preparation tasks are required to bring up OC/Line Printer Daemon:

- Build the executable phases and apply site protection
- Create the LPD startup job

8.2.3.1 Build the Executable Phases and Apply Site Protection

Figure 74 on page 116 lists the job we used to assemble and link-edit the OC/Line Printer Daemon phase into the TCPOCS.LPD sublibrary and additionally apply the site protection modification. The source of this job is provided in PRODBLDL.PROC.

```
* $$ JOB JNM=PRODBLDL,DISP=D,PRI=3,NTFY=YES,LDEST=(,RSCS),CLASS=0
* $$ LST CLASS=A
// JOB PRODBLDL
*  ---------------------------------------------------  *
*  BUILDING COMPONENT EXECUTION MODULE(S).              *
*  ---------------------------------------------------  *
// OPTION CATAL,NOXREF
// LIBDEF OBJ,SEARCH=TCPOCS.LPD,TEMP
// LIBDEF SOURCE,SEARCH=TCPOCS.LPD,TEMP
// LIBDEF PHASE,SEARCH=TCPOCS.LPD,CATALOG=TCPOCS.LPD,TEMP
/*
   PHASE LPDMAIL,*
    INCLUDE LPDMAIL
/*
// EXEC PGM=LNKEDT,PARM=¢MSHP¢
/*
   PHASE LPD,*
    INCLUDE LPDTVT
    INCLUDE LPD
    INCLUDE LPDCPU
    INCLUDE LPDCONF
    INCLUDE LPDALLOC
    INCLUDE IJDFAZZW
    INCLUDE SOCK$AM
    INCLUDE SOCK$MF
    INCLUDE TABLE001
    INCLUDE SOCK$PL
    INCLUDE SOCK$VO
    INCLUDE SOCK$EX
    INCLUDE SOCK$CIS
    INCLUDE SOCK$V62
    INCLUDE SOCK$TR
    INCLUDE SOCK$DNR
    INCLUDE SPAM
    INCLUDE
/*
// EXEC PGM=ASSEMBLY
   COPY LPDXLATE
   END ,
/*
// EXEC PGM=ASSEMBLY
   COPY LPDBANER
   END ,
/*
   ENTRY LPD
/*
// EXEC PGM=LNKEDT,PARM=¢MSHP¢
/*
```

Figure 74 (Part 1 of 2). Build the Executable Phases for OC/Line Printer Daemon and Apply Site Protection

```
*  ----------------------------------------------------  *
*  APPLYING LICENSE SITE PROTECTION MODIFICATION.         *
*  ----------------------------------------------------  *
// EXEC PGM=MSHP
   REMOVE  5758-PC-418-100 APAR=SP41801
   CORRECT 5758-PC-418-100:SP41801 IRREVOKABLE
     AFFECTS PHASE=LPD
       ALTER 0000 /16/AA:DE90F69F2F7977885B6B6DDADCD2FB8B
       ALTER 0010 /16/AA:A3B2A6B1057D02D8DFC4708742F0DA4E
       ALTER 0020 /16/AA:89CDBA19A46646A3DB24D712EBCA6791
     RESOLVES ¢LICENSE SITE PROTECTION¢
/*
// IF $RC<=4 THEN
// GOTO BYPASS
*  ----------------------------------------------------  *
*  LICENSE SITE PROTECTION FAILURE; TERMINATED.           *
*  ----------------------------------------------------  *
// PAUSE
// GOTO $EOJ
/*
/. BYPASS
* $$ EOJ
```

Figure 74 (Part 2 of 2). Build the Executable Phases for OC/Line Printer Daemon and Apply Site Protection

8.2.3.2 Create the LPD Startup Job

The source of this job is provided in EXAMPLEL.PROC. We added the necessary JCL to start OC/Line Printer Daemon in a dynamic partition.

Figure 75 lists the job we used.

```
* $$ JOB JNM=LPD,CLASS=Y,DISP=L,NTFY=YES
* $$ LST CLASS=A
// JOB LPD START LINE PRINTER DAEMON
// LOG
// DLBL OCSPLIB,¢TCPOCS.PARMLIB.FTP¢
// DLBL IJSYS01,¢IJSYS01¢,0,VSAM,CAT=VSESPUC,DISP=(,KEEP),              C
               RECSIZE=4096,RECORDS=(000512,000256)
// DLBL LPDFILE,¢LPDFILE¢,0,VSAM,CAT=VSESPUC,DISP=(,KEEP),              C
               RECSIZE=4096,RECORDS=(000512,000256)
// LIBDEF *,SEARCH=(TCPOCS.PARMLIB,TCPOCS.LPD),TEMP
// EXEC PGM=LPD,SIZE=LPD
/*
* $$ EOJ
```

Figure 75. OC/Line Printer Daemon Startup Job

The DLBL statements in this job are used for the following reasons:

- ***DLBL OCSPLIB*** causes OC/Line Printer Daemon to use the configuration members needed for E-mail notification from TCPOCS.PARMLIB with a member type of FTP. These are the same members used when setting up the OC/FTP Client.

- ***DLBL IJSYS01*** changes the VSAM characteristics of the IJSYS01 workfile in order to use this file as a buffer for OC/Line Printer Daemon.
- ***DLBL LPDFILE*** specifies the characteristics for the LPDFILE data set that will be implicitly defined as a VSAM SAM ESDS file when a print request for the LPDFILE queue is performed.

This job is submitted to the POWER RDR queue and is ready to start. The operation and testing of OC/Line Printer Daemon are discussed in Chapter 20, “OC/Line Printer Daemon Operation and Examples” on page 273.

8.2.4 Customizing the OC/Line Printer Daemon Translation Tables

This step is optional. OC/Line Printer Daemon uses an assembler source file ′LPDXLATE.A′ for the data translation table. Refer to Figure 74 on page 116 for the inclusion of ′LPDXLATE′ during the assembly of the startup phase.

The customization of OC/Line Printer Daemon translation table consists of the following steps:

1. Change the translation table source code. Please refer to 6.2.5, “Customizing the Translation Tables” on page 77 for changing the EBCDIC and ASCII tables.
2. Assemble and link-edit the ′LPD′ phase to include the new translation table object module.
3. Restart the OC/Line Printer Daemon.

Figure 76 on page 119 shows the translation source code in hexadecimal. Finally, we resubmit the job shown in Figure 74 on page 116 to rebuild the ′LPD′ phase.

```
* $$ JOB JNM=CATXLATE,CLASS=A
// JOB CATXLATE - CATALOG SOURCE TRANSLATION TABLE
// EXEC LIBR,PARM=¢MSHP¢
   ACCESS S=TCPOCS.LPD
   CATALOG LPDXLATE.A  REPLACE=YES
LPDXLATE CSECT
*-----------------------------------------------------------------------*
*                  TABLE FOR ASCII TO EBCDIC
*-----------------------------------------------------------------------*
         DC    A(EPDEBCDI) ADDRESS OF ASCII TO EBCDIC TABLE
LPDEBCDI DS    0F
*                 0 1 2 3 4 5 6 7 8 9 A B C D E F    ASCII
         DC    X¢00010203372D2E2F1605150B0C0D0E0F¢   00-0F
         DC    X¢101112003C3D322618193F271C1D1E1F¢   10-1F
         DC    X¢405A7F7B5B6C507D4D5D5C4E6B604B61¢   20-2F
         DC    X¢F0F1F2F3F4F5F6F7F8F97A5E4C7E6E6F¢   30-3F
         DC    X¢7CC1C2C3C4C5C6C7C8C9D1D2D3D4D5D6¢   40-4F
         DC    X¢D7D8D9E2E3E4E5E6E7E8E980E0905F6D¢   50-5F
         DC    X¢79818283848586878889919293949596¢   60-6F
         DC    X¢979899A2A3A4A5A6A7A8A9C06AD0A107¢   70-7F
         DC    X¢00000000000000000000000000000000¢   80-8F
         DC    X¢00000000000000000000000000000000¢   90-9F
         DC    X¢00000000000000000000000000000000¢   A0-AF
         DC    X¢00000000000000000000000000000000¢   B0-BF
         DC    X¢00000000000000000000000000000000¢   C0-CF
         DC    X¢00000000000000000000000000000000¢   D0-DF
         DC    X¢00000000000000000000000000000000¢   E0-EF
         DC    X¢F0F1F2F3F4F5F6F7F8F9FAFBFCFDFEFF¢   F0-FF
/+
/*
/&
* $$ EOJ
```

Figure 76. Job to Catalog Translation Table Source Code

8.2.5 Customizing the OC/Line Printer Daemon Print Banner

This step is optional. OC/Line Printer Daemon uses an assembler source member ′LPDBANER.A′ that contains the banner that precedes every Line Printer Request. Refer to Figure 74 on page 116 for the inclusion of ′LPDBANER′ during the assembly of the startup phase.

The customization of OC/Line Printer Daemon print banner consists of the following steps:

1. Change the print banner source code.

> **Warning**
>
> Follow the instructions in the default LPDBANER.A member when changing the Print Banner source code.

2. Assemble and link-edit the ′LPD′ phase to include the new print banner object module.
3. Restart the OC/Line Printer Daemon.

Figure 77 on page 120 shows the translation source code in hexadecimal. Finally, we resubmitted the job shown in Figure 74 on page 116 to rebuild the 'LPD' phase.

```
* $$ JOB JNM=CATBANER,CLASS=A
// JOB CATBANER - CATALOG SOURCE PRINT BANNER
// EXEC LIBR,PARM=¢MSHP¢
   ACCESS S=TCPOCS.LPD
   CATALOG LPDBANER.A  REPLACE=YES
         TITLE ¢LPD - BANNER DEFINITION FILE¢
*======================================================================
*   PROGRAM: LPDBANER
*   PURPOSE: Allow flexability for defining the banner page data
*            printed by the LPD Deamon.
*   WRITTEN: 07/15/96
*   SYSTEM:  VSE/ESA 2.1
*======================================================================
*   Those characters defined from BANER1N thru BANER1E define the
*   left half of the banner.
*   Those characters defined from BANER2N thru BANER2E define the
*   right half of the banner.
*   Change the characters between the ¢......¢ only and reassemble
*   and link using the provided sample jcl as a guide.
*   Warning: Both sides of the banner must contain the same number
*            of lines.
*======================================================================
LPDBANER CSECT
         DC    A(BANER1)
         DC    A(BANER2)
         DC    A(BANER1EE)
         DC    A(BANER2EE)
         DC    CL8¢LPDBANER¢           CSECT NAME
         DC    CL8¢08.00.00¢           TIME
         DC    CL8¢07/15/96¢           DATE
BANER1   DS    0H
BANER1N  DS    0H
  DC CL60¢                             0000000  CCCCCCC  SSSSSS     /¢
  DC CL60¢                            00   00  CC       SS    SS   / ¢
  DC CL60¢                           00   00  CC         SS       /  ¢
  DC CL60¢                          00   00  CC      SS   SS     /   ¢
  DC CL60¢                         0000000  CCCCCCCC  SSSSSSS   /    ¢
BANER1E  DS    0H
BANER1EE EQU   (*-BANER1)/60    NUMBER OF LINES
BANER2   DS    0H
BANER2N  DS    0H
  DC CL60¢      LL      PPPPPPPP  DDDDDDDD                           ¢
  DC CL60¢     LL      PP    PP  DD    DD                            ¢
  DC CL60¢    LL      PPPPPPPP  DD    DD                             ¢
  DC CL60¢   LL      PP        DD    DD                              ¢
  DC CL60¢ LLLLLL  PP        DDDDDDDD                                ¢
BANER2E  DS    0H
BANER2EE EQU   (*-BANER2)/60    NUMBER OF LINES
/+
/*
/&
* $$ EOJ
```

Figure 77. Job to Catalog Print Banner Source Code

8.2.6 Queue Definition in AIX

Every line printer queue for OC/Line Printer Daemon as described in 8.2.2, "LPD Customization" on page 112 should also be defined on the OCS II Gateway RISC/6000. This can be easily accomplished by the following SMIT path:

- Spooler (Print Jobs)
- Manage Remote Printer Subsystem
- Client Services
- Remote Printer Queues
- Add a Remote Printer Queue

Figure 78 shows the definitions used for the SYSOUT queue :

```
                              Add a Remote Queue

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

                                                        [Entry Fields]
* NAME of queue to add                               [sysout]
  ACTIVATE the queue?                                 yes                   +
  Will this become the DEFAULT queue?                 no                    +
  Queueing DISCIPLINE                                 first come first serve  +
  ACCOUNTING FILE pathname                           []
* DESTINATION HOST for remote jobs                   [192.61.100.81]
* Pathname of the SHORT FORM FILTER for queue        [/usr/lpd/aixshort]     +
    status output
* Pathname of the LONG FORM FILTER for queue         [/usr/lpd/aixlong]      +
    status output
* Name of QUEUE on remote printer                    [sysout]
* NAME of device to add                              [sysout]
* BACKEND PROGRAM pathname                           [/usr/lpd/rembak]
F1=Help             F2=Refresh          F3=Cancel           F4=List
Esc+5=Reset         F6=Command          F7=Edit             F8=Image
F9=Shell            F10=Exit            Enter=Do
```

Figure 78. OC/Line Printer Daemon Queue Definition on OCS II Gateway

Listed below is a short description of the entry fields:

- ***NAME of queue to add.*** The name under which the OC/Line Printer Daemon print queue will be known to AIX. We recommend to use the same name as defined in OC/Line Printer Daemon.
- ***DESTINATION HOST for remote jobs.*** The hostname or IP-address of the OCS II Gateway.
- ***Name of QUEUE on remote printer.*** The name of the OC/Line Printer Daemon print queue as specified in the LPDCONF member.
- ***NAME of device to add.*** Use the same name as for ***Name of QUEUE on remote printer*** for this field.
- All other SMIT fields contain default values that should not be overwritten.

Chapter 9. OC/RSH Client Installation and Customization

This chapter covers the installation and customization of the OC/RSH Client. It consists of two steps:

1. Installation - Describes the preparation and procedures to install OC/RSH Client in VSE/ESA
2. Customization - Provides an explanation of the product customization.

The installation instructions provided in this chapter assume you already have installed and are using the OCS II Gateway. The OC/RSH Client is an extension to the OC/FTP Client program. The OC/FTP Client has already been installed (see Chapter 7, "OC/FTP Client Installation and Customization" on page 81) and we are using the same parameter library members.

9.1 OC/RSH Client Installation

There are two major steps involved to install the product:

1. Prepare the installation environment for the product
2. Install the product from tape to the VSE/ESA sublibrary

9.1.1 Installation Preparation

To prepare for installation of the OC/RSH Client the following definitions are required:

- OCS II Gateway to VSE/ESA

  Please refer to Chapter 5, "Defining the OCS II Gateway to the Host" on page 41 for the setup instructions.

- The VSE/ESA library and sublibraries for the product installation

  Library TCPOCS definition is discussed and defined in 6.1, "OC/FTP Server Installation" on page 61.

  Sublibrary TCPOCS.RSH is defined using the LIBR DEF SUB Librarian command. This sublibrary is used to store the OC/RSH Client product, and the RSH client customization.

9.1.2 Installation Procedures

Installation of the product from tape to the product sublibrary involves the following steps:

- Use the VSE/ESA II Installation dialog to generate the installation jobstreams.
- Scan the product tape by submitting the job in Figure 35 on page 63.
- Submit the job created using the VSE/ESA II Installation dialog to install OC/RSH Client into the TCPOCS.RSH sublibrary.

There is no installation verification job provided by the software. You may use the sample job in Figure 36 on page 64 to verify that the product is recorded by MSHP. Testing of the product can only be performed after the product customization. Please refer to Chapter 21, “OC/RSH Client Operation and Examples” on page 281 for the testing and operation procedures.

9.2 OC/RSH Client Customization

The customization is based on the samples in PRODBILD.PROC provided by OC/RSH Client. After the product is installed in the product sublibrary, PRODBILD.PROC is punched into an ICCF library. It is then edited into individual jobs to tailor the environment for OC/RSH Client.

The OC/RSH Client customization consists of the following steps:

- VTAM customization
- CICS customization
- RSH client customization
- Remote Command Execution Access at RS/6000
- Startup customization

Since we have already installed the OC/FTP Client and no additional LUs are being added to support OC/RSH Client, no further VTAM and RSH customization is required.

9.2.1 VTAM Customization

The following customization tasks are required in VTAM:

- Define the logmode table and entries for the applications and LUs
- Specify the APPL definitions for the OC/RSH Client
- Add the LU definitions for the OCS II Gateway LUs

These tasks are the same as for the OC/FTP Client and described in Chapter 7, “OC/FTP Client Installation and Customization” on page 81.

9.2.1.1 VTAM APPL Definitions

In addition to the discussions in Chapter 7, “OC/FTP Client Installation and Customization” on page 81 an understanding of how RSH works is helpful.

Figure 79 on page 126 illustrates the operation of the OC/RSH Client and the relationships between FTP and RSH clients within the OC/FTP Client.

- VTAM, CICS and OCS II Gateway are initialized, and the OCS II Gateway is connected to VTAM.
- The OC/RSH Client is executed within the OC/FTP Client dynamic partition, the communication program OCSFTP62 automatically establishes the parallel sessions with the corresponding ISC modules in the CICS partition. The number of concurrent sessions is determined in the OCSFTP62 APPL statement. The OC/RSH Client is now ready to accept RSH transactions from CICS terminals. OCSFTP62 handles also the requests from FTP transactions.
- When the RSH transaction is entered from a CICS terminal, the OC/RSH Client online interface is initialized.

- When the OC/RSH Client is invoked from a batch job, the OC/RSH Client runs in the batch dynamic partition and initializes the batch interface.
- After the online or batch interface is started, the OC/RSH Client must ′know′ its parameter library. This is done via the OC/RSH Client PARMLIB command. Now OC/RSH Client is ready to send commands to a remote TCP/IP host.
- For each command execution, the OC/RSH Client dispatches a program with the name ′OCSFTP**nn**′, where ′**nn**′ ranges from 01 to the maximum number of sessions allowed. OCSFTPnn creates an LU-LU session using the first available LU in the OCS II Gateway.
- OCSFTPnn is a VTAM application and must be defined in the VTAM APPL major node. Keep in mind that we use the same applications we defined for OC/FTP Client.

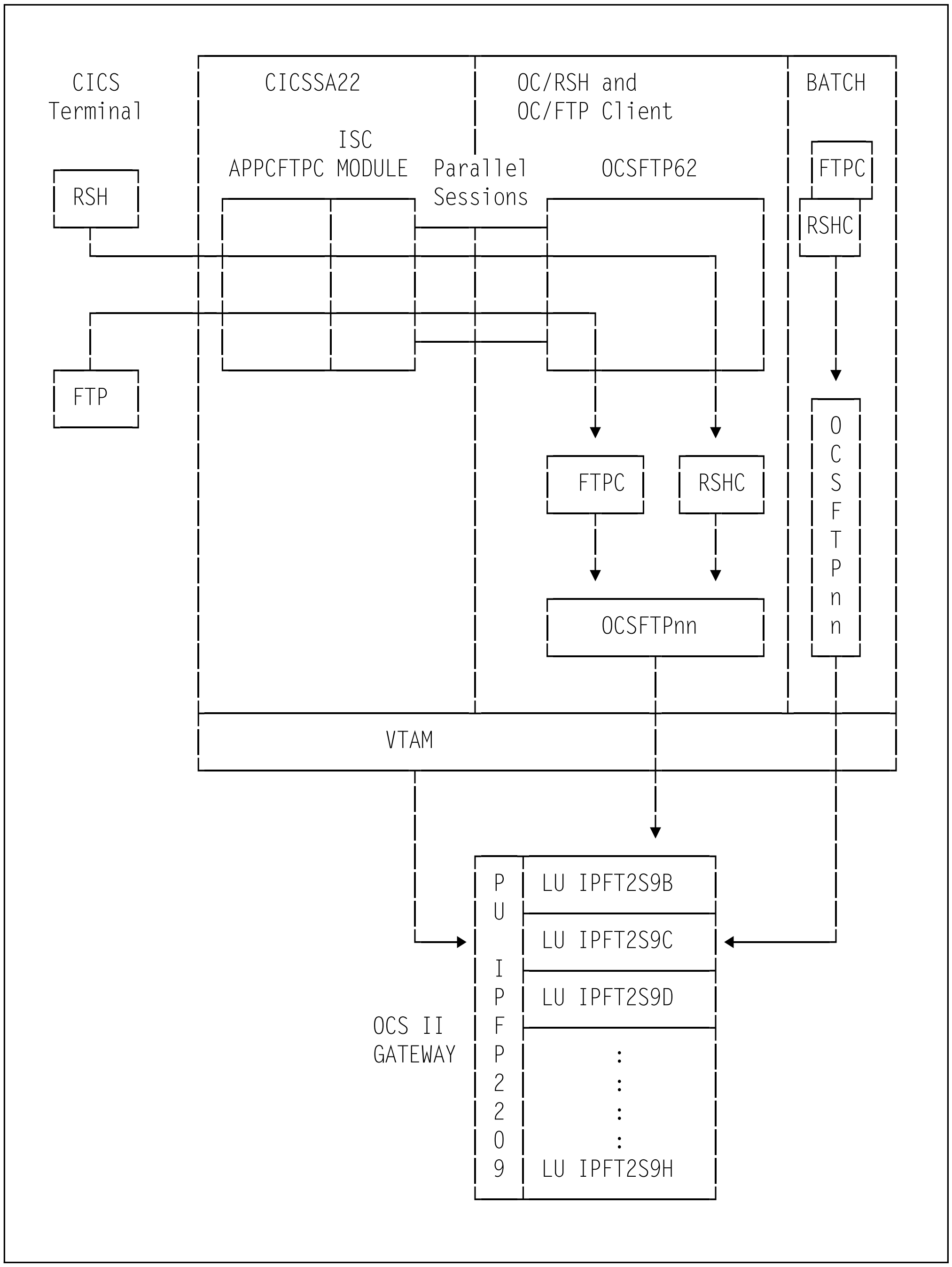


Figure 79. OC/RSH Client Operation

9.2.2 CICS Customization

This section covers the definitions in CICS to enable VSE/ESA users to use the OC/RSH Client online. Our definition is based on the samples in PRODBILD.PROC provided by OC/RSH Client. We used the CEDA transaction to define the following resources to CICS:

- Connection VSE
- Session VSESESS
- Program APPCFTPC

- Transaction RSH

Connection, session and program definitions are the same as described for OC/FTP Client (see Chapter 7, “OC/FTP Client Installation and Customization” on page 81). We only show the definition of the transaction ′RSH′ in the section below.

9.2.2.1 Define the ″RSH″ Transaction

CEDA DEF TRAN(RSH) is used to define the OC/RSH Client transaction RSH to CICS as shown below.

```
CEDA  DEFINE
 TRansaction    : RSH
 Group          : VSEGROUP
 PROGram        : APPCFTPC
 TWasize        : 00256             0-32767
 PROFile        : DFHCICST
 PArtitionset   :
 STatus         : Enabled           Enabled | Disabled
 PRIMedsize     : 00000             0-65520
REMOTE ATTRIBUTES
 DYnamic        : No                No | Yes
 REMOTESystem   :
 REMOTEName     :
 TRProf         :
 Localq         :                   No | Yes
SCHEDULING
 PRIOrity       : 001               0-255
 TClass         : No                No | 1-10
ALIASES
 Alias          :
 TAskreq        :
 Xtranid        :
RECOVERY
 DTimout        : No                No | 1-7000
 Indoubt        : Backout           Backout | Commit | Wait
 REStart        : No                No | Yes
 SPurge         : No                No | Yes
 TPurge         : No                No | Yes
 DUmp           : Yes               Yes | No
 TRACe          : Yes               Yes | No
SECURITY
 Extsec         : No                No | Yes
 TRANsec        : 01                1-64
 RSL            : Public            0-24 | Public
 RSLC           : No                No | Yes | External
```

The key parameters are:

- ***TRANSACTION.*** The name of the OC/RSH Client transaction.
- ***GROUP.*** The name of the group to which this definition belongs.
- ***PROGRAM.*** The first CICS program that this transaction invokes.
- ***TWASIZE.*** The transaction work area. 256 bytes is recommended by OCS.

9.2.3 RSH Client Customization

This section covers the definitions required for the RSH client. The following library members are needed:

- ***FAPPL.*** Defines the programs used by the OC/RSH Client.
- ***NETWORKS.*** Defines IP network names and addresses.
- ***NETWORKS LU List ″IPFNET″.*** Allocates LUs to the networks defined in the NETWORKS member.
- ***HOSTS.*** Defines TCP/IP IP addresses and host names.
- ***NETMASK.*** Defines the IP subnet mask.

These members have already been set up for the OC/FTP Client in Chapter 7, "OC/FTP Client Installation and Customization" on page 81 and filed in sublibrary TCPOCS.FTPC.

9.2.4 Prepare for Remote Command Execution Access at an AIX Host

The OC/RSH Client is able to execute a command or a sequence of commands on a remote TCP/IP host only if a user is allowed to do so. That means, access for remote command execution first has to be provided on the ′target′ TCP/IP host.

To understand the remote command execution, it is necessary to know that the following authorization information is passed from the OC/RSH Client on VSE/ESA to the RSH Server on the target TCP/IP node:

Client User ID
: The user ID of the RSH client. When using the OC/RSH Client batch program this is always ′batch′.

User ID at the RSH Server
: The user ID to be used at the local RSH server. This is specified in the -l flag of the RSH command and is mandatory in the OC/RSH Client.

To obtain the authorization for remote command execution on a TCP/IP RSH server host the following is required:

1. A valid user ID at the RSH server
2. The VSE user ID for the RSH client needs to be authorized for remote command execution. This can be done in two different ways:
    a. Specifying the client user ID as well as the VSE/ESA system itself in the /etc/hosts.equiv file of the RSH server. Then no further authorization check is performed. Figure 80 on page 129 shows the /etc/hosts.equiv file we used.
    b. The client user ID as well as the VSE/ESA system itself is defined in the $HOME/.rhosts file; then the name of the server system needs also to be defined in the $HOME/.rhosts file. Figure 81 on page 129 shows the $HOME/.rhosts file we used.

```
aix320           # allows access to VSE/ESA system
aix320 batch     # allows access to VSE/ESA RSH Batch program itself
aix320 maru      # allows access to VSE/ESA user maru
aix320 wacker    # allows access to VSE/ESA user wacker
aix320 jyiu      # allows access to VSE/ESA user jyiu
aix320 kami      # allows access to VSE/ESA user kami
aix320 user      # allows access to VSE/ESA user user
# other VSE/ESA host users need access via $HOME/.rhosts file
```

Figure 80. Sample /etc/hosts.equiv File

```
aix320 wack      # allows access to VSE/ESA user wack
```

Figure 81. Sample $HOME/.rhosts File for User wack

9.2.5 Startup Customization

The following is required to bring up the OC/RSH Client:

- Build the executable phases
- Run the site protection modification job
- Create the RSH Client startup job
- Create the sample RSH client batch job

9.2.5.1 Build the Executable Phase

Figure 82 lists the job we used to assemble and link-edit the OC/RSH Client phase into the TCPOCS.RSH sublibrary. The source of this job is provided in PRODBILD.PROC.

```
* $$ JOB JNM=CRASM,CLASS=5,DISP=D,NTFY=YES
// JOB CRASM
*  BUILDING COMPONENT EXECUTABLE PHASE
// OPTION CATAL,NOXREF
// LIBDEF OBJ,SEARCH=(TCPOCS.RSH,TCPOCS.FTPC),TEMP
// LIBDEF PHASE,CATALOG=TCPOCS.RSH,TEMP
   PHASE RSHC,*
    INCLUDE RSHMAIN
    INCLUDE RSHCPU
    INCLUDE RSHBTCH
    INCLUDE RSHSMP
    INCLUDE SOCK$AM
/*
   ENTRY RSH
/*
// EXEC PGM=LNKEDT,PARM=¢MSHP¢
/*
/&
* $$ EOJ
```

Figure 82. Build the Executable Phases for OC/RSH Client

9.2.5.2 Run the Site Protection Modification Job

Figure 83 lists the job we used for the OCS license protection. The source of this job is provided in PRODBILD.PROC.

```
* $$ JOB JNM=CRASM2,CLASS=5,DISP=D,NTFY=YES
// JOB CRASM2
*  ----------------------------------------------------  *
*  APPLYING LICENSE SITE PROTECTION MODIFICATION.         *
*  ----------------------------------------------------  *
// EXEC PGM=MSHP
   REMOVE  5758-PC-317-215 APAR=SP31701
   CORRECT 5758-PC-317-215:SP31701 IRREVOKABLE
     AFFECTS PHASE=RSHC
       ALTER 0000 /16/AA:DD3126AC98742608270864AACDB6B274
       ALTER 0010 /16/AA:A34C76D331729CA8F65F7DAB4212294F
       ALTER 0020 /16/AA:FBCD1709EE4B10351D4D5CFCFC9F4291
     RESOLVES ¢LICENSE SITE PROTECTION¢
/*
/&
* $$ EOJ
```

Figure 83. Site Protection Job for OC/RSH Client

9.2.5.3 Create the RSH Client Startup Job

The source of this job is provided in RSHSAMP.PROC. We added the necessary JCL to start OC/FTP Client in a dynamic partition.

Figure 84 lists the job we used.

```
* $$ JOB JNM=FTPC,DISP=L,PRI=3,NTFY=YES,LDEST=(,RSCS),CLASS=Z
* $$ LST CLASS=A
// JOB FTPC  STARTS FTP AND RSH CLIENT
// LIBDEF *,SEARCH=(TCPOCS.FTPC,PRD2.CONFIG),TEMP
// EXEC PGM=OCSFTPC,SIZE=(OCSFTPC,256K)
/*
/&
* $$ EOJ
```

Figure 84. OC/RSH Client Startup Job

This job is submitted to the POWER RDR queue and is ready to start. Operation and test of the OC/RSH Client is discussed in Chapter 21, "OC/RSH Client Operation and Examples" on page 281.

9.2.5.4 Create the Sample RSH Client Batch Job

The sample RSH client job in RSHSAMP.PROC may be customized, to start and execute OC/RSH Client in a batch partition.

Be sure that user names and commands for AIX are specified in lower case, otherwise the command will not execute as expected. When preparing the RSH batch job with the ICCF editor use the editor command ″CASE M″ to enable input in mixed mode. Figure 85 lists the job we tailored to test RSH functions.

```
* $$ JOB JNM=CRBATCH,DISP=L,PRI=3,NTFY=YES,LDEST=(,RSCS),CLASS=Y
* $$ LST CLASS=A
// JOB CRBATCH
// LIBDEF PHASE,SEARCH=TCPOCS.FTPC,TEMP
// LIBDEF SOURCE,SEARCH=TCPOCS.FTPC,TEMP
// EXEC PGM=RSHC,SIZE=(RSHC,256K)
PARMLIB TCPOCS.FTPC
RSH AIX320 -l wacker -n date; echo $HOME;ls -l
RSH AIX320 -l wacker -n cat /etc/hosts
RSH AIX320 -l maru -n ls -l   grep Oct
RSH 192.61.100.81 -l maru -n ls -l   grep Oct
BYE
/*
/&
* $$ EOJ
```

Figure 85. Sample OC/RSH Client Batch Job

Chapter 10. OC/TELNET FS Installation and Customization

This chapter covers the installation and customization of OC/TELNET FS. It consists of two steps:

1. Installation - Describes the preparation and procedures to install OC/TELNET FS in VSE/ESA
2. Customization - Provides a detailed explanation of the product customization

10.1 OC/TELNET FS Installation

There are two major steps involved to install the product:

1. Prepare the installation environment for the product
2. Install the product from tape to the VSE/ESA sublibrary

10.1.1 Installation Preparation

To prepare for installation of the OC/TELNET FS the following definitions are required:

- OCS II Gateway to VSE/ESA

  Please refer to Chapter 5, “Defining the OCS II Gateway to the Host” on page 41 for the setup instructions.

- The VSE/ESA library and sublibraries for the product installation

  Library TCPOCS definition is discussed and defined in 6.1, “OC/FTP Server Installation” on page 61.

  Sublibrary TCPOCS.TNFS is defined using the LIBR DEF SUB command in ICCF. This sublibrary is used to store the OC/TELNET FS product, and the TELNET client customization.

- VSAM catalog and space definitions for the TELNET customization

  We used the same VSAM catalog TCPCAT defined in 6.1, “OC/FTP Server Installation” on page 61 for the OC/TELNET FS.

10.1.2 Installation Procedures

Installation of the product from tape to the product sublibrary involves the following steps:

- Scan the product tape by submitting the job in Figure 35 on page 63 to VSE/ESA.
- Use VSE/ESA II ’*Install Product(s) from Tape*’ panel to generate the job to install the product into sublibrary TCPOCS.TNFS.

  The VSE/ES II panel also displays the minimum library space requirement for the product.

- Submit the job to VSE/ESA to install the product. Our installation step took less than 10 minutes.

There is no installation verification job provided by the software. You may use the sample job in Figure 36 on page 64, to verify that the product is recorded by MSHP. Testing of the product can only be performed after the product customization. Please refer to Chapter 22, “OC/TELNET FS Operation and Examples” on page 287 for the testing and operation procedures.

10.2 OC/TELNET FS Customization

Customization is based on the samples in PRODBILD.PROC provided by OC/TELNET FS. After the product is installed in the product sublibrary, PRODBILD.PROC is punched into an ICCF library. We then edited it into individual jobs to tailor the environment for OC/TELNET FS.

The OC/TELNET FS customization consists of three steps:

- VTAM customization
- TELNET Client customization
- Startup customization

10.2.1 VTAM Customization

The following customization tasks are required in VTAM:

- Logmode table and entries for the applications and LUs
- APPL definitions for the OC/TELNET FS
- LU definitions for the OCS II Gateway LUs
- USSTAB for VTAM logon to OC/TELNET FS

10.2.1.1 Logmode Table OCSBIND

The sample job listed in A.6, “OCSBIND MODETAB for OCS Products” on page 351 is used to create the OCSBIND logmode table. OC/TELNET FS uses the OCSLU0 entry listed in Figure 86 for the OCS II Gateway LUs.

```
OCSLU0   MODEENT  LOGMODE=OCSLU0,                                   C
               TYPE=X¢00¢                                           C
               FMPROF=X¢06¢,                                        C
               TSPROF=X¢03¢,                                        C
               PRIPROT=X¢00¢,                                       C
               SECPROT=X¢00¢,                                       C
               COMPROT=X¢0000¢,                                     C
               RUSIZES=X¢F8F8¢,                                     C
               PSERVIC=X¢000000000000000000000000¢
```

Figure 86. MODETAB Entry for OC/TELNET FS

10.2.1.2 VTAM APPL Definitions

There are two types of APPL statement for OC/TELNET FS:

- VSTELNET APPL statement for the OC/TELNET FS control application
- VSNET**nnn** APPL statement, where **nnn** ranges from 001 to the maximum number of concurrent TELNET sessions

Before we discuss the VTAM APPL definitions, it is important to understand how a VSE/ESA TELNET client establishes a session with the TCP/IP node using OC/TELNET FS.

- When a VSE/ESA TELNET client wants to establish a connection to a TCP/IP TELNET server, the client logs on to VSTELNET from a VTAM terminal LU
- VSTELNET dispatches the first available VSNETnnn application (for example VSNET001), and VSNETnnn is connected to the VTAM terminal with an LU-LU session
- When the VSE/ESA TELNET client selects the OPEN option to connect to a TCP/IP node, an LU-LU session is established between VSNETnnn and the first available TELNET LU in OCS II Gateway
- VSNETnnn passes the session request to OCS II Gateway, and a connection request is sent to the TCP/IP node
- After the TCP/IP node accepts the OPEN request, the VSE/ESA TELNET client is requested to log in to the TCP/IP node
- Once login is successful, the VTAM terminal emulates the TCP/IP node ASCII terminal

Since VSTELNET is the OC/TELNET FS control program, an APPL entry must be defined to VTAM in the APPL major node.

The number of VSNETnnn APPL entries depends on the number of concurrent VSE/ESA TELNET clients required. We defined 64 concurrent TELNET sessions using APPL entries VSNET001 to VSNET064 for our environment. The VSNETnnn application connects to the VTAM terminal LU and the OCS II Gateway LU at the same time during the TELNET session. The APPL definition must be able to support two concurrent sessions.

Figure 87 is an extract from the complete VTAM APPL definition listed in A.4, "VTAM APPL Major Node for OCS Products" on page 344.

```
VSTELNET APPL  AUTH=(ACQ,PASS),ACBNAME=VSTELNET
VSNET001 APPL  AUTH=(ACQ),EAS=2,ACBNAME=VSNET001
VSNET002 APPL  AUTH=(ACQ),EAS=2,ACBNAME=VSNET002
     :
     :
     :
VSNET064 APPL  AUTH=(ACQ),EAS=2,ACBNAME=VSNET064
```

Figure 87. APPL Definition Sample for OC/TELNET FS

The key definitions and parameters are:

- ***APPL entry for VSTELNET.*** The ACBNAME should be the same as the APPL label. It must also match the ACB name used in the SET MAJOR ACBNAME command of the SYSIN file. For details of the SYSIN file, please refer to 10.2.2.4, “SYSIN File” on page 139.

- ***VSNETnnn APPL entries.*** The number of VSNETnnn APPL entries depends on the number of concurrent TELNET sessions required and it should match the maximum number of users defined in the SYSIN file. The key parameters are:
  - ***ACBNAME.*** Should be the same as the APPL label.
  - ***The ACBNAME prefix VSNET.*** Must match the value specified in the SET MINOR ACBNAME command of the SYSIN file.
  - ***EAS=2.*** Allows two concurrent sessions.

10.2.1.3 VTAM LU Definition

The completed OCS II Gateway LUs definition is listed in Figure 24 on page 47, and A.3, “VSE TCPSW.B Switched Major Node” on page 342.

Figure 88 lists the LU definitions used by OC/TELNET FS. We defined three LUs for our test environment. To support the 64 sessions defined in the APPL major node, 64 LUs should be defined.

```
IPFT2S9J LU    LOCADDR=10,DLOGMOD=OCSLU0,                             C
               ISTATUS=ACTIVE,MODETAB=OCSBIND,                        C
               MDLTAB=VTMMDL,MDLENT=VSELU2A,                          C
               PACING=1,VPACING=2,                                    C
               SSCPFM=USSSCS,USSTAB=VTMUSSTR
IPFT2S9K LU    LOCADDR=11,DLOGMOD=OCSLU0,                             C
               ISTATUS=ACTIVE,MODETAB=OCSBIND,                        C
               MDLTAB=VTMMDL,MDLENT=VSELU2A,                          C
               PACING=1,VPACING=2,                                    C
               SSCPFM=USSSCS,USSTAB=VTMUSSTR
IPFT2S9L LU    LOCADDR=12,DLOGMOD=OCSLU0,                             C
               ISTATUS=ACTIVE,MODETAB=OCSBIND,                        C
               MDLTAB=VTMMDL,MDLENT=VSELU2A,                          C
               PACING=1,VPACING=2,                                    C
               SSCPFM=USSSCS,USSTAB=VTMUSSTR
```

Figure 88. Logical Unit Definitions for OC/TELNET FS

The key parameters are:

- ***The LU name.*** This must match:
  - The LU name in the OCS II Gateway customization
  - The LU name in the Network LU List Member
- ***LOCADDR.*** Must match the LU number in the OCS II Gateway customization.
- ***MODETAB and DLOGMOD.*** Identify the appropriate logmode table entry.
- ***PACING AND VPACING.*** Turn on VTAM pacing for the TELNET sessions.

10.2.1.4 VTAM USSTAB

The VTAM USSTAB is modified to allow VSE/ESA users to log on to VSTELNET from the VTAM logon menu. Please refer to A.5, "VTAM USSTAB" on page 345 for the definitions.

10.2.2 TELNET Client Customization

This section covers the definitions required by the TELNET client function of OC/TELNET FS. Under VSE/ESA, each OC/TELNET client command file is an individual sublibrary member in the OC/TELNET client sublibrary with a member type of ″PARMLIB″. The default members provided by OC/TELNET may not apply to your installation. You should review and tailor the command files discussed in this section. If you need to do any changes to the defaults, you should punch the members membername.PARMLIB into your ICCF library using the LIBRP command. From the ICCF command line enter the following command:

LIBRP TCPOCS.TNFS membername.PARMLIB membername.

After you define a command file, you must catalog it into the OC/TELNET client sublibrary under the name: membername.PARMLIB, where membername is the name of the command file. Catalog the changed members back using the following LIBRC command:

LIBRC TCPOCS.TNFS membername.PARMLIB REPLACE.

The following files are covered in this section:

- ***NETWORKS.*** Defines IP network names and addresses.
- ***NETWORKS LU List.*** Allocates LUs to the networks defined in the NETWORKS member.
- ***HOSTS.*** Defines TCP/IP IP addresses and host names.
- ***SYSIN.*** Defines system initialization commands during system initialization.
- ***DFLTSYS.*** Defines system default session commands.
- ***DFLTUSR*** Defines user default session commands.
- ***SYSCMDS.*** Defines the operator commands during system initialization.
- ***Terminal Keyboard Profiles.*** Defines the terminal keyboard mapping profiles for the IBM 3278 emulation.

The commands in the SYSIN, DFLTSYS, DFLTUSR, SYSCMDS and Terminal Keyboard Profile files are documented in *OpenConnect/TELNET Client Full Screen User and Administration Guide, 350-0145-101*.

10.2.2.1 NETWORKS File

The NETWORKS and NETWORK LU List members are used to define to OC/TELNET FS, the:

- IP network names and addresses
- OCS II Gateway LU assignments for a specific network

The concept is the same as the network definition and LUs assignment discussed in 7.2.4.3, “NETWORKS Member” on page 95.

Our definition is based on the network diagram in Figure 13 on page 34, and the IP address and name assignments in Table 3 on page 36. In the NETWORKS file, we defined the network IPFNET with IP network address 192.61.100.

Figure 89 lists the job we used. ″IPFNET″ is the only network defined and the last line without an IP address defines it as the default network.

```
CATALOG NETWORKS.PARMLIB  EOD=/+          REPLACE=YES
#
# SAMPLE NETWORKS FILE
#
IPFNET   192.61.100
IPFNET
/+
```

Figure 89. NETWORKS File for OC/TELNET FS

10.2.2.2 NETWORKS LU List File IPFNET

We used the IPFNET file to assign three LUs to the IPFNET network. The NETWORK LU List file name must correspond to the network name defined in the NETWORKS member. The LU names must correspond to the LU names defined for OCS II Gateway. We assigned LUs IPFT2S9J, K and L to the IPFNET network.

Figure 90 lists the job we tailored.

```
CATALOG IPFNET.PARMLIB                    REPLACE=YES
**********************************************************************
* SAMPLE GATEWAY
*
GATEWAY BEGIN
LU IPFT2S9J
LU IPFT2S9K
LU IPFT2S9L
GATEWAY END
/+
```

Figure 90. IPFNET LU List File for OC/TELNET FS

10.2.2.3 HOSTS File

This file is used to define the local and remote IP addresses and host names to OC/TELNET FS. With the specifications in this member, the VSE/ESA TELNET clients may use the host name to initiate a TELNET session to a TELNET server. The definitions in this file are based on the IP name and address assignments in Table 3 on page 36.

Figure 91 on page 139 lists the job we used.

```
CATALOG HOSTS.PARMLIB     EOD=/+          REPLACE=YES
#
# SAMPLE HOSTS FILE
#
192.61.100.81   AIX320
192.61.100.83   KWLOS2
/+
```

Figure 91. HOSTS File for OC/TELNET FS

10.2.2.4 SYSIN File

This file contains initialization commands for starting OC/TELNET FS. Please refer to *OpenConnect/TELNET Client Full Screen User and Administration Guide, 350-0145-101* for details of system initialization commands.

Figure 92 lists the job we used.

```
CATALOG SYSIN.PARMLIB     EOD=/+          REPLACE=YES
**********************************************************************
*  THE FOLLOWING SYSTEM INITIALIZATION COMMANDS ARE AUTOMATICALLY
*  EXECUTED WHEN THE PRODUCT IS STARTED.
**********************************************************************
*  SYSTEM INITIALIZATION COMMANDS FOR ALL OPERATING SYSTEMS:
**********************************************************************
*
  SET ACBNAME MAJOR VSTELNET
  SET ACBNAME MINOR VSNET
  SET ACBNAME START 1
  SET ACBNAME INQUIRE YES
*
  SET LUMAX 64
*
  SET USERDATA OPTIONAL
*
  SET USERMAX 64
*
  SET POOL SAS0020 START
  SET POOL SAS0020 COUNT 5
  SET POOL SAS0020 SIZE  20480
  SET POOL SAS0020 MODE  ANY
  SET POOL SAS0020 END
/+
```

Figure 92. SYSIN File for OC/TELNET FS

The key definitions and parameters are:

- ***SET ACBNAME MAJOR command.*** Defines the major ACB name used by OC/TELNET FS. The ACB name must match the ACB name of VSTELNET defined in the VTAM APPL major node.
- ***SET ACBNAME MINOR command.*** Defines the minor ACB name used by OC/TELNET FS. The value must match the name prefix of the VSNETnnn APPL entries defined in the VTAM APPL major node.

- ***SET ACBNAME START command.*** Determines the first available VSNETnnn application to invoke. A value of 1 causes VSNET001 as the first available application to connect to a VSE/ESA TELNET client session.
- ***SET ACBNAME INQUIRE YES command.*** Instructs OC/TELNET FS to inquire the active status of VSNETnnn, before allowing VSE/ESA TELNET clients to log on.
- ***SET LUMAX command.*** Defines the maximum number of LUs allowed in the NETWORK LU List file(s).
- ***SET USERDATA command.*** Determines whether a user command file must be provided, when the VSE/ESA TELNET clients log on to OC/TELNET FS.

  If SET USERDATA REQUIRED is specified, OC/TELNET FS terminates the logon session when a user command file can not be found. The following preparation tasks are required for this option:

  1. Create the user command file. Figure 93 lists the sample we used.
  2. Modify the OC/TELNET FS USSTAB specification to include the command file name as user data, for example:

     USSPARM PARM=P2,REP=DATA,DEFAULT=USER

  If SET USERDATA OPTIONAL is specified, the user command file is not required for the OC/TELNET FS logon. The VSE/ESA TELNET clients may choose to specify a user command file with the VTAM LOGON command. For example,

  LOGON APPLID(VSTELNET) DATA(USER)

  The first preparation task is required before this option can be used.

```
CATALOG USER.PARMLIB EOD=/+ REPLACE=YES
SET TERM DISC CONTINUE
OPEN TCPGTW1
TERMINAL
QUIT
/+
```

Figure 93. Sample User Command File

- ***SET USERMAX command.*** Determines the maximum number of VSE/ESA TELNET clients required.
- ***SET POOL command.*** Allow each installation to control the number and size of storage allocated to a storage pool in OC/TELNET client.
- ***SET POOL START command.*** This must be the first command in a storage pool definition for the pool named SAS0020.
- ***SET POOL COUNT command.*** Indicates the number of pool entries OC/TELNET client should build for the named pool.
- ***SET POOL SIZE command.*** Indicates the size for each pool entry.
- ***SET POOL MODE command.*** Defines if the storage pool should be above or below the 16MB line. Specification of ANY allows OC/TELNET client to build the pool either above or below the 16MB line.
- ***SET POOL END command.*** This command must be the last command in a storage pool definition.

Note

We removed the SETTIMER LIMIT definition from SYSIN.PARMLIB as we received the error message:

TA100I Set command parameter is invalid during startup

Set DBCS NO is supported by MVS only and removed.

10.2.2.5 DFLTSYS File

This file contains session control commands to determine how the OC/TELNET FS should be started. The commands in this file are executed during OC/TELNET FS initialization. This file may be used to set system default parameters to ease the operation of the VSE/ESA TELNET clients.

For details of OC/TELNET FS session commands, please refer to *OpenConnect/TELNET Client Full Screen User and Administration Guide, 350-0145-101*.

Figure 94 lists the job we used.

```
CATALOG DFLTSYS.PARMLIB   EOD=/+          REPLACE=YES
**********************************************************************
*  THE SESSION CONTROL COMMANDS CONTAINED IN THIS MEMBER ARE
*  AUTOMATICALLY EXECUTED ON BEHALF OF EACH USER WHEN THEY LOGON.
**********************************************************************
SET KEYBOARD TERMTYPE YES
/+
```

Figure 94. DFLTSYS File for OC/TELNET FS

The definition is taken from PRODBILD.PROC, without modification. The SET KEYBOARD TERMTYPE command informs OC/TELNET FS to automatically execute a terminal keyboard profile whenever the terminal is reset.

10.2.2.6 DFLTUSR File

This file contains session control commands to determine how OC/TELNET FS sessions should be started for all VSE/ESA TELNET clients.

This file is used to set up default session commands for the VSE/ESA TELNET clients. It may be used to automate the logon to a TELNET server. See Figure 95 for the sample we used.

```
CATALOG DFLTUSR.PARMLIB   EOD=/+          REPLACE=YES
&USERDATA
/+
```

Figure 95. DFLTUSR File for OC/TELNET FS

10.2.2.7 SYSCMDS File

This file contains operator commands to be executed during OC/TELNET FS initialization. The file is taken from the PRODBILD.PROC without modification.

For details of OC/TELNET FS operator commands, please refer to *OpenConnect/TELNET Client Full Screen User and Administration Guide, 350-0145-101*. See Figure 96 for the sample we used.

```
CATALOG SYSCMDS.PARMLIB   EOD=/+          REPLACE=YES
**********************************************************************
*  THE FOLLOWING SYSTEM OPERATOR COMMANDS ARE AUTOMATICALLY
*  EXECUTED WHEN THE PRODUCT IS STARTED:
**********************************************************************
SET QUERY COND
/+
```

Figure 96. SYSCMDS File for OC/TELNET FS

10.2.2.8 Terminal Keyboard Profiles

The terminal keyboard profiles are used to define the IBM 3278 keyboard mapping. The following profile definitions are available as profilename.parmlib members:

- ***D210***
- ***ESPRIT***
- ***EXPLORER***
- ***HP***
- ***IBM3161***
- ***LINE***
- ***MEAD***
- ***NATIVE***
- ***PAGE***
- ***SMIT***
- ***VAX***
- ***VIPROF***
- ***VT100***
- ***VT100A..E***
- ***VT200***
- ***VT52***

These terminal keyboard profiles may be used as samples to tailor terminal definitions not provided by OC/TELNET FS.

Please refer to *OpenConnect/TELNET Client Full Screen User and Administration Guide, 350-0145-101* for the terminal keyboard profile descriptions.

10.2.3 Startup Customization

This step is to tailor the sample OC/TELNET FS startup procedure from PRODBILD.PROC.

Figure 97 lists the job we created. The following changes are made:

- Add the necessary JCL to start the job in a dynamic partition.
- Specify the authorization code in the EXEC statement. The authorization code is obtained from OCS for license protection. It must be specified in the startup job.

```
* $$ JOB JNM=TNFS,CLASS=Y,DISP=L,NTFY=YES
// JOB TNFS   START OC/TELNET CLIENT FULL SCREEN
// LOG
// LIBDEF PHASE,SEARCH=(TCPOCS.TNFS,PRD2.CONFIG),TEMP
// LIBDEF SOURCE,SEARCH=(TCPOCS.TNFS,PRD2.CONFIG),TEMP
// EXEC PGM=TNFS,SIZE=TNFS,PARM=¢E9B591767631¢
/*
/&
* $$ EOJ
```

Figure 97. OC/TELNET FS Startup Job

Note: The OC/TELNET Software Authorization Code is specified as PARM=′ ′ on the EXEC statement of the OC/TELNET software startup procedure.

Chapter 11. OC/SAM Installation and Customization

This chapter covers the installation and customization of OC/SAM. It consists of two steps:

1. Installation - Describes the preparation and procedures to install OC/SAM in VSE/ESA
2. Customization - Provides an explanation of the product customization.

The installation instructions provided in this chapter assume you already have installed and are using the OCS II Gateway. OC/SAM is a program development toolkit that provides a socket application program interface on VSE/ESA systems to enable development of TCP/IP client/server applications.

11.1 OC/SAM Installation

There are three major steps involved to install the product:

1. Prepare the installation environment for the product
2. Restore the product from tape to the VSE/ESA sublibrary
3. Verify the installation

11.1.1 Installation Preparation

To prepare for installation of the OC/SAM the following definitions are required:

- An OCS II Gateway to VSE/ESA

  Please refer to Chapter 5, "Defining the OCS II Gateway to the Host" on page 41 for the setup instructions.

- The VSE/ESA library and sublibraries for the product installation

  The definition of library TCPOCS is discussed in 6.1, "OC/FTP Server Installation" on page 61.

  The Librarian command LIBR DEF SUB is used to define the required sublibraries in library TCPOCS:

  Sublibrary used for

  SAMOBJ SAM object decks required to be linked into the application

  SAMMAC SAM assembler macros

  SAMSMP SAM sample programs

11.1.2 Installation Procedures

Installation of the product from tape to the product sublibrary involves the following steps:

- To scan the tape before starting installation, use the following job, as illustrated in Figure 98 on page 146.

```
* $$ JOB JNM=LIBSCN,DISP=D,PRI=3,                                         C
* $$ NTFY=YES,                                                            C
* $$ LDEST=*,                                                             C
* $$ CLASS=0
// JOB LIBSCN  SCAN VSE LIBRARY BACKUP TAPE
* THIS FUNCTION USES A TAPE FOR INPUT
* MOUNT LABELED TAPE WITH VOLUME ID=SOCKET ON DEVICE 180
* THEN CONTINUE. IF NOT POSSIBLE CANCEL THIS JOB.
// PAUSE
// ASSGN SYS005,181
// TLBL TAPE1,¢SOCK.LIBS¢,,¢SOCKET¢
// EXEC LIBR,PARM=¢MSHP¢
   RESTORE *                          /* LIBRARY IDENTIFICATION */ -
          ID = *                      /* MNEMONIC-ID OF THE     */ -
                                      /*       (SUB-)LIB ON TAPE*/ -
          SCAN  = YES                 /* SCAN SPECIFICATION     */ -
          TL   = TAPE1                /* Pointer to TLBL statement*/ -
          TAPE = SYS005               /* TAPEADDRESS            */
/*
// MTC RUN,181
/&
* $$ EOJ
```

Figure 98. Sample Job to Scan Library Backup Tape

- Use the II (Interactive Interface) '*Restore VSE Library from Tape*' panel as illustrated below to generate the job to restore the three product libraries from tape, as shown in Figure 103 on page 148.
- Submit the job to restore OC/SAM into subject sublibraries.

First enter general restore parameters and the first sublibrary.

```
 SVR$LRS1                RESTORE VSE LIBRARY FROM TAPE

 Enter the required data and press ENTER.

LIBRARY NAME................ OCS____     Name of library to be restored.
                                         Use * for entire backup file.
                                         Press PF5 to finish the selection.
SUBLIBRARY NAME............. SOCKOBJ_    Enter name for sublibrary selection
MEMBER NAME................. ________    Enter name for member selection
MEMBER TYPE................. ________    Enter type for member selection
Does the backup file have an identification (mnemonic-id)?
Use * to restore the complete tape.
IDENTIFICATION.............. *_______         ________
NEWNAME..................... 1            1 to restore under a new name,
                                         else 2
TIME STAMP.................. 1            1 to keep the old date,
                                         else 2
LIST........................ 1            1 for a listing, else 2
OLD FORMAT.................. 2            1 for libraries in old form,
                                         else 2

 PF1=HELP      2=REDISPLAY  3=END                    5=PROCESS
```

Figure 99. Restore VSE Library from Tape

Then specify the new name to be used.

```
 SVR$LRS2                RESTORE VSE LIBRARY FROM TAPE

 Enter the required data and press ENTER.

Enter the new names you will use after RESTORE.
At least one of these names must differ from those on the previous panel.



LIBRARY NAME................. TCPOCS     Library name


SUBLIBRARY NAME.............. SAMOBJ      Sublibrary name







 PF1=HELP       2=REDISPLAY  3=END

 USE ONLY THOSE FIELDS WHICH HAVE ALREADY DATA IN IT.
```

Figure 100. New Name Specification

Repeat this for the other two sublibraries. Then enter the required data for the tape.

```
 SVR$TAPA                RESTORE VSE LIBRARY FROM TAPE

 Enter the required data and press ENTER.



TAPE ADDRESS................. 181         Address of the input tape (cuu)
                                          For valid addresses enter a †?†

VOLUME SERIAL NUMBER......... SOCKET      Volume serial number of the
                                          input tape.

LABEL PROCESS................ 1           Enter 1 if you want a labeled
                                          tape, otherwise enter 2.





 PF1=HELP       2=REDISPLAY  3=END
```

Figure 101. Tape Information

Enter the last part of the tape information.

```
 DSF$TAP1                    TAPE SPECIFICATION

 Enter the required data and press ENTER.



LABEL CHECK.................. 1           Enter 1 if you want label
                                         checking. Otherwise enter 2.



TAPE FILE ID...... SOCK.LIBS________      Unique name of the file on the
                                         volume.




 PF1=HELP       2=REDISPLAY  3=END
```

Figure 102. Tape Specification

This creates a job to restore the libraries from tape as shown in Figure 103.

```
* $$ JOB JNM=SAMRES,DISP=D,PRI=3,                                         C
* $$ NTFY=YES,                                                            C
* $$ LDEST=*,                                                             C
* $$ CLASS=0
// JOB SAMRES  RESTORE VSE LIBRARIES
*
* THIS FUNCTION USES A TAPE FOR INPUT
* MOUNT LABELED TAPE WITH VOLUME ID=SOCKET ON DEVICE 180
* THEN CONTINUE. IF NOT POSSIBLE CANCEL THIS JOB.
// PAUSE
// TLBL TAPE1,¢SOCK.LIBS¢,,¢SOCKET¢
// ASSGN SYS004,181
// EXEC LIBR,PARM=¢MSHP¢
   RESTORE OCS.SOCKOBJ.*.*:TCPOCS.SAMOBJ             -
                                /* MEMBER  IDENTIFICATION */ -
           OCS.SOCKMAC.*.*:TCPOCS.SAMMAC            -
           OCS.SOCKSMP.*.*:TCPOCS.SAMSMP            -
          ID = *                /* MNEMONIC-ID OF THE     */ -
                                /*      (SUB-)LIB ON TAPE*/ -
          LIST  = YES           /* LIST SPECIFICATION     */ -
          REPLACE = YES         /* REPLACE OPTION         */ -
          DATE = OLD            /* Indicator to keep timestamp */ -
          TL   = TAPE1          /* Pointer to TLBL statement   */ -
          TAPE = SYS004         /* TAPEADDRESS            */
/*
// MTC RUN,181
/&
* $$ EOJ
```

Figure 103. OC/SAM Library Restore Job

11.1.3 Installation Verification

11.1.3.1 Installation Verification Job

An installation verification job is provided with the sample material. The members used for installation verification in sublibrary TCPOCS.SAMSMP are:

Member	Description
IVP.A	Assembler source for verification program IVP
$VSEJCL.A	Job to assemble, link and run the test program

To make the necessary changes to these members copy them into your ICCF library, enter the ICCF command line, switch to the proper ICCF library and copy the members from the product library into the ICCF library, as shown below:

```
/SWITCH 96
LIBRP TCPOCS.SAMSMP IVP.A IVPVSE
LIBRP TCPOCS.SAMSMP $VSEJCL.A RUNIVP
```

Figure 104. Copy Librarian Member into ICCF Library

11.1.3.2 Modifications in IVPVSE

The verification program is delivered in an MVS version. It has to be adapted to the VSE environment by changing a few lines as shown below.

1. At label OPT01: replace ′MVS′ with ′VSE′
2. At label OPT51: replace the parameter library with your own, in our case ′CL44′TCPOCS.FTPC′′

Catalog the changed ICCF member IVPVSE back to the product library using the following LIBRC ICCF procedure:

```
LIBRC TCPOCS.SAMSMP IVP.A IVPVSE
REMOVE
```

Figure 105. Catalog ICCF Member into Sublibrary

11.1.3.3 Modifications in RUNIVP

To execute the verification program modify the RUNIVP member as shown below:

```
* $$ JOB JNM=RUNIVP,CLASS=0,DISP=D
* $$ LST DISP=D,CLASS=A,DEST=(,WSTIEBER)
// JOB RUNIVP  RUN VERIFICATION PROGRAM IVP
// LIBDEF SOURCE,SEARCH=TCPOCS.SAMMAC
// LIBDEF OBJ,SEARCH=TCPOCS.SAMOBJ
// LOG
// OPTION LINK
// EXEC PGM=ASMA90,SIZE=(ASMA90,50K),PARM=¢SYSPARM(VSE)¢
* $$ SLI MEM=IVP.A
/*
// EXEC PGM=LNKEDT
/*
// EXEC ,SIZE=AUTO
/*
/&
* $$ EOJ
```

Figure 106. Sample RUNIVP Job

11.1.3.4 Run the Verification Program

To assemble the source program we used the High Level Assembler. This sample program lets you generate the exit program ′EDECKXIT′.

Submit the RUNIVP job for execution. After the job finishes, check the listing to make sure that:

1. The assembler completed without errors.
2. The linkage editor included all required object decks into the phase.
3. All trace events ended with return code 0, except for ′GET SERVICE BY NAME′ for the service ′UNKNOWN′.
4. The job ends with return code 4.

Now OC/SAM has been correctly installed and the sockets interface is usable for your application programs.

11.2 OC/SAM Customization

No particular customization steps are required before OC/SAM can be used.

An optional customization step for OC/SAM is:

- ′GET SERVICE BY NAME′ table customization

11.2.1 GET SERVICE BY NAME Table Customization

The ′GET SERVICE BY NAME′ table contains the names of common service systems, such as FTP server, domain name servers and so on. If you want to define your own service system you have to add an entry to the ′GET SERVICE BY NAME′ table. You do this by appending an entry to the source file TABLE001.A in sample sublibrary TCPOCS.SAMSMP. Then a new version of

TABLE001.OBJ has to be assembled and cataloged into the object sublibrary TCPOCS.SAMOBJ.

These are the steps to add an entry to the 'GET SERVICE BY NAME' table:

1. Copy TABLE001.A source file from TCPOCS.SAMSMP into ICCF library member TABLE001
2. Add new entries to ICCF member TABLE001
3. Catalog the member you changed
4. Run the assembly and use the $VSETABJ job to catalog the updated table into the TCPOCS.SAMOBJ sublibrary

11.2.1.1 Updating TABLE001

If you want to define a new entry in the 'GET SERVICE BY NAME' table proceed as follows:

Parameters Description

MYSERVICE Is the name of a service system running somewhere in the network.

50 Is the TCP/IP port number that MYSERVICE is using.

UDP Is the TCP/IP transport protocol used by MYSERVICE.

MINE Is an alias name that can also be used to locate MYSERVICE in the 'GET SERVICE BY NAME' table.

The resulting entry in the TABLE001 member appears as follows:

```
         DC    CL20¢MYSERVICE¢         Name of service
         DC    H¢50¢                   Port number service is using
         DC    CL4¢UDP¢                Protocol service is using
         DC    CL20¢MINE¢              Alias name for service
```

Figure 107. Add Entry to 'GET SERVICE BY NAME' Table

11.2.1.2 Assemble and Catalog the GET SERVICE BY NAME Table

Copy $VSETABJ into ICCF member VSETABJ. Modify the job according to your requirements as shown in the example below:

- Provide your VM user ID in the DEST parameter of the $$ LST statement ('WSTIEBER' in the example).
- Change the 'ACCESS SUBLIB' value to the sublibrary containing the object deck (we used TCPOCS.SAMOBJ).
- Insert an End-of-Data marker ('/*') behind the input cards of the first invocation of the Assembler.

Now the job to generate the new 'GET SERVICE BY NAME' table can be submitted. Figure 108 on page 152 lists the job we used.

```
* $$ JOB JNM=TABLE001,CLASS=0,DISP=D
* $$ LST DISP=D,CLASS=A,DEST=(,WSTIEBER)
* $$ PUN DISP=I,CLASS=0,PRI=9
// JOB TABLE001  CUSTOMIZE THE GET SERVICE BY NAME TABLE
// LIBDEF SOURCE,SEARCH=TCPOCS.SAMSMP
// LIBDEF PHASE,SEARCH=PRD2.PROD
// LOG
// OPTION DECK,NOXREF
// EXEC PGM=ASMA90,SIZE=ASMA90
   PUNCH ¢* $$ LST DISP=D,CLASS=A,DEST=(,WSTIEBER)¢
   PUNCH ¢* $$ PUN DISP=I,CLASS=0,PRI=9¢
   PUNCH ¢// JOB TABLE001¢
   PUNCH ¢// LOG¢
   PUNCH ¢// EXEC PGM=LIBR¢
   PUNCH ¢   ACCESS SUBLIB=TCPOCS.SAMOBJ¢
   PUNCH ¢   CATALOG TABLE001.OBJ REPLACE=YES¢
   END
/*
// EXEC PGM=ASMA90,SIZE=ASMA90,PARM=¢SYSPARM(VSE)¢
* $$ SLI MEM=TABLE001.A
/*
// EXEC PGM=ASMA90,SIZE=ASMA90
   PUNCH ¢/*¢
   PUNCH ¢/&&¢
   END
/*
/&
* $$ EOJ
```

Figure 108. Sample VSETABJ Job

Chapter 12. OCS II Gateway Installation and Customization

This chapter covers the OCS II Gateway part of OpenConnect Systems running on a RISC/6000 and provides:

- a functional overview of the OCS II Gateway
- detailed instructions on how to install the OCS II Gateway
- detailed instructions on how to customize the OCS II Gateway
- special consideration on implementing OC/TELNET Server
- set up alternative port number for AIX TELNET, FTP Servers

12.1 OCS II Gateway Overview

The OCS II Gateway provides gateway and protocol conversion functions as described in 3.1, "OCS II Gateway Functional Overview" on page 16. In this way a RISC/6000 workstation acts as a gateway between TCP/IP and SNA networks.

The OCS II Gateway software resides on a RISC/6000 and communicates to the VSE host (″upstream″) via SNA protocols, while it connects to TCP/IP nodes (″downstream″) using standard TCP/IP protocols.

As shown in Figure 2 on page 6, the OCS II Gateway part of OpenConnect Systems consists of:

- ***the base*** (called ″OCS II″), which provides basic gateway functions such as mapping IP addresses to/from SNA LUs, conversion and translation of IP datagrams to/from SNA PIUs and so on.
- ***the OC/TELNET Server*** (optional) which allows TCP/IP nodes to log on to the VSE host thereby emulating a 3270 type terminal.

The other components of OpenConnect Systems (OpenConnect/TELNET Client Full Screen and OpenConnect/File Transfer Program) reside in the VSE host. Refer to Chapter 3, "OCS Software Functional Overview" on page 15 for a description of these components.

12.2 Preinstallation Tasks

Chapter 2 of the *OpenConnect Systems OpenConnect Server II Installations and Operations Guide RISC/6000, 350-0285-101.15* manual describes the pre-installation tasks in detail. Please refer to this document which is delivered with the product diskette. This document only wants to point out that, before you begin the installation and customization process, you must understand your network configuration. The OCS II product offers a number of services; (see Figure 2 on page 6); you may not require all of them, but you will have to identify them in order to understand the hardware and software requirements. In our installation the following OCS functions were defined and tested:

- SNA gateway server (PU 2.0, LU 2 and Token-Ring Network)
- 3270 Telnet Server (used by TCP/IP clients on PS/2, SUN and RISC/6000)

12.3 Installation

This is a very simple step: It basically consists of installing the base product diskette. Please refer to Chapter 3 of the *OpenConnect Systems OpenConnect Server II Installations and Operations Guide RISC/6000, 350-0285-101.15* that describes the installation task in detail. The installation command sequence is the following:

1. **mkdir /usr/tmp/oc**
2. **cd /usr/tmp/oc**
3. **tar xvf /dev/***rdevname*
   where *rdevname* = I/O device name (for example, rfd0 for the diskette drive)
4. **./ocinstall**, to start the installation job

Figure 109 shows the menu displayed when the installation exec (ocinstall) is invoked.

```
                  OpenConnect Systems Incorporated
                OCS II Gateway Installation Procedure

                1)  INSTALL or UPGRADE the OCS II Gateway Software
                2)  VERIFY Checksums
                3)  REMOVE the OCS II Gateway Software
                4)  Get HELP for this Installation Procedure
                5)  Sun-Specific Installation Procedure
                6)  EXIT


                Copyright 1992.  OpenConnect Systems Incorporated.
                   2711 LBJ Freeway, Dallas, TX  75234
                   Technical Assistance:  (214) 888-0678
                   Sales and Information: (214) 484-5200

Enter selection number:
_
```

Figure 109. Installation Exec Main Menu

By choosing selection "1" the OCS II product code is loaded into the OCS directories. The contents of these directories after installation are shown in Figure 110 on page 155. This figure shows the OCC directories in addition to the OSC directories.

```
root@aix320: /usr/oc >ls -Rx /usr/oc
INSTALL                    INSTALL.OCC
OCC.OCILOG                 OCC.OCILOGP
OCC.OCINFO                 OCILOG
OCILOGP                    OCINFO
OCMSG                      OCMSG.OCC
OCSUM                      OCSUM.OCC
READ.ME                    README.OCC
bin                        config
config.list                core
eps                        occ
occinstall                 ocinstall
scr                        sna
tn3179g
/usr/oc/bin:
 ..cone9164935  conf928.text    fd0             libSEC.sl       occonfig
ocsna           peek            scaad           scacl           sna_start
sna_status      status
/usr/oc/config:
3030menu  boxmenu   cfdelgto  chnconf   cnos      comitmnu  configms  dddlu
delconf   deletepu  delgto    devconf   devconfg  devmem    devmenu   dfltmnu
dlcconf   dlcdelcf  dlcdelet  dlcmenu   dlcpunb   dlctype   dlinkall  errexit
exitconf  filmenu   filmenux  gensys    help      hpbase    hpconf    hpdelcf
hpdelet   hpmenu    hppunb    hpsvcmnu  hptokn    hpx25     hpx25pvc  hpx25svc
hpxadv    hpxbasic  hpxnetwk  instconf  instdel   instdel1  instdel2  instexit
instld1   instld2   instld3   instld4   instld5   instldc1  instnew1  instnew2
instprt1  instprt2  instsv1   instsv2   instsv3   ipbase    ipinfo    ipnets
iprcnfrm  iprdacf   iprdel    iprdscf   iprout    iprout1   keepaliv  loadm2
loadmenu  luconf    luinfo    luinfo1   luinfo2   luinfo3   lumenu    luusage
mainmenu  mrssinfo  necsdlc   ocs2menu  printmnu  puconfg   rs6base   rs6bsdlc
rs6btokn  rs6bx25   rs6conf   rs6delcf  rs6delet  rs6menu   rs6punb   rs6sdlc
rs6tokn   rs6x25    rtminfo   savdflt   savebox   savefil   savefilx  savem2
scabase   selchan   seldev    seldevg   snacnfrm  snaconfg  snamenu   snmpcom1
snmpcom2  snmpcom3  snmpinfo  soltokn   sunbase   sunconf   sundelcf  sundelet
sunmenu   sunpunb   sunsdlc   sysmenu   tcpinfo   tcpmenu   telinfo   telsec
tn3270    tn5250    tntel     toknmenu  ts3270    ts5250    x25adv    x25basic
x25netwk  x25pcomn  x25scomn  x25svc
/usr/oc/config/help:
3030menu.hlp ...    x25svc.help  (Many help files but omitted)
/usr/oc/eps:
01             config         config.save    defaults       defaults.save
eps            eps_start      eps_status     eps_stop       ocs_elmadmin
ocs_elmalert   ocs_elmd       ocs_elmrpt     ocs_elmusage   ocs_elmver
printcap       read.me        tracutil
/usr/oc/occ:
OCINFO            core              lfw_util.s2       lsca.s2
sca.ini           sca.ini.org                                           G
/usr/oc/scr:
pid      starteps  startsca  starttok  status
/usr/oc/sna:
OCSNA.LOG.8090       core                 dlcfg
dlcfg.occ1           dlcfg.org            dlcfg.syn
dlcfg.vm_via3174     etc                  instance
ocsvse               parse                parseout
pid                  sna.cfgsna.bin       sna_bin.occ1
sna_bin.vm_via3174   sna_cfg              start
status               tmp                  tn3179gvse
tn3270vse            tnf                  trace
tsvse                tvse
```

Figure 110 (Part 1 of 2). OCS and OCC Directories Contents

```
/usr/oc/sna/etc:
cfgtext            ockeys             octermcap          options     .cfg
options.occ1       options.vm_via3174  tstext            tstextp     p
/usr/oc/sna/tmp:
Ex08290            Rx08290
/usr/oc/sna/trace:
/usr/oc/tn3179g:
api      cpsna    lbin     man      read.me
/usr/oc/tn3179g/api:
Oc3270           Xapi3179g.a      Xapi3270.a       api.h
Oc3270           Xapi3179g.a      Xapi3270.a       api.h
api3270.a        apidemo.c        compile_apidemo  implem.h
lu2io.h          parmfile         scdefs.h         scv.h
thlulu.h
/usr/oc/tn3179g/cpsna:
printcap        printers        snatermkey      snatermkey.org  snatext
/usr/oc/tn3179g/lbin:
sna3270     tn3179g     tn3270      vfytermkey
/usr/oc/tn3179g/man:
man1  man4
/usr/oc/tn3179g/man/man1:
sna3270.1     tn3179g.1     tn3270.1      vfytermkey.1
/usr/oc/tn3179g/man/man4:
snatermkey.4
```

Figure 110 (Part 2 of 2). OCS and OCC Directories Contents

12.4 Customization

The customization of the OCS II Gateway is a more complex part. Various configuration files have to be modified according to your specific requirements. There are two different ways to do this:

- Using the Configurator program ('occonfig').
- Manually editing the various files.

It's strongly recommended to use the first method, since this is the easiest way to configure the OpenConnect Server II Gateway. It offers the advantage of a menu driven utility where you are asked to enter the values reflecting your real network configuration and your particular needs. The program is able to do some checking of the values entered and to signal the possible errors.

Starting with the configurator for Version 3.6 of the OCS II Gateway, the configuration process is designed to configure instances. An instance is one iteration of the OCSNA process running its specific configuration files. For more information on the **ocsna** command, please see 13.1, "Operation Samples (start,stop,status....)" on page 189. Each instance has its own set of configuration files associated with it. Unique instances are maintained through the use of file name suffixes.

The instance file is created and maintained via the configurator. The contents of the file consist of the names of one or more instances and the associated file name suffix. A sample entry for an instance file is shown in Figure 111 on page 157.

```
INSTANCE_ENTRY:
NAME = occ1
SUFFIX = 01
INSTANCE_ENTRY_END:
```

Figure 111. Sample Entry for an Instance File

The maximum number of OCSNA instances which can be configured is 16. If an instance name is present with no suffix, the configurator loads and writes the default configurator files. Both the instance name and the suffix must be unique for each OCSNA instance. The default file name is **/usr/oc/sna/instance**.

The different panels described in Chapter 4 of the *OpenConnect Systems OpenConnect Server II Installations and Operations Guide RISC/6000, 350-0285-101.15* are sometimes difficult to understand regarding the exact meaning of the values requested. The following paragraphs describe the OCS II Gateway customization using the configuration program ′occonfig′; the ″manual″ method will not be described.

The configuration exec is called ***occonfig*** and, if you accept the installation defaults, resides in the directory */usr/oc/bin*. To invoke it you have to enter the command **occonfig** (or ./occonfig if you have not made appropriate path definitions in your ′.profile′).

Figure 112 on page 158 shows the Main menu that appears when the configuration exec is invoked. From this Main menu, you can select **Select OCSNA Instance to Configure** to update an existing configuration, or **Load OCSNA Instance** to create a new configuration.

Please see the *OCS II Gateway Installation and Operations Guide, 350-0285-101* for detailed information. Figure 113 on page 158 shows the screen after we select: **Select OCSNA Instance to Configure**.

```
OCS occonfig                    Main Menu


Select OCSNA Instance to Configure
Delete OCSNA Instance
Generate OCSNA Instance Listing
Load OCSNA Instance
Save OCSNA Instance















ESC+1=Help      ESC+2=ExitPgm   ESC+3=PrevMenu        OCS II RISC/6000 V3.7.5
ESC+4=ConfMenu  ESC+5=MainMenu  Return=AcceptMods               Instance=occ1
```

Figure 112. Configuration Main Menu

```
OCS occonfig               Configure OCSNA Instance

  Enter an Instance Number to Configure...............  2

  Instance        Instance           Instance        Instance
  Number          Name               Number          Name

   1              occ1                2              vm_via3174
   3                                  4
   5                                  6
   7                                  8
   9                                  10
   11                                 12
   13                                 14
   15                                 16





 Numeric values ONLY allowed in this field.
ESC+1=Help      ESC+2=ExitPgm   ESC+3=PrevMenu        OCS II RISC/6000 V3.7.5
ESC+4=ConfMenu  ESC+5=MainMenu  Return=AcceptMods               Instance=occ1
```

Figure 113. Configure OCSNA Instance Menu

Warning

During the configuration process, after having completed a screen and entered your appropriate values, press the ***ENTER*** key. If the values are correct you will receive the message: **Input accepted**. The parameters entered will become effective as soon as you save the configuration update (see following steps).

```
  OCS occonfig                Instance Configuration Menu


  Local Area Network Configuration
  SNA Level Data Link Configuration
  Physical/Logical Unit Configuration
  System Wide Configuration Options
  Modify/View Data Link Configuration















ESC+1=Help      ESC+2=ExitPgm   ESC+3=PrevMenu        OCS II RISC/6000 V3.7.5
ESC+4=ConfMenu  ESC+5=MainMenu  Return=AcceptMods         Instance=vm_via3174
```

Figure 114. Instance Configuration Menu

The individual configuration steps shown in the configuration menu will be discussed in the following sections.

12.4.1 Local Area Network (TCP/IP) Configuration

```
 OCS occonfig               TCP/IP Configuration


 Specify Internet Addresses
 Specify TCP/IP Keepalive Options
















ESC+1=Help      ESC+2=ExitPgm   ESC+3=PrevMenu        OCS II RISC/6000 V3.7.5
ESC+4=ConfMenu  ESC+5=MainMenu  Return=AcceptMods          Instance=vm_via3174
```

Figure 115. LAN Configuration Menu

The LAN configuration is a required step to correctly attach the OpenConnect Server II Gateway to the TCP/IP network. Several parameters entered in these screens (Figure 116 on page 161 and Figure 118 on page 162) have to match the ones specified in the AIX TCP/IP setup. The customization setup and parameter values shown in the following sections are taken from the actual installation at the ITSO Center Boeblingen. They reflect and correspond to the network diagram shown in Figure 13 on page 34.

```
OCS occonfig             Specify Internet Addresses


Internet address of the OCS II Server.......... 192.61.100.81




NOTE - Addresses may be entered either in decimal notation or in
       hexadecimal. If entering hexadecimal addresses, precede them
       with OX.









ESC+1=Help      ESC+2=ExitPgm   ESC+3=PrevMenu        OCS II RISC/6000 V3.7.5
ESC+4=ConfMenu  ESC+5=MainMenu  Return=AcceptMods          Instance=vm_via3174
```

Figure 116. Internet Address Configuration Menu

- ***Internet address of the OpenConnect unit.*** This field represents the address assigned to your OCS II Gateway by the network administrator: It **must match** the Internet Address specified in ′TCP/IP Minimum Configuration and Startup′ panel of the AIX System Management Interface Tool (SMIT) facility (Figure 117 on page 162). The **ALTERNATE TOKEN RING address** in SMIT′s DEVICES part of ′System Management′ has to match the **MACADDR** parameter in the VTAM **PU** statement of the switched major node definition (see Figure 31 on page 55).

  In order to find the SMIT TCP/IP Configuration Panel, you can specify **SMIT** at the AIX System prompt, this will give you the System Management Panel, here you have to select: **Communications Applications and Services**. On the next panel you have to specify **TCP/IP**, and on the following panel, specify: **Minimum Configuration & Startup**. On the next panel, please specify: **Standard Ethernet Network Interface**, and you should now be able to see the SMIT TCP/IP Configuration Panel, and to check the Internet Address.

- ***Enable Keepalive Option.*** This option allows you to turn on or off the ′Keepalive′ option. The default setting is ′N′ (off). This allows for sending Keepalive packets when there was no activity on the connection for the Keepalive interval set by AIX. For more information about the Keepalive interval, refer to the appropriate AIX documentation.

```
                      Minimum Configuration & Startup

 To Delete existing configuration data, please use Further Configuration menus


Type or select values in entry fields.
Press Enter AFTER making all desired changes.

                                                          Entry Fields
* HOSTNAME                                             aix320
* Internet ADDRESS (dotted decimal)                    192.61.100.81
  Network MASK (dotted decimal)                        255.255.255.0
* Network INTERFACE                                    tr0
  NAMESERVER
           Internet ADDRESS (dotted decimal)           192.61.100.81
           DOMAIN Name                                 itsc.ibm.com
  Default GATEWAY Address                              192.61.200.2
  (dotted decimal or symbolic name)
  RING Speed                                           16                      +
  START Now                                            no                      +

F1=Help             F2=Refresh          F3=Cancel           F4=List
Esc+5=Reset         F6=Command          F7=Edit             F8=Image
F9=Shell            F10=Exit            Enter=Do
```

Figure 117. SMIT TCP/IP Configuration Panel

```
  OCS occonfig          Specify TCP/IP Keepalive Options


  Keepalive Option enabled(Y/N)...... Y   N=Keepalive disabled (Default)
                                          Y=Keepalive enabled














ESC+1=Help      ESC+2=ExitPgm   ESC+3=PrevMenu        OCS II RISC/6000 V3.7.5
ESC+4=ConfMenu  ESC+5=MainMenu  Return=AcceptMods         Instance=vm_via3174
```

Figure 118. TCP/IP Keepalive Option Menu

12.4.2 SNA Level Data Link Configuration

Figure 119 is the first of a sequence of panels describing the SNA interface to the host, in our case an IBM 9221-150 running VSE/ESA. In order to provide the appropriate values, close cooperation with your SNA network administrator is required.

```
  OCS occonfig              Data Link Values

  The values entered here will be in effect for ALL Physical Units defined.

  Datalink buffer size 2057      MUST match the MAXDATA parm in the NCP gen,
                                 or the MAXFRAME parm on the controller
                                 definition.

                                 NOTE - default value is 265.

  Datalink encoding    NRZ       Specify this parameter as either NRZ or NRZI.
                                 This parameter MUST match the corresponding
                                 entry in the host definition.

                                 NOTE - default value is NRZ.




ESC+1=Help      ESC+2=ExitPgm   ESC+3=PrevMenu        OCS II RISC/6000 V3.7.5
ESC+4=ConfMenu  ESC+5=MainMenu  Return=AcceptMods          Instance=vm_via3174
```

Figure 119. SNA DLC Configuration Menu

The fields appearing in the DLC configuration screen describe the attributes common to all PUs (Physical Units) defined in the next section.

- ***Datalink buffer size.*** This field specifies the maximum number of bytes that your physical units can receive in one PIU, including the transmission header (TH = 6 bytes) and the request/response header (RH = 3 bytes). It **must match** the value specified in the **MAXDATA** parameter of the **PU** VTAM or NCP statement (see Figure 31 on page 55).
- ***Datalink encoding.*** This field describes the line encoding technique used on an SDLC link and **must match** the **NRZI** parameter of the **LINE** NCP statement. Although this entry isn′t applicable to a Token-Ring line connection it′s a required parameter and you have to specify it: The field will be ignored.

12.4.3 Physical/Logical Unit Configuration

These panels allow you to configure the characteristics of all workstations/terminals requiring access to VSE/ESA and vice versa via the OCS II Gateway controller. The number of configurable workstations/terminals depends on the OpenConnect Server II Gateway feature you have installed on your system (the base minimum feature contains 16 LUs).

```
  OCS occonfig               PU/LU Configuration


  Work with Physical/Logical Unit Configuration
  Delete a Physical Unit, all its Logical Units and Datalink Entry
  Display Logical Unit Usage
















ESC+1=Help      ESC+2=ExitPgm   ESC+3=PrevMenu        OCS II RISC/6000 V3.7.5
ESC+4=ConfMenu  ESC+5=MainMenu  Return=AcceptMods         Instance=vm_via3174
```

Figure 120. PU/LU Configuration Menu

With the first option on this menu you are asked to introduce the relative Physical Unit number. This number is used to create an ordered list of the PUs defined and if it is your first PU definition, **1** (the default) has to be entered, since a discontinuous numeration isn't allowed.

```
 OCS occonfig               Physical Unit Definition



 PU name............ IPFP2209  Will be used as the LOCAL LOCATION NAME for
                               Midrange (S/36, S/38, AS/400) installations.
 Data Link Name..... IPFP2209  Data Link Name is required.
 Station Address.... 04        MUST match the host definition.
 Termination Rule... T         L = LUSTAT, T = TERM-SELF.
 PU type............ PU2.0     Specify as either PU2.0 or PU2.1.
 IDBLK value........ 017       Defines capabilities the PU provides.
                               †000† or †   † will default to 0x017 for
                               PU2.0 or 0x03E for PU2.1.
 IDNUM value........ E0009     MUST match the IDNUM parm in host definition,
                               if specified.
 PU connected to a   N         Enter Y if this PU is to be DIRECTLY CONNECTED
 Midrange system?              to a MIDRANGE (S/36, S/38, AS/400) system, or
                               N if this PU is to be DIRECTLY CONNECTED to a
                               MAINFRAME class system.


ESC+1=Help      ESC+2=ExitPgm   ESC+3=PrevMenu        OCS II RISC/6000 V3.7.5
ESC+4=ConfMenu  ESC+5=MainMenu  Return=AcceptMods         Instance=vm_via3174
```

Figure 121. Physical Unit Configuration

- ***PU name***. It describes the network node name assigned to this PU and **must match** the label name of the **PU** statement of your VTAM or NCP definition (see Figure 24 on page 47 and Figure 31 on page 55).
- ***Data Link Name***.
- ***Station address***. This field contains the two digit number assigned to this PU. For an SDLC link it **must match** the **ADDR** parameter value of the **PU** VTAM/NCP statement. For a Token-Ring connection it **must match** the Service Access Point (SAP) of OCS II Gateway station (for a RISC/6000 it′s always **4**).
- ***Termination Rule***.The recommended setting for S/370 is ′T′ which makes sure that VTAM is notified at session termination. For 3274 controllers, the recommended setting is ′L′ which lets the application control the session termination.
- ***PU type***. The only two values accepted in this field are PU2.0 and PU2.1. The PU2.0 identifies a standard SNA cluster (for example, 3274) working in the classic SNA hierarchical mode. The PU2.1 defines the new type of cluster controllers (for example, 3174) or midrange systems (for example, AS/400) supporting the APPN (Advanced Peer-to-Peer Networking) functions. For more details on this please refer to *VTAM Network Implementation Guide, SC31-6434*.
- ***IDNUM value***. This field is a user value and **must match** the **IDNUM** parameter of the **PU** statement in the VTAM Switched major node definition. If you have configured the Gateway station as a local major node, you can insert any mnemonic value (for example, PU001).

- ***Midrange connection***. If your OCS II Gateway station is connected to a midrange system (for example AS/400) specify **Y**. In our case (mainframe connection) we specified **N**.

After pressing ENTER the next screen (Figure 122) appears.

Select **Display all Logical Units** and the next screen (Figure 123 on page 167) appears.

```
  OCS occonfig                Logical Unit Menu


  Configure Logical Units
  Display All Logical Units
  Configure Datalink













ESC+1=Help      ESC+2=ExitPgm   ESC+3=PrevMenu         OCS II RISC/6000 V3.7.5
ESC+4=ConfMenu  ESC+5=MainMenu  Return=AcceptMods          Instance=vm_via3174
```

Figure 122. Logical Unit Configuration

```
 OCS occonfig              Display All Logical Units
To configure LU¢s, enter the LU name and LU type. See Help for LU types.

    LU Name  Type       LU Name  Type       LU Name  Type       LU Name  Type
 1) IPFT2S9A   14    2) IPFT2S9B   14    3) IPFT2S9C   14    4) IPFT2S9D   14
 5) IPFT2S9E   14    6) IPFT2S9F   14    7) IPFT2S9G   14    8) IPFT2S9H   14
 9) IPFT2S9I   14   10) IPFT2S9J   14   11) IPFT2S9K   14   12) IPFT2S9L   14
13) IPFT2S9M   10   14) IPFT2S9N   1    15) IPFT2S9O   1    16) IPFT2S9P   14
17)                 18)                 19)                 20)
21)                 22)                 23)                 24)
25)                 26)                 27)                 28)
29)                 30)                 31)                 32)
33)                 34)                 35)                 36)
37)                 38)                 39)                 40)
41)                 42)                 43)                 44)
45)                 46)                 47)                 48)
49)                 50)                 51)                 52)
53)                 54)                 55)                 56)
57)                 58)                 59)                 60)
61)                 62)                 63)                 64)

ESC+1=Help      ESC+2=ExitPgm   ESC+3=PrevMenu        OCS II RISC/6000 V3.7.5
ESC+4=ConfMenu  ESC+5=MainMenu  Return=AcceptMods          Instance=vm_via3174
```

Figure 123. Logical Unit Configuration

In this configuration panel you are asked to introduce the values describing the LU characteristics owned by the PU you just configured. You can define up to 64 LUs for each PU.

- ***LU name***. This field contains the symbolic name representing this LU in your network. The LU name and its corresponding number **must match**, in the same order, the label **name** and the **LOCADDR** parameter in the VTAM/NCP **LU** statement (see A.3, "VSE TCPSW.B Switched Major Node" on page 342).
- ***LU type***. This field requires a number whose value is shown in the upper side of this screen. When specifying the number consider how the workstation assigned to it will be used. Specify:
  - ″**14**″ for LUs interfacing the OCS II products:
    - OpenConnect/File Transfer Program Client
    - OpenConnect/Remote Shell (remote shell)
    - OpenConnect/File Transfer Program Server
    - OpenConnect/TELNET Client Full Screen
    - OpenConnect/Line Printer Daemon
    - and also for Socket Access Method Application programs
  - ″**1, 5, 6, 7, 8**″ for LUs assigned to the OCS II Telnet Server function of the OCS II Gateway
  - ″**10**″ for LU assigned to a terminal printer LU 3 (3287 LU 3).
  - ″**15, 16**″ for APPC (not used in our installation).

12.4.4 System Wide Configuration Options

In this menu you have the chance to customize or enable some services of the OCS II Gateway Software. This is an optional step; if you don't perform it the system uses the default values.

The menu shown in Figure 124 provides for the customization of several services of the OCS II Gateway. The relevant services for our environment are:

- Miscellaneous System Wide Parameters
- TN and TELNET Server Configuration
- PU T2.1 Node Configuration

```
  OCS occonfig                System Information


  Miscellaneous System Wide Parameters
  Dynamic Definition of Dependent LU (DDDLU) Configuration
  Response Time Monitor (RTM) Configuration
  TN and Telnet Server Configuration
  PU T2.1 Node Configuration
  Midrange FTP Server Support (MRSS) Configuration













ESC+1=Help      ESC+2=ExitPgm   ESC+3=PrevMenu        OCS II RISC/6000 V3.7.5
ESC+4=ConfMenu  ESC+5=MainMenu  Return=AcceptMods         Instance=vm_via3174
```

Figure 124. System Wide Options Configuration Menu

When you select **Miscellaneous System Wide Parameters** you will see the following screen:

```
OCS occonfig          Miscellaneous System Wide Parameters

Instance Administration Port Number.......(Default is 2000). 2000

Suppress BASIC ALERTS...................................... N
Keyboard reset on program check............................ Y
Assign only an active dependent LU......................... N

AYT interval for 3270 sessions and APPC applications........ 100
  Valid values are from 100 to 300 seconds.  A value of 0
  indicates that NO †Are you there† messages will be sent.









ESC+1=Help      ESC+2=ExitPgm   ESC+3=PrevMenu        OCS II RISC/6000 V3.7.5
ESC+4=ConfMenu  ESC+5=MainMenu  Return=AcceptMods         Instance=vm_via3174
```

Figure 125. Miscellaneous System Wide Parameters

- ***Instance Administration Port Number,*** defines the TCP port to enable connections with OCS admin utilities. The default is 2000.
- ***Suppress BASIC ALERTS*** option controls sending alerts to IBM′s NETVIEW. The default value is N.
- ***Keyboard reset on program check,*** option controls the keyboard operation when a negative response is received containing an end bracket. The default value is Y.
- ***Assign only an active dependent LU*** option controls the requests to obtain an LU when there is no SSCP-LU session active. The default value is N.
- ″***Are you there***″ polling interval for 3270 sessions and APPC applications can range from 100 to 300 seconds. If no polling is desired, set value to 0.

```
OCS occonfig            TN and Telnet Server Configuration


Telnet Server Configuration
TN 3270 Configuration
TN 5250 Configuration
Telnet 3270 Configuration















ESC+1=Help      ESC+2=ExitPgm   ESC+3=PrevMenu        OCS II RISC/6000 V3.7.5
ESC+4=ConfMenu  ESC+5=MainMenu  Return=AcceptMods         Instance=vm_via3174
```

Figure 126. TN and TELNET Server Configuration

In our installation we changed the default values from

- Telnet Server Configuration and
- TN 3270 Configuration

```
OCS occonfig           TELNET Server Configuration


Do you want Telnet Server to use negotiated TERM types?.... N
If desired, enter an alternate port number for Telnet Srvr  23
Enter size (1-128KB) of the Termcap RAMDISK(64KB default).. 64
Enter the number of TELNET Redirector sessions............ 16
Response time interval in seconds for IP Address
   Health Check feature (0 = disable)..................... 0
Response time interval in seconds for Requested LU
   Health Check feature (0 = disable)..................... 0
Queue session activation requests until Health Check
   completes? (Requested LU Health Check is required)...... N
Do you want to log sessions terminated by Health Check?.... N







ESC+1=Help      ESC+2=ExitPgm   ESC+3=PrevMenu        OCS II RISC/6000 V3.7.5
ESC+4=ConfMenu  ESC+5=MainMenu  Return=AcceptMods         Instance=vm_via3174
```

Figure 127. TELNET Server Configuration

- ***Term types negotiation.*** You can enable the OCS II Telnet Server to negotiate, with the remote host, the terminal type.
- ***Alternate PORT number.*** This field is a four digit number defining the port number reserved for Telnet sessions with the VSE/ESA host. It must be different from the AIX TELNET Server port number. Refer to 12.6, "Set up Alternative Ports for AIX TELNET and FTP Servers" on page 186 for assigning an alternate port for AIX TELNET Server.
- ***Termcap file size.*** This field is the maximum size, in Kbytes, of the 'octermcap' file that contains the display station's definition (please refer to *OCS II Telnet Server Manager, 350-0193-101* for details).
- ***TELNET Redirector Sessions.*** Defaults to the current configuration in the option file. If not currently defined in the Option File, it will default to the number of LUs purchased. Disable = 0.

  NOTE - TELNET Redirector Option may not be available on all platforms.

```
OCS occonfig           TN 3270 Server Configuration

Enter the number of TN3270 sessions......................... 8
Enable TN3270E support?..................................... N

TN3270E Functions

Enable SCS-CTL-CODES function support?...................... Y
Enable DATA-STREAM-CTL function support?.................... Y
Enable RESPONSES function support?.......................... Y
Enable BIND-IMAGE function support?......................... Y
Enable SYSREQ function support?............................. Y








ESC+1=Help      ESC+2=ExitPgm   ESC+3=PrevMenu        OCS II RISC/6000 V3.7.5
ESC+4=ConfMenu  ESC+5=MainMenu  Return=AcceptMods         Instance=vm_via3174
```

Figure 128. TN 3270 Server Configuration

- ***Number of TN3270 sessions***. Determines the number of concurrent Telnet 3270 sessions you want to support. The maximum value depends on the number of sessions you have purchased.
- ***Enable TN3270E Support***. Allows the user to enable/disable support for RFC 1647 in the OC/Telnet Server. The default is disabled.
- ***TN3270E Function Support***. If TN3270E support is enabled, the user may enable/disable individual TN3270E functions. The default is ALL enabled.
- ***SCS-CTL-CODES function***. Required for SCS (LU type 1) Print sessions
- ***DATA-STREAM-CTL function***. Required for 3270 Data Stream (LU type 3) Print sessions.
- ***RESPONSES function***. Provides support for definite, exception and no response requests.
- ***BIND-IMAGE function***. Forwards BIND/UNBIND notifications to client.
- ***SYSREQ function***. Function emulates the SYSREQ key in an SNA environment.

```
OCS occonfig            Telnet 3270 Server Configuration

Enter the number of 3270 Telnet Server sessions............. 16




ESC+1=Help      ESC+2=ExitPgm   ESC+3=PrevMenu        OCS II RISC/6000 V3.7.5
ESC+4=ConfMenu  ESC+5=MainMenu  Return=AcceptMods          Instance=vm_via3174
```

Figure 129. TELNET 3270 Server Configuration

- ***Number of 3270 Telnet Server sessions.*** Determines the maximum number of allowable requests to the Telnet Server for a 3270 emulation. This number depends on the OCS II Gateway Software you have purchased.

```
OCS occonfig                PU T2.1 Node Configuration

Enter the Network Identifier for this unit.................. DEIBMIPF

Session increment for CNOS source negotiator................ 8
Number of above sessions that are †source winners†.......... 4
Gateway initiated sessions select source losers?............ Y




ESC+1=Help      ESC+2=ExitPgm   ESC+3=PrevMenu        OCS II RISC/6000 V3.7.5
ESC+4=ConfMenu  ESC+5=MainMenu  Return=AcceptMods          Instance=vm_via3174
```

Figure 130. PU T2.1 Node Configuration

- ***Network Identifier***. Network Identifier defines the SNA NETID associated with OCS Server. This field requires an eight character symbolic name identifying your network name; it **must match** the **NETID** parameter in the **ATCSTRxx** VTAM startup member (see A.2, "VSE VTAM Start Options List" on page 341).
- Keep the defaults of CNOS source negotiator and number of sessions. They do not apply to our configuration.

12.4.5 OCS II RISC/6000 Data Link Configuration

This is the last configuration menu. Here you must describe the physical data link characteristics of your RISC/6000. Presently the OCS II Gateway supports the SDLC and Token-Ring interfaces; in our installation we have configured and tested the Token-Ring interface.

```
  OCS occonfig                Instance Configuration Menu


  Local Area Network Configuration
  SNA Level Data Link Configuration
  Physical/Logical Unit Configuration
  System Wide Configuration Options
  Modify/View Data Link Configuration













ESC+1=Help      ESC+2=ExitPgm   ESC+3=PrevMenu        OCS II RISC/6000 V3.7.5
ESC+4=ConfMenu  ESC+5=MainMenu  Return=AcceptMods         Instance=vm_via3174
```

Figure 131. Instance Configuration Menu

With the fifth option on the Instance Configuration Menu you select: **Modify/View Data Link Configuration**. You are then asked to introduce the relative Link Station (LS) number. This number is used to create an ordered list of the LSs and if it is your first LS definition, **1** (the default) has to be entered, since non-continuous numbering is not allowed.

```
OCS occonfig               Link Configuration

Link Station Type.......... TOKN      Valid values - SDLC, TOKN, X.25 or SCA.


ESC+1=Help      ESC+2=ExitPgm   ESC+3=PrevMenu         OCS II RISC/6000 V3.7.5
ESC+4=ConfMenu  ESC+5=MainMenu  Return=AcceptMods          Instance=vm_via3174
```

Figure 132. RISC/6000 Link Configuration

```
OCS occonfig           Token Ring Data Link Configuration

Adapter Name............... tok0      Adapter name for this station
Call Type.................. C         C=Call out, L=Listen for call
Drop Link on Inact?(Y/N)... N         Y=Drop Link on Inact
Receive Window............. 1         Receive Window size
Transmit Window............ 7         Transmit Window size
Retransmit Count........... 8         Number (1-50) of xmit retries
Retransmit timeout......... 14        Number (1-250) of (.5 sec) intervals
                                      to wait before retransmitting
                                      an unacknowledged I-frame.
Inactivity timeout......... 48        Idle Line timeout value (1-120 secs)
Force Disconnect Timeout... 120       Number (1-600) of seconds to wait for
                                      disconnect response before forcing
                                      the link disconnect
DLC Buffer Size............ 2057      Datalink buffer size
Link Name.................. IPFP2209  Link name for this DLC entry


ESC+1=Help      ESC+2=ExitPgm   ESC+3=PrevMenu         OCS II RISC/6000 V3.7.5
ESC+4=ConfMenu  ESC+5=MainMenu  Return=AcceptMods          Instance=vm_via3174
```

Figure 133. RISC/6000 Token Ring Data Link Configuration

In Figure 133 and Figure 135 on page 178 you describe the Link Configuration parameters. They are presented in a different order than their actual appearance for the sake of easier understanding.

Link Station parameters. The following parameters specify the physical link configuration of your RISC/6000.

- ***Link Station type.*** This field has only two possible values:
  - ***SDLC*** = SDLC data link control
  - ***TOKN*** = Token-Ring data link control
- ***Adapter name.*** This name **must match** the adapter name specified in the Token-Ring adapter configuration panel of the AIX SMIT facility (Figure 135 on page 178).

  In order to find the SMIT Change/Show Characteristics of a Token Ring Adapter, you can specify **SMIT** at the AIX System prompt, this will give you the System Management Panel, here you have to select: **Devices**, then on the next panel: **Communication,** followed by: **Token Ring Adapter**. On the next panel you have to specify **Adapter**, and on the following panel, specify: **Change/Show Characteristics of a Token Ring Adapter**, and you should now be able to see the Change/Show Characteristics of a Token Ring Adapter, and to find the Adapter Name.
- ***Call type.*** This field indicates the call procedure for the connection to the host. Specify:
  - ***L*** = the RISC/6000 station waits for the host activation to make the data link connection with it. In this case it is necessary to specify a **PATH** statement in the VTAM Switched major node (see A.3, "VSE TCPSW.B Switched Major Node" on page 342).
  - ***C*** = the RISC/6000 station initiates a data link connection by calling the host. We specify ′C′ because a remote link address is specified in Figure 134 on page 177.
- ***Remote SAP Address.*** This field specifies the SAP Address of the VSE host and **must match** the **SAPADDR** parameter of the **PORT** statement in the VTAM LAN major node (see Figure 30 on page 53).
- ***Remote Link Address.*** This field contains the Token-Ring adapter address of the VSE host and **must match** the **MACADDR** parameter of the **PORT** statement in the VTAM LAN major node (see Figure 30 on page 53).

Network tuning. The following parameters influence the throughput rate, so you have to specify them with care considering compromises between network reliability and throughput efficiency.

- ***Drop link inact.*** This question asks if you want to disconnect the link with the remote host when the RISC/6000 doesn′t receive any data on the link for a specified period of time. This period is calculated from the values from the *Inactivity timeout, Retransmit count* and *Retransmit timeout* keywords.
- ***Receive window.*** This field contains the maximum number of frames to receive from the host before sending an acknowledgment.
- ***Transmit window.*** This field contains the maximum number of frames to send to the host before receiving an acknowledgment.
- ***Retransmit count.*** This field specifies the maximum number of retries to retransmit the same frame when a transmission error occurs.

- ***Retransmit timeout.*** This field specifies the maximum period of time (for Token-Ring it is measured in 0.5 second intervals) that the RISC/6000 waits for a response from the host before retransmitting the same frame.
- ***Inactivity timeout.*** This field represents the number of seconds that the RISC/6000 waits for a data transmission from the host.
- ***Force disconnect timeout.*** This field specifies the number of seconds that the RISC/6000 waits after requesting a disconnection from the link, before the system forces the disconnection.
- ***Acknowledge Timeout.*** This field specifies the maximum period of time (for Token-Ring it is measured in 0.5 second intervals) that the RISC/6000 waits before sending an acknowledgment to the host.
- ***Dynamic Window.*** This field specifies the number of consecutive acknowledged frames from the host before the RISC/6000 can increase its transmit window.

```
  OCS occonfig             Token Ring Configuration


  Remote Sap Address.... 04                    SAP address of the gateway¢s
                                               partner.

  Remote Link........... 0X400020201001        Token Ring address of the
                                               gateway¢s partner.

  Acknowledge Timeout... 1                     Number (1-40) of .5 sec intervals
                                               before acknowledgement is sent.

  NOTE - See Help for default values.







ESC+1=Help      ESC+2=ExitPgm   ESC+3=PrevMenu        OCS II RISC/6000 V3.7.5
ESC+4=ConfMenu  ESC+5=MainMenu  Return=AcceptMods         Instance=vm_via3174
```

Figure 134. Token-Ring Configuration

```
                        Minimum Configuration & Startup

 To Delete existing configuration data, please use Further Configuration menus


Type or select values in entry fields.
Press Enter AFTER making all desired changes.

                                                        Entry Fields
* HOSTNAME                                           aix320
* Internet ADDRESS (dotted decimal)                   192.61.100.81
  Network MASK (dotted decimal)                      255.255.255.0
* Network INTERFACE                                  tr0
  NAMESERVER
           Internet ADDRESS (dotted decimal)         192.61.100.81
           DOMAIN Name                               itsc.ibm.com
  Default GATEWAY Address                            192.61.200.2
  (dotted decimal or symbolic name)
  RING Speed                                         16                      +
  START Now                                          no                      +
  ALTERNATE TOKEN RING address                       0x400010101009          +
  Apply change to DATABASE only                      no                      +
 BOTTOM

F1=Help            F2=Refresh         F3=Cancel          F4=List
Esc+5=Reset        F6=Command         F7=Edit            F8=Image
F9=Shell           F10=Exit           Enter=Do
```

Figure 135. AIX SMIT Token-Ring Adapter Configuration Panel

12.4.6 Generate a Configuration Listing

Choosing this option generates a flat file containing all your definitions of the OCS II Gateway of the instance file selected. The information is formatted in a user friendly way which makes it very easy to check all parameters. Here′s the list of our OCS II configuration:

```
*****************************************************************
*  Instance File
*    Filename : /usr/oc/sna/instance
*****************************************************************
INSTANCE_ENTRY:
NAME = occ1
SUFFIX = occ1
INSTANCE_ENTRY_END:

INSTANCE_ENTRY:
NAME = vm_via3174
SUFFIX = vm_via3174
INSTANCE_ENTRY_END:

*****************************************************************
*  Feature Manager Keys
*    Filename : /usr/oc/sna/etc/ockeys
*****************************************************************
6bx1u-h3lhp-zkkkk-jzmkl-m1qge-5zzx3-qk8ta

*****************************************************************
*  SNA Configuration for Instance vm_via3174
*    Filename : /usr/oc/sna/sna_bin.vm_via3174
*****************************************************************
The total number of control units in this configuration is 1.
Internet address: 192.61.100.81       Gateway Address: 192.61.100.83
Data Link encoding technique: NRZ    Data Link Buffer Size: 2048

Control Unit Information for relative CU number 1:
Unit type: 3274     Controller Name: IPFP2209     XID: 2017E0009
Physical Unit type: PU2.0    Logical Unit Termination Rule: Terminate Self.
Host type: MAINFRAME   Data Link Role: SECONDARY   Station Address: 4
Link Name: IPFP2209

Logical Unit Configuration:
Highest Logical Unit configured is 16.
  LU:  1  Name: IPFT2S9A  Type: Display  Model: TCP/IP
  LU:  2  Name: IPFT2S9B  Type: Display  Model: TCP/IP
  LU:  3  Name: IPFT2S9C  Type: Display  Model: TCP/IP
  LU:  4  Name: IPFT2S9D  Type: Display  Model: TCP/IP
  LU:  5  Name: IPFT2S9E  Type: Display  Model: TCP/IP
  LU:  6  Name: IPFT2S9F  Type: Display  Model: TCP/IP
  LU:  7  Name: IPFT2S9G  Type: Display  Model: TCP/IP
  LU:  8  Name: IPFT2S9H  Type: Display  Model: TCP/IP
  LU:  9  Name: IPFT2S9I  Type: Display  Model: TCP/IP
  LU: 10  Name: IPFT2S9J  Type: Display  Model: TCP/IP
  LU: 11  Name: IPFT2S9K  Type: Display  Model: TCP/IP
  LU: 12  Name: IPFT2S9L  Type: Display  Model: TCP/IP
  LU: 13  Name: IPFT2S9M  Type: Printer  Model: 3287 -- DSC
  LU: 14  Name: IPFT2S9N  Type: Display  Model: 3278 Model 2
  LU: 15  Name: IPFT2S9O  Type: Display  Model: 3278 Model 2
  LU: 16  Name: IPFT2S9P  Type: Display  Model: TCP/IP
```

Figure 136 (Part 1 of 2). Configuration Listing

```
*****************************************************************
*  OPTIONS Configuration for Instance vm_via3174
*    Filename : /usr/oc/sna/etc/options.vm_via3174
*****************************************************************
KEEPALIVE   =   Y
PS_PORT   #   2000
SUPPRESS_ALERTS   =   N
KEYBOARD_RESET_ON_PROG_CHECK   =   Y
ASSIGN_ONLY_ACTIVE_LU   =   N
AYT_INTERVAL   #   100
USERS_TN3270   #   8
NETID   =   DEIBMIPF
NUMSESS   #   8
SRCWIN   #   4
NEGTERM   =   N
TSPORT   #   23
TERMCAP_RAMDISK_SIZE   #   65536
USERS_TSPASS   #   16

*****************************************************************
*  Data Link Configuration for Instance vm_via3174
*    Filename : /usr/oc/sna/dlcfg.vm_via3174
*****************************************************************
TR_DLC_ENTRY :
ADAPTER             =    tok0
CALL_TYPE           =    C
RECEIVE_WINDOW      #    1
TRANSMIT_WINDOW     #    7
DROP_LINK_ON_INACT  =    N
INACTIVITY_TIMEOUT  #    48
RETRANSMIT_COUNT    #    8
RETRANSMIT_TIMEOUT  #    14
FORCE_DISC_TIMEOUT  #    120
REMOTE_SAP_ADDR        #    0X04
REMOTE_LINK_ADDR    =    0X400020201001
ACKNOWLEDGE_TIMEOUT    #    1
DLC_BUF_SIZE        #    2057
LINK_NAME           =    IPFP2209
DLC_ENTRY_END
```

Figure 136 (Part 2 of 2). Configuration Listing

```
****************************************************************
*  Instance File
*    Filename : /usr/oc/sna/instance
****************************************************************
INSTANCE_ENTRY:
NAME = occ1
SUFFIX = occ1
INSTANCE_ENTRY_END:

INSTANCE_ENTRY:
NAME = vm_via3174
SUFFIX = vm_via3174
INSTANCE_ENTRY_END:

****************************************************************
*  Feature Manager Keys
*    Filename : /usr/oc/sna/etc/ockeys
****************************************************************
6bx1u-h3lhp-zkkkk-jzmkl-m1qge-5zzx3-qk8ta

****************************************************************
*  SNA Configuration for Instance occ1
*    Filename : /usr/oc/sna/sna_bin.occ1
****************************************************************
The total number of control units in this configuration is 1.
Internet address: 192.61.100.81      Gateway Address: 192.61.100.83
Data Link encoding technique: NRZ    Data Link Buffer Size: 256

Control Unit Information for relative CU number 1:
Unit type: 3274     Controller Name: IPFP2901     XID: 201700000
Physical Unit type: PU2.0    Logical Unit Termination Rule: Terminate Self.
Host type: MAINFRAME   Data Link Role: SECONDARY   Station Address: 1
Link Name: IPFP2901

Logical Unit Configuration:
Highest Logical Unit configured is 16.
  LU:  1  Name: IPFT2S9A  Type: Display  Model: TCP/IP
  LU:  2  Name: IPFT2S9B  Type: Display  Model: TCP/IP
  LU:  3  Name: IPFT2S9C  Type: Display  Model: TCP/IP
  LU:  4  Name: IPFT2S9D  Type: Display  Model: TCP/IP
  LU:  5  Name: IPFT2S9E  Type: Display  Model: TCP/IP
  LU:  6  Name: IPFT2S9F  Type: Display  Model: TCP/IP
  LU:  7  Name: IPFT2S9G  Type: Display  Model: TCP/IP
  LU:  8  Name: IPFT2S9H  Type: Display  Model: TCP/IP
  LU:  9  Name: IPFT2S9I  Type: Display  Model: TCP/IP
  LU: 10  Name: IPFT2S9J  Type: Display  Model: TCP/IP
  LU: 11  Name: IPFT2S9K  Type: Display  Model: TCP/IP
  LU: 12  Name: IPFT2S9L  Type: Display  Model: TCP/IP
  LU: 13  Name: IPFT2S9M  Type: Printer  Model: 3287 -- DSC
  LU: 14  Name: IPFT2S9N  Type: Display  Model: 3278 Model 2
  LU: 15  Name: IPFT2S9O  Type: Display  Model: 3278 Model 2
  LU: 16  Name: IPFT2S9P  Type: Display  Model: TCP/IP
```

Figure 137 (Part 1 of 2). OCC Configuration Listing

```
****************************************************************
*  OPTIONS Configuration for Instance occ1
*    Filename : /usr/oc/sna/etc/options.occ1
****************************************************************
KEEPALIVE   =   Y
PS_PORT   #   2001
SUPPRESS_ALERTS   =   N
KEYBOARD_RESET_ON_PROG_CHECK   =   Y
ASSIGN_ONLY_ACTIVE_LU   =   N
AYT_INTERVAL   #   100
UNSOLICITED_PSID   =   N
NEGTERM   =   N
TERMCAP_RAMDISK_SIZE   #   65536
USERS_TSPASS   #   16
NETID   =   DEIBMIPF
NUMSESS   #   8
SRCWIN   #   4
CHECK_IP   #   0
CHECK_LU   #   0
CHECK_LU_QUE   =   N
CHECK_LOG   =   N
USERS_TN3270   #   16
TN3270E_SERVER   =   N
TN3270E_FUNCTIONS   #   31

****************************************************************
*  Data Link Configuration for Instance occ1
*    Filename : /usr/oc/sna/dlcfg.occ1
****************************************************************
SCA_DLC_ENTRY :
LOCAL_SAP_VALUE  # 0X01
LINK_NAME        = IPFP2901
DLC_ENTRY_END
```

Figure 137 (Part 2 of 2). OCC Configuration Listing

12.4.7 Load Configuration File Set

For your convenience you can have different configuration file sets saved on various media (for example, local disk). This menu lets you choose your configuration for subsequent processing. For more details about this operation see Chapter 4 of the *OpenConnect Systems OpenConnect Server II Installations and Operations Guide RISC/6000, 350-0285-101.15*.

```
OCS occconfig               Load OCSNA Instance

  Enter an Instance Number to Load or †ALL†...........  2

  Instance        Instance            Instance        Instance
  Number          Name                Number          Name

   1              occ1                 2              vm_via3174
   3                                   4
   5                                   6
   7                                   8
   9                                   10
   11                                  12
   13                                  14
   15                                  16




ESC+1=Help      ESC+2=ExitPgm   ESC+3=PrevMenu        OCS II RISC/6000 V3.7.5
ESC+4=ConfMenu  ESC+5=MainMenu  Return=AcceptMods         Instance=vm_via3174
```

Figure 138. Load Instance Files

Figure 138 displays each OCSNA instance name and it′s corresponding instance number.

A sample entry for an instance file is shown in Figure 111 on page 157.

12.4.8 Save Configuration File Set

After having finished the configuration process, **don′t forget to save** the changes you made, in order to use the actual configuration values at the next OCS II Gateway Software startup. For more details about this operation see Chapter 4 of the *OpenConnect Systems OpenConnect Server II Installations and Operations Guide RISC/6000, 350-0285-101.15*.

```
 OCS occonfig              Save OCSNA Instance

   Enter an Instance Number to Save or †ALL†...........  2

   Instance        Instance          Instance        Instance
    Number           Name             Number           Name

    1              occ1               2              vm_via3174
    3                                 4
    5                                 6
    7                                 8
    9                                 10
    11                                12
    13                                14
    15                                16




ESC+1=Help      ESC+2=ExitPgm   ESC+3=PrevMenu        OCS II RISC/6000 V3.7.5
ESC+4=ConfMenu  ESC+5=MainMenu  Return=AcceptMods          Instance=vm_via3174
```

Figure 139. Save Configuration Updates

12.5 Special Considerations

12.5.1 TCP/IP Node Considerations

To test our environment we configured several TCP/IP nodes in our network (see Figure 13 on page 34). The following notes apply to all TCP/IP nodes which want to communicate with VSE/ESA via the OCS II Gateway.

- The TCP/IP implementation is a standard implementation and you must follow the guidance given in the appropriate book, (for example for the OS/2 2.1 environment: *TCP/IP V2.0 for OS/2 Installation and Administration, SC31-6075*).
- You must include the IP address and hostname for the appropriate OCS II Gateway in the definitions for the route.
- The OCS II Gateway uses the standard TCP/IP ports for TELNET and FTP operations to VSE. The user doesn't need to specify the port number parameter when opening a session to VSE. If the TELNET and FTP servers of AIX (where the OCS II Gateway is installed) are addressed, a different port number needs to be specified. For details see 12.6, "Set up Alternative Ports for AIX TELNET and FTP Servers" on page 186 for setting up alternative port numbers on AIX.

12.5.2 Host Access Table (HAT)

The OC/TELNET Server provides Host Access Table (HAT) services to control terminals for logging on to VSE and to define certain characteristics to classes of terminals. This involves a two-step process:

1. Selection

   The incoming TELNET user is matched with a **HAT** entry in the file 'tstext' which resides in directory */usr/oc/sna/etc*. A **HAT** entry has the format:

   class_name:ip_address:port_number:[ip_address_mask:
   [*port_number_mask:*]]

   After a match is found for that ip_address, the class_name of that entry is used for the next step.

2. Inclusion/Exclusion Process

   The class_name from the first step found will then be used to search for a match in the file 'octermcap' which also resides in directory */usr/oc/sna/etc*. The result of this matching process determines the capabilities of that terminal such as:

   - Auto bypass the TELNET Server prompt
   - Default Service Type/Terminal Type
   - Ability to access a specific LU
   - Whether access to the TELNET Server is denied or permitted.

In our configuration, we made a HAT customization to check the ability to specify TELNET Server prompt bypass and TELNET Server access denial.

Figure 140 and Figure 141 on page 186 show the contents of the HAT we have customized for 'tstext' and 'octermcap'.

```
#------------------------------------------------------------------------#
# Host Access Table (HAT) Section - the @@TSHAT section can reside anywhere
# in the tstext file.
#
# HAT Entry Syntax:
#     entry_name:ip_address:port_#:ip_address_mask:port_#_mask:
#
# The †allhosts† entry will cover all hosts that do not match any other HAT
# entry.
#------------------------------------------------------------------------#
@@TSHAT
tcpcl3:192.61.100.199
allhosts:0:0:0:0:
```

Figure 140. Sample Host Access Table (HAT) 'tstext'

The HAT entries in the 'tstext' file shown above provide for a match of all IP_addresses for the class 'allhosts' except for address '192.61.100.199' which matches class 'tcpcl3'. For more information on the HAT entries for 'tstext' please refer to the OC/TELNET Server Manager documentation.

```
#
# HAT related termcap entry names
#
# Modify the following lines for HAT implementation.
# Example:
#
tcpcl3:Da:Dp:sv=3278:tt=DEC-VT220:
allhosts:No:
#host1:Tl:tl=5:Da:Dp:sv=3278:tt=vt100:Su:
#host2:Tl:tl=5:tt=tvi925:tc=host1:
```

Figure 141. Sample Host Access Table (HAT) ′octermcap′

The first entry in ′octermcap′ shown above gives user ′tcpcl3′ the ability to bypass the TELNET prompt when logging on to VSE. The second entry denies TELNET access for ′allhosts′ that is, no TCP/IP host can access VSE. Since ′tcpcl3′ was placed before the ′allhosts′ entry, it takes priority over the second entry and allows ′tcpcl3′ to login with the service 3278 (3278 emulation) and is recognized as terminal type VT220. For more information on the HAT entries for ′octermcap′ please refer to the OC/TELNET Server Manager documentation.

12.6 Set up Alternative Ports for AIX TELNET and FTP Servers

If OCS II Gateway is installed on a RS/6000 system as the TCP/IP solution for VSE/ESA, there is a high probability that this machine will mainly be used for that purpose. Based on this assumption, it is easier for TCP/IP users to access VSE/ESA via the ′well known′ standard TCP/IP ports, that is, they don′t have to specify a port number when they use TELNET or FTP to access VSE/ESA.

However, for OCS II Gateway administrators and system programmers there might be a need to access the AIX TELNET and FTP Servers. In this case, the file ′/etc/services′ on AIX has to be modified to assign different port numbers for the local (AIX) TELNET and FTP servers. We assigned ports ′2020 and 2021′ to the FTP server (FTP requires two ports) and port ′2023′ to the TELNET server. After ′/etc/services′ has been changed a refresh of the super daemon ′inetd′ is necessary to make the new port assignments active. This is done by issuing:

root@aix320: /> refresh -s inetd

Now the AIX TELNET and FTP servers can be accessed, as shown in the example below:

[C:\]telnet aix320 -p 2023

[C:\]ftp aix320 2021

Note

For FTP, you have to use the listen port 2021 for connection, port 2020 is for I/O data only.

There also are differences for the TELNET and FTP commands. One has to specify ′-p′ for the port, the other does not. Please refer to your host′s TCP/IP command reference to check the syntax of subject commands.

Figure 142 on page 187 shows part of the file '/etc/services' we used for our environment.

```
echo            7/tcp
echo            7/udp
discard         9/tcp           sink null
discard         9/udp           sink null
systat          11/tcp          users
daytime         13/tcp
daytime         13/udp
netstat         15/tcp
qotd            17/tcp          quote
chargen         19/tcp          ttytst source
chargen         19/udp          ttytst source
ftp-data        2020/tcp        # aix ftp io port
ftp             2021/tcp        # aix ftp listen port
telnet          2023/tcp        # aix telnet listen port
smtp            25/tcp          mail
time            37/tcp          timserver
time            37/udp          timserver
rlp             39/udp          resource # resource location
nameserver      42/udp          name # IEN 116
whois           43/tcp          nicname
domain          53/tcp          nameserver # name-domain server
domain          53/udp          nameserver
mtp             57/tcp           # deprecated
```

Figure 142. Sample for TCP/IP Service Port Number Assignments

Chapter 13. OCS II Gateway Operation

This chapter:

- provides instructions on how to start and operate the OCS II Gateway.

13.1 Operation Samples (start,stop,status....)

The OCS II Gateway is started by an authorized AIX user via the **ocsna** command as shown in Figure 143. Verify that Front Panel Code is always ″001″ after initialization, otherwise the OCS II Gateway will not work properly. If you installed your OCS II in a directory other than ′/**usr/oc/sna**′, you can direct the OCS II process to look for support files by setting an environment variable in the ′/**etc/profile**′ as follows:

OCSNA=perm_path/sna
  export OCSNA

Figure 143 shows how the gateway is started from an AIX command line prompt. By using the ′&′ sign, the gateway program executes as a process in the background. If you first started the gateway without ′&′ and you decide at a later time to run the process in background, simply press **CTRL-Z** to stop the process and enter **bg** followed by the **ENTER** key. The process will then be resumed in the background.

We can also have the program executed automatically at AIX system startup. This requires some customization of the AIX system initialization table. The last command in Figure 144 on page 190 was added to AIX′s system initialization table ′/**etc/inittab**′ to provide for automatic startup of the OCS II Gateway.

```
root@aix320: /usr/oc/bin >./ocsna -i vm_via3174&

OCSNA Process Started :PID=6796 Fri Jul 05 09:24:26 1996


0705 09:24:28 38 days remaining before expiration of the OCSNA
0705 09:24:28 OCS II RISC/6000 V3.7.3 (95/11/14 14:20)
0705 09:24:32 ==> Front Panel Code 001 <==
```

Figure 143. OCSNA Command Sample

```
init:2:initdefault:
brc::sysinit:/sbin/rc.boot 3 >/dev/console 2>&1 # Phase 3 of system boot
powerfail::powerfail:/etc/rc.powerfail >/dev/console 2>&1  # d51225
rc:2:wait:/etc/rc > /dev/console 2>&1  # Multi-User checks
fbcheck:2:wait:/usr/lib/dwm/fbcheck >/dev/console 2>&1 # run /etc/firstb
srcmstr:2:respawn:/etc/srcmstr         # System Resource Controller
rcsna:2:wait:/etc/rc.sna > /dev/console 2>&1 # Start sna daemons
rctcpip:2:wait:/etc/rc.tcpip > /dev/console 2>&1 # Start TCP/IP daemons
cons:0123456789:respawn:/etc/getty /dev/console
piobe:2:wait:/bin/rm -f /usr/lpd/pio/flags/*  # Clean up printer flags files
cron:2:respawn:/etc/cron
qdaemon:2:wait:/bin/startsrc -sqdaemon
writesrv:2:wait:/bin/startsrc -swritesrv
uprintfd:2:respawn:/etc/uprintfd
rcncs:2:wait:sh /etc/rc.ncs
infod:2:once:startsrc -s infod
hcon:2:once:/etc/rc.hcon
ocsna:2:once:/usr/oc/scr/ocsna -i vm_via3174 > /dev/console 2>&1
```

Figure 144. AIX Inittab Sample for OCSNA Auto Startup

To look at the status of your connections with the VSE host you can enter the command **sna_status** and specify the hostname of the gateway (aix320) as shown in Figure 145 on page 191.

```
root@aix320: /usr/oc/bin >sna_status -h aix320

HOST NAME: AIX320
PORT ID:   2000

THE TOTAL NUMBER OF CONTROL UNITS IN THIS CONFIGURATION IS 1.
INTERNET ADDRESSES :
CONTROL UNIT : 192.61.100.81  (C03D6451)    GATEWAY : 192.61.100.83  (C03D64237)


INFORMATION FOR RELATIVE CONTROL UNIT 1:
CONTROL UNIT NAME    : IPFP2209             CONTROL UNIT TYPE : 41-C
SDLC STATION ADDRESS : 04                   DATA LINK NAME    : IPFP2209
SDLC STATION ROLE    : SECONDARY            NRZ/NRZI INDICATOR: NRZ
PHYSICAL UNIT TYPE   : PU2.0                EXCHANGE ID(XID)  : 2017E0009
LOGICAL UNIT TERMINATION RULE : TERMINATE SELF
HIGHEST LOGICAL UNIT CONFIGURED FOR THE CONTROL UNIT IS : 16

TOTAL PIUS IN:      46
TOTAL PIUS OUT:     46
CURRENT PU STATE:   UP - (ACTIVE)

LU DEV     NAME     UNAME     LC    PID ST SS LS TG HTLK BRKT  SNSQ  RNSQ
 1 TCP/IP  IPFT2S9A TCP/IP   2003     13 00 A  I  SS 0000 BETB 00000 00000
 2 TCP/IP  IPFT2S9B TCP/IP   2003     12 00 A  I  SS 0000 BETB 00000 00000
 3 TCP/IP  IPFT2S9C TCP/IP   2003     11 00 A  I  SS 0000 BETB 00000 00000
 4 TCP/IP  IPFT2S9D TCP/IP   2003     10 00 A  I  SS 0000 BETB 00000 00000
 5 TCP/IP  IPFT2S9E TCP/IP   2003      9 00 A  I  SS 0000 BETB 00000 00000
 6 TCP/IP  IPFT2S9F TCP/IP   2003      8 00 A  I  SS 0000 BETB 00000 00000
 7 TCP/IP  IPFT2S9G TCP/IP   2003      7 00 A  I  SS 0000 BETB 00000 00000
 8 TCP/IP  IPFT2S9H TCP/IP   2003      6 00 A  I  SS 0000 BETB 00000 00000
 9 TCP/IP  IPFT2S9I TCP/IP   2003      5 00 A  I  SS 0000 BETB 00000 00000
10 TCP/IP  IPFT2S9J TCP/IP   2003      4 00 A  I  SS 0000 BETB 00000 00000
11 TCP/IP  IPFT2S9K TCP/IP   2003      3 00 A  I  SS 0000 BETB 00000 00000
12 TCP/IP  IPFT2S9L TCP/IP   2003      2 00 A  I  SS 0000 BETB 00000 00000
13 PRT DSC IPFT2S9M            0      0 00 A  I  SS 0000 BETB 00000 00000
14 3278-2  IPFT2S9N            0      0 00 A  I  SS 0000 BETB 00000 00000
15 3278-2  IPFT2S9O            0      0 00 A  I  SS 0000 BETB 00000 00000
16 TCP/IP  IPFT2S9P TCP/IP   2003      1 00 A  I  SS 0000 BETB 00000 00000
```

Figure 145. SNA_STATUS Command Sample

To stop OCSNA processing and end all sessions established you just need to ″kill″ the OCS process. After logging in as an authorized user, you may want to display the process id of OCSNA and use the command ′kill′ to stop the process. The shutdown process is shown in Figure 146 below.

```
ROOT@AIX320: / >PS -EF|GREP OCSNA
    ROOT 7294    1   0 09:42:15      -  0:02 OCSNA -I VM_VIA3174
    ROOT 8407 7879   4 10:21:59  PTS/1  0:00 GREP OCSNA
ROOT@AIX320: / >KILL 7294
```

Figure 146. OCSNA Shutdown Sample

Chapter 14. OCS Print Server Installation and Customization

This chapter covers the OCS Print Server running on a RISC/6000 and provides:

- a functional overview of the OCS Print Server
- detailed instructions on how to install the OCS Print Server
- detailed instructions on how to customize the OCS Print Server
- detailed instructions on how to define OCS Print Server sessions in CICS and VTAM on VSE/ESA

14.1 OCS Print Server Overview

The OCS Print Server is an AIX process to emulate multiple 3287 CICS Report Controller terminal printers.

The print output is automatically converted from EBCDIC to ASCII and routed to any local or remote printer defined on the AIX system.

OCS Print Server may be installed on any AIX system and may communicate with any OCS II Gateway via TCP/IP.

Communication with the Gateway is performed using one of the following protocols:

- TN3287 protocol developed by OCS
- OCS protocol

The OCS Print Server can be attached to CICS by using one of the following SNA protocols:

- LU1 (also known as SCS operation)
- LU3 (DSC operation)

14.2 OCS Print Server Installation

This is a very simple step: It basically consists of installing the OCS Print Server product diskette. Please refer to Chapter 1 of the *OCS Print Server for UNIX - User's Guide* that describes the installation task in detail. The installation command sequence is the following:

1. **mkdir /usr/tmp/oc**
2. **cd /usr/tmp/oc**
3. **tar -xvf /dev/***rdevname* **./install** where *rdevname* is the I/O device name, (for example, rfd0 for the diskette drive)
4. **./install** to start the installation script

The following figures shows the messages appearing during the installation process.

```
Installation Script for the OCS Print Server
Copyright 1994 OpenConnect Systems, Inc.

This script will install the Print Server product on this workstation.
You will be able to install the product files in directories of your
choice.  The product files will first be extracted from the distribution
media and stored in a temporary directory.

Press the [Return] key to continue.
```

Press the **Enter** key. You will then be asked for the installation device.

```
Enter a character device name from which to read the installation media.
```

enter ***/dev/****rdevname* where *rdevname* is the I/O device you used on the tar command above.

The following messages will appear:

```
x ./eps, 257195 bytes, 503 media blocks.
x ./eps_start, 14470 bytes, 29 media blocks.
x ./eps_status, 25969 bytes, 51 media blocks.
x ./eps_stop, 14438 bytes, 29 media blocks.
x ./install, 11157 bytes, 22 media blocks.
x ./ocs_elmadmin, 71391 bytes, 140 media blocks
x ./ocs_elmalert, 48161 bytes, 95 media blocks.
x ./ocs_elmd, 107776 bytes, 211 media blocks.
x ./ocs_elmrpt, 26248 bytes, 52 media blocks.
x ./ocs_elmusage, 43304 bytes, 85 media blocks.
x ./ocs_elmver, 7656 bytes, 15 media blocks.
x ./printcap, 0 bytes, 0 media blocks.
x ./read.me, 16219 bytes, 32 media blocks.
x ./tracutil, 64888 bytes, 127 media blocks.
Product files extracted.

The product files will now be moved to selected directories.
The default directories are:

      /usr/oc/eps              -  Print Server executables
      /usr/oc/eps              -  Configuration files
      /usr/oc/eps              -  Readme file
      /usr/oc/eps              -  License Manager executables

During installation, you will have the opportunity to change the default
directory paths.

Press the [Return] key to continue.
```

Press the **Enter** key.

You will be asked if you want to change the default paths.

```
1. Change default                        /usr/oc/eps
2. Print Server program executables     /usr/oc/eps
3. Configuration files                  /usr/oc/eps
4. Readme file                          /usr/oc/eps
5. License Manager program executables  /usr/oc/eps

Enter choice of directory to change or [Return] to continue
```

Either enter the corresponding number if you want to change the defaults or press **ENTER** to accept the defaults. If you decided to change the defaults, follow the instructions on the screen. After having either accepted the default or changed the installation paths, you will get the following messages:

```
You have chosen the following directories for installation of the Print Server:

    Print Server program executables    - /usr/oc/eps
    Configuration files                 - /usr/oc/eps
    Readme file                         - /usr/oc/eps
    License Manager program executables - /usr/oc/eps

Enter [Y] to continue with installation, or anything
else to abort.
```

Reply **Y** if the paths are correct.

You will get the following final messages.

```
Installing Print Server program executables into /usr/oc/eps
Installing License Manager program executables into /usr/oc/eps.
Installing configuration files into /usr/oc/eps.
Installing readme file into /usr/oc/eps.


Installation complete.
```

14.3 OCS Print Server Customization

The customization of the OCS II Gateway is a more complex part. It consists of the following three steps:

1. Customization of the OCS Print Server product described in 14.3.1, “Customization of the OCS Print Server” on page 196
2. Customization of the OCS II Gateway described in 14.3.2, “Customization of the OCS II Gateway for OCS Print Server” on page 200
3. Customization in VSE/ESA described in 14.3.3, “Customization in VSE/ESA” on page 200

14.3.1 Customization of the OCS Print Server

The customization of the OCS Print Server product consists of the following steps:

- Providing the license manager key
- Customizing the defaults file
- Customizing the config file
- Starting the OCS Print Server

14.3.1.1 Providing the License Manager Key

Switch to the path on which you installed the License Manager (default is **/usr/oc/eps**) and enter the following command:

./ocs_elmadmin -c -n -e ***path***

where ***path*** is the path you installed OCS Print Server to (default is */usr/oc/eps*). You will be given a code that is used to obtain the product key from OCS. Enter the license key provided by OCS. A file named *01* will be created with the information needed by the License Manager.

14.3.1.2 Customizing the Defaults File

The defaults file contains configuration-wide defaults. It must be placed in the path in which the OCS Print Server is installed (default is */usr/oc/eps*).

> **Note**
>
> OCS Print Server supports both OCS and TN3278 protocol.
>
> In our installation we only used the OCS protocol, and therefore only describe the configuration required by this protocol.
>
> For a description of how to set up OCS Print Server for the TN3278 protocol, refer to OCS Print Server for UNIX - User′s Guide.

Figure 147 shows the defaults file used in our configuration.

```
GATEWAY     =   aix320          # Name (or internet addr) of OC unit
THRESHOLD   =   85              # Congestion stage 1 at 85% usage.
DIRECTORY   =   /tmp            # Monitor /tmp directory for usage
SHELL       =   /bin/sh         # Default shell for print processes
```

Figure 147. Defaults File Used in Our Installation

Table 7 on page 197 contains a short description of the possible parameters used in this file.

Parameter	Description	Default
GATEWAY	Specifies the Name or IP-Adress of the OCS II Gateway	
OCS_PORT	port number for OCS II Gateway	2000
PROTOCOL	protocol type (OCS or TN).	OCS
FORMFEED	YES: causes the emulator to perform a formfeed by inserting a formfeed character into the Print datastream. NO: formfeed is performed by a series of linefeed characters.	NO
DIRECTORY	path to monitor to avoid a disk-full condition.	No monitoring
THRESHOLD	threshold value for the disk utilization of the path specified in the **DIRECTORY** value. If this value is reached, OCS Print Server enters ″congestion avoidance″ mode. For a description of this mode, refer to OCS Print Server for UNIX - User′s Guide.	No monitoring
PTHRESH	maximum number of subprocesses of OCS Print Server before ″congestion avoidance″ mode is entered. For a description of this mode, refer to OCS Print Server for UNIX - User′s Guide.	No subprocess monitoring
PRINTCAPDIR	path that contains the printcap file used to perform the EBCDIC to ASCII conversion. For customization of the printcap file, refer to OCS Print Server for UNIX - User′s Guide.	path, where OCS Print Server is installed.
PREFIXDIR	path to prefix files	no usage of prefix files
POSTFIXDIR	path to postfix files	no usage of postfix files
LINELEN	maximum number of characters that fit on a line at the target printer.	132
BUFSIZE	buffer size used for LU3 type printers	3168
WRITE2FILE	**YES:** Incoming print requests are directly sent to the printer **NO:** print requests are first written to a file and then printed offline	NO
SHELL	fully qualified path to the command shell used for the printing subprocesses.	/usr/bin/sh

Table 7. Default File Parameters and Defaults

14.3.1.3 Customizing the Config File

The config file specifies the actual printer emulation sessions. It must be placed in the path in which the OCS Print Server is installed (default is **/usr/oc/eps**).

Note

OCS Print Server supports both OCS and TN3278 protocol.

In our installation we only used the OCS protocol, and therefore only describe the configuration required by this protocol.

For a description of how to set up OCS Print Server for the TN3278 protocol, refer to OCS Print Server for UNIX - User′s Guide.

Figure 148 shows the defaults file used in our configuration.

```
LU_ENTRY_START:
LUNAME          =       IPFT2S9M
PRTCMD          =       †lpr -Pibm4029l†
LU_ENTRY_END:
```

Figure 148. Config File Used in Our Installation

Every session defined in the config file must be enclosed by **LU_ENTRY_START** and **LU_ENTRY_END**. Table 8 on page 199 contains a short description of the possible parameters used in this file.

Parameter	Description	Default
SESSID	Name of printer emulation session	Value specified for **LUNAME**
LUNAME	name of the LU to be used on OCS II Gateway The LU must be defined as described inChapter 5, "Defining the OCS II Gateway to the Host" on page 41	
LUNUM	LU number to the used on OCS II Gateway (valid only in combination with PUNUM)	
PUNAME	name of the PU to be used on OCS II Gateway (valid only in combination with LUNUM)	
TRACELVL	trace level for this session. For details about tracing see OCS Print Server for UNIX - User′s Guide	No trace
PRTCMD	AIX command used to print requests from this session.	lpr
EXTENDSCS	Allow ″invalid″ control codes in LU1 or LU3 datastreams	YES
SUBCHAR	Character to be substituted for invalid control codes.	′-′
PREFIX	file to be inserted in the print stream before the print request only valid when PREFIXDIR specified in **defaults** file	
POSTFIX	file to be inserted in the print stream after the print request only valid when POSTFIXDIR specified in **defaults** file	
PRINTCAP	path to printcap file. May be specified absolute (leading ′/′) or relative to PRINTCAPDIR (specified in **defaults** file). The printcap file may be used to change the EBCDIC to ASCII translation tables	printcap
PCAPENT	specifies entry in printcap file to use for this sessions. For a detailed description of printcap files refer to OCS Print Server for UNIX - User′s Guide	
INDEX	add job number suffix when WRITE2FILE=YES.	YES
FILENAME	name of the file to use when WRITE2FILE=YES	see OCS Print Server for UNIX - User′s Guide

Table 8. Config File Parameters and Defaults

Additionally the following **defaults** file parameters (as described in14.3.1.2, "Customizing the Defaults File" on page 196) may be overwritten on a session basis:

- PORT
- PROTOCOL
- GATEWAY
- FORMFEED
- LINELEN
- BUFSIZE

- WRITE2FILE
- SHELL

14.3.1.4 Starting the OCS Print Server

To start the OCS Print Server, the following steps have to be performed:

- Startup of the Elan License Manager
- Startup of OCS Print Server backgroup process

To automate this procedure, the AIX in Figure 149 was created.

```
/usr/oc/eps/ocs_elmd -e /usr/oc/eps
/usr/oc/eps/eps&
sleep 10
/usr/oc/eps/eps_status
```

Figure 149. OCS Print Server Startup Shell Script

This shell script performs three tasks:

1. Start the Elan License Manager using the keys from the /usr/oc/eps path
2. Start the OCS Print Server process
3. Check the status of the OCS Print Server process after 10 seconds.

14.3.2 Customization of the OCS II Gateway for OCS Print Server

The LUs used by OCS Print Server must be defined to OCS II Gateway using one of the following LU type definitions:

- Type 9 3287 LU 1 (SCS Operation)
- Type 10 3287 LU 3 (DSC Operation)

The definitions must match the LU type definitions in VTAM as described in 14.3.4, "VTAM Customization." These definitions are done using **occonfig** on OCS II Gateway as described in 12.4.2, "SNA Level Data Link Configuration" on page 163. In our installation we used LU IPFT2S9M defined as LU 3, as shown in Figure 123 on page 167.

14.3.3 Customization in VSE/ESA

The OCS Print Server customization consists of the following steps:

- VTAM customization
- CICS customization

14.3.4 VTAM Customization

The following customization tasks are required in VTAM:

- Define the logmode table and entries for the applications and LUs
- Add the LU definitions for the OCS II Gateway LUs

14.3.4.1 Logmode Table OCSBIND

The sample job listed in A.6, "OCSBIND MODETAB for OCS Products" on page 351 is used to create the OCSBIND logmode table. OCS Print Server uses the OCSSCS or OCSDSC entries listed in Figure 150.

```
OCSSCS   MODEENT LOGMODE=OCSSCS,                                     C
               FMPROF=X¢03¢,                                         C
               TSPROF=X¢03¢,                                         C
               PRIPROT=X¢B1¢,                                        C
               SECPROT=X¢90¢,                                        C
               COMPROT=X¢3080¢,                                      C
               RUSIZES=X¢8587¢,                                      C
               SRCVPAC=X¢02¢,                                        C
               PSNDPAC=X¢04¢,                                        C
               PSERVIC=X¢01000000E100000000000000¢
OCSDSC   MODEENT LOGMODE=OCSDSC,                                     C
               FMPROF=X¢03¢,                                         C
               TSPROF=X¢03¢,                                         C
               PRIPROT=X¢B1¢,                                        C
               SECPROT=X¢90¢,                                        C
               COMPROT=X¢3080¢,                                      C
               RUSIZES=X¢8587¢,                                      C
               SRCVPAC=X¢02¢,                                        C
               PSNDPAC=X¢04¢,                                        C
               PSERVIC=X¢030000000000000000000200¢
```

Figure 150. Logmode Entries Used for OCS Print Server

In our case, OCSSCS can be used for LU1, OCSDSC for LU3 operation.

14.3.4.2 VTAM LU Definition

The complete OCS II Gateway LUs definition is listed in Figure 24 on page 47, and A.3, "VSE TCPSW.B Switched Major Node" on page 342.

Figure 151 lists the LU definitions used by OCS Print Server. We defined LU IPFT2S9M for our test environment.

```
IPFT2S9M LU    LOCADDR=13,DLOGMOD=OCSDSC
```

Figure 151. LU Used for OCS Print Server

The DLOGMOD parameter has to match the definitions made in OCS II Gateway described in 14.3.2, "Customization of the OCS II Gateway for OCS Print Server" on page 200.

- If the LU in OCS II Gateway is defined as 3278 LU1 (code 9), the VTAM DLOGMOD must be an LU1 logmode, in our case OCSSCS.
- If the LU in OCS II Gateway is defined as 3278 LU3 (code 10), the VTAM DLOGMOD must be an LU3 logmode, in our case OCSDSC.

The logmodes OCSSCS and OCSDSC can be found in Figure 150.

14.3.5 CICS Customization

This section covers the definitions in CICS to enable VSE/ESA users to use the OCS Print Server CICS terminal printer. We used the CEDA transaction to define the following resources to CICS:

- Typeterm PSLU3
- Terminal 2S9M

We filed these resources under the group VSEGROUP. After the resources are defined, we used CEDA Install Group(VSEGROUP) to activate the definitions. Partial screen images of the CEDA sessions are reproduced to show the definitions we added or changed.

14.3.5.1 Define CICS Terminal Printer (DSC)

CEDA DEFine TYpeterm(PSLU3) is used to define the OCS Print Server terminal PSLU3 to CICS as shown below.

```
CEDA  DEFine
 TYpeterm       : PSLU3
 Group          : VSEGROUP
RESOURCE TYPE
 DEVice       ==> LUTYPE3
 TERmmodel    ==> 2
 SESsiontype  ==>
 PRINTErtype  ==> 3284
 LDclist        :
 SHippable    ==> Yes                  No | Yes
MAPPING PROPERTIES
 PAGesize     ==> 024 , 080            0-999
 ALTPage      ==> 000 , 000            0-999
 ALTSUffix    ==>
 FMhparm      ==> No                   No | Yes
 OBOperid     ==> No                   No | Yes
PAGING PROPERTIES
 AUTOPage     ==> Yes                  No | Yes
DEVICE PROPERTIES
 DEFscreen    ==> 024 , 080            0-999
 ALTSCreen    ==>     ,                0-999
 APLKybd      ==> No                   No | Yes
 APLText      ==> No                   No | Yes
     .
     .
 FOrmfeed     ==> Yes                  No | Yes
     .
 BAcktrans    ==> No                   No | Yes
 CGcsgid      ==> 00000 , 00000        0-65535
SESSION PROPERTIES
 AScii        ==> No                   No | 7 | 8
 SENdsize     ==> 03840                0-30720
 RECEivesize  ==> 03840                0-30720
 BRacket        : Yes                  Yes | No
LOGMode      ==>
DIAGNOSTIC DISPLAY
ERRLastline    : No                  No | Yes
ERRIntensify   : No                  No | Yes
ERRColor       : NO                  NO | Blue | Red | Pink | Green |
                                     Turquoise | Yellow | NEutral
ERRHilight     : No                  No | Blink | Reverse | Underline
OPERATIONAL PROPERTIES
 AUTOConnect  ==> No                   No | Yes | All
 ATi          ==> Yes                  No | Yes
 TTi          ==> Yes                  Yes | No
 CReatesess   ==> Yes                  No | Yes
 RELreq       ==> Yes                  No | Yes
 DIscreq      ==> Yes                  Yes | No
 Nepclass     ==> 000                  0-255
 SIgnoff      ==> Yes                  Yes | No | Logoff
MESSAGE RECEIVING PROPERTIES
 ROutedmsgs   ==> All                  All | None | Specific
 LOGOnmsg     ==> No                   No | Yes
APPLICATION FEATURES
 BUildchain   ==> No                   No | Yes
 USerarealen  ==> 255                  0-255
 Ioarealen    ==> 00080 , 00080        0-32767
 UCtran       ==> No                   No | Yes | Tranid
RECOVERY
 RECOvoption  ==> Sysdefault           Sysdefault | None
```

CEDA DEFine TErminal(2S9M) is used to define the OCS Print Server terminal 2S9M to CICS as shown below.

```
CEDA  DEFine
 TErminal       : 2S9M
 Group          : VSEGROUP
 AUTINSTModel ==> No                  No | Yes | Only
 AUTINSTName  ==>
TERMINAL IDENTIFIERS
 TYpeterm     ==> PSLU3
 NEtname      ==> IPFT2S9M
 Console      ==> No                  No | 0-99
 REMOTESystem ==>
 REMOTEName   ==>
 Modename     ==>
ASSOCIATED PRINTERS
 PRINTER      ==>
 PRINTERCopy  ==> No                  No | Yes
 ALTPRINTEr   ==>
 ALTPRINTCopy ==> No                  No | Yes
 SPOOLTo      ==>
PIPELINE PROPERTIES
 POol         ==>
 TAsklimit    ==> No                  No | 1-32767
OPERATOR DEFAULTS
 OPERId       ==>
 OPERPriority ==> 000                 0-255
 OPERRsl      ==> 1                                                 0-24,...
 OPERSecurity ==> 1                                                 1-64,...
 Userid       ==>
 NAtlang      ==>
TERMINAL USAGES
 TRansaction  ==>
 TErmpriority ==> 000                 0-255
 Inservice    ==> Yes                 Yes | No
PRINTER DATA
 SPOOLDest    ==> IPFT2S9M
 SPOOLPRTRsl  ==> Public              0-24 | Public
 SPOOLPRTTo   ==> 01                  0-59
 PRINTEDmsg   ==> No                  No | Yes
 PRINTImmed   ==> Yes                 No | Yes
SESSION SECURITY
 SEcurityname ==>
 ATtachsec    ==> Local               Local | Identify | Verify
 Bindpassword ==>                     PASSWORD NOT SPECIFIED
```

Chapter 15. OCS Print Server Operation

This chapter:

- provides instructions on how to start and operate the OCS Print Server
- shows examples how to print a file on OCS Print Server

15.1 Operation Samples (start,stop,status....)

The OCS Print Server is started using the startup script as shown in Figure 149 on page 200.

Figure 152 shows how the print server is started from an AIX command line prompt.

```
root@aix320: /usr/oc/scr >starteps
Jul 25 09:46:48: Elan License Manager - Copyright 1993 Elan Computer Group, Inc
Jul 25 09:46:48: ELMD 2.3.2 started on aix320 (aix320c03d6451).
Jul 25 09:46:48: Vendor ID = 1b94ae; Hostcode = 0500 = HOSTNAME+IPADDR.
Jul 25 09:46:48: PID = 10202.
Jul 25 09:46:48: Port = 6421.
Jul 25 09:46:48: Log file = <none>.
Jul 25 09:46:48: Key path = /usr/oc/eps.
Jul 25 09:46:48: Resource file = <none>.
Jul 25 09:46:48: Message level = 3.
Jul 25 09:46:48: Loaded key /usr/oc/eps/01 (ocsps) 10 tokens exp Dec 22 09:00:0
 1996.

IPFT2S9M
        Status   = Up   (PRINTER_AVAILABLE)
        Gateway  = aix320
        Luname   = IPFT2S9M
        Stats    = No jobs have been started on this session.
root@aix320: /usr/oc/scr >
```

Figure 152. OCS Print Server Startup Sample

The following commands are used to operate the OCS Print Server:

1. **eps_status** to check the status of the print server
2. **eps_stop** to stop one or more print session(s) or the whole print server
3. **eps_start** to start one or more print session(s) stopped using eps_stop.

All three commands have the following common parameters:

- **-h | -H** to specify the hostname or ipaddress where the OCS Print Server is running (default is the local host)
- **-p | -P** to specify the command port for the print server (default is **3050**)
- **-s | -S** to specify the print session to which this command relates. For specifiying multiple sessions, a trailing wildcard character '*' may be used (default is **all** sessions)

The **eps_status** command returns the status of the print server sessions. The status is listed in short form as shown in Figure 152.

To get a detailed status, use the **-f** parameter. Figure 153 shows the status returned when using this parameter.

```
root@aix320: / >/usr/oc/eps/eps_status -f

IPFT2S9M
        Status   = Up   (PRINTER_AVAILABLE)
        Gateway  = aix320
        Luname   = IPFT2S9M
        Stats    = No jobs have been started on this session.
        Protocol = OCS    Port        = 2000
        Linelen  = 132    Buffer size = 3168   Line limit = 0
        Formfeed = NO     Index       = YES
        Ex-SCS   = NO     SubXCS      = YES    CharXCS  = -
        PRTCMD   = lpr -Pibm40291
        Printcap = /usr/oc/eps/printcap
        Pcapent  = NONE
        Prefix   = NONE
        Postfix  = NONE
        Shell    = /bin/sh
root@aix320: / >
```

Figure 153. eps_status -f Sample Output

The **eps_stop** command stops the session(s) specified with the **-s** parameter.

If the last active session is stopped, or if this parameter is omitted, the print server process **eps** will automatically stop.

Note

If you accidentally stopped the print server process by stopping the last active print session, you may restart it by starting the eps program in background:

```
/usr/oc/eps/eps&
```

Print sessions that are currently printing will finish before they are stopped.

To stop the session immediately, the **-n** parameter may be specified.

The **eps_start** command is used to start sessions, that were stopped with **eps_stop**.

Note

Since we only defined one print session, we were not able to test the eps_start command, since the print server is always terminated if the last active print session is stopped.

If you stopped the print server process and want to terminate the license manager, you have to kill the licence manager process. Figure 154 on page 207 shows a sample scenario:

```
root@aix320: / >ps -ef | grep ocs_elmd
    root  9473  7923   3 12:05:57  pts/2  0:00 grep ocs_elmd
    root 10202     1   0 09:46:48      -  0:00 /usr/oc/eps/ocs_elmd -e /usr/oc/eps
root@aix320: / >kill 10202
Jul 25 12:08:35: Received signal 15. ELMD Terminated.

root@aix320: / >
```

Figure 154. Stopping the Licence Manager

15.2 Printing Files from CICS Report Controller

Before you can use the OCS Print Server you have to start the printer as defined to CICS.

The printer is either started in CICS Report Controller (transaction **CEMS**) using **printer selection** or by using the VSE/POWER command **PSTART DEV**. Figure 155 shows a sample response.

```
pstart dev,2s9m,syscicsa,a
AR 0015 1C39I COMMAND PASSED TO VSE/POWER
F1 0001 1QY3I  DEVICE 2S9M STARTED, DDS=SYSCICSA, TIME=16:32:43
F1 0001 1QY2I  DEVICE 2S9M WAITING FOR WORK, DDS=SYSCICSA
F2 0002 DFH5491  TASK 00610 , TERMINAL 2S9M - THIS TERMINAL, SERVING
DESTINATION IPFT2S9M HAS BEEN STARTED. 16:32:43 07/25/96
```

Figure 155. PSTART DEV Sample Console Output

After the device has been started, you can print files using CICS Report Controller or by specifying DEST=(*,*dest*) on the VSE/POWER *** $$ LST** cards.

dest stands for the destination that is served by this printer, as indicated by message **DFH5491** in Figure 155.

When a print request is processed by OCS Print Server, you will get messages on the AIX user ID that started the print server process on AIX. Figure 156 shows some of these messages.

```
Jul 25 14:54:20: root@aix320 [1], ocsps (01): LID 1 issued.
Jul 25 14:54:21: root@aix320 [1], ocsps (01): LID 1 returned (used: 0:
```

Figure 156. PSTART DEV Sample Console Output

To stop the printer on VSE, either stop it in CICS Report Controller or use the VSE/POWER **PSTOP DEV** command.

Chapter 16. OpenConnection for Channel Installation and Customization

This chapter:

- provides instructions on how to install and customize the OCC hardware.

Please see 2.2, "OpenConnection for Channel (OCC)" on page 8, for a short overview of the OCC.

16.1 Installation Summary

The following will briefly describe the OCC installation and customization procedures. The different steps required for local installation of the OCC are:

1. RISC/6000 host software installation
2. Installing OCC Hardware
3. Configuring OCC Hardware
4. OCC Customization
5. Configuring the SNA host for OCS II Gateway

16.1.1 RISC/6000 Host Software Installation

The OCC software installation on a RISC/6000 host can be classified as the OCS II Gateway Installation.

The OCC product is being installed on a RISC/6000 workstation running the OCS II Gateway with the following software components:

- OCS II Gateway Operating Software
- OCC Administration Software

When the OCC product is installed on a RISC/6000 workstation with OCS II Gateway software, the gateway operating software must be at Version 3.6.4 or later in order to support OCC communication.

16.1.1.1 OCC Administration Software Installation

This is a very simple step: It basically consists of installing the OCC product diskette. Please refer to Chapter 2 of the *OpenConnection for Channel Installation, Administration, and Operations Guide*, 350-0466-101.01 for detailed instructions.

The installation command sequence is the following:

1. **mkdir /usr/tmp/occ**
2. **cd /usr/tmp/occ**
3. **tar -xvf /dev/***rdevname*
   where *rdevname* = I/O device name (for example, rfd0 for the diskette drive)

4. **./occinstall**, to start the installation job

Figure 157 shows the menu displayed when the installation exec (occinstall) is invoked.

```
                    OpenConnect Systems Incorporated
                 OpenConnection Channel Installation Procedure

                 1)  INSTALL or UPGRADE the OCC Supporting Software
                 2)  VERIFY Checksums
                 3)  REMOVE the OCC Supporting Software
                 4)  Get HELP for this Installation Procedure
                 5)  EXIT



                 Copyright 1995.  OpenConnect Systems Incorporated.
                    2711 LBJ Freeway, Dallas, TX  75234
                    Technical Assistance:  (214) 888-0678
                    Sales and Information: (214) 484-5200

Enter selection number:

_
```

Figure 157. OCC Installation Exec Main Menu

By choosing selection "1" the OCC product code is loaded into the following directories:

usr/tmp/oc, usr/oc/occ, and **usr/oc/bin**.

16.1.2 Installing OCC Hardware

The hardware installation of the OCC assumes that the SCSI host (for example, OCS II Gateway) has already been installed on the TCP/IP network, tested, and is operational.

For detailed information on the OCC Hardware Installation, please refer to: *OpenConnection for Channel Installation, Administration, and Operations Guide*, 350-0466-101.01

16.1.3 Configuring OCC Hardware

The OCC is configured as an RMT SCSI tape device with an SCSI channel interface address ID. Under the SCSI address ID, OCC creates eight Logical Units with eight SCSI sub-addresses.

During RISC/6000 power-up initialization, the AIX operating system obtains definitions for SCSI devices attached to the interface.

The RISC/6000 UNIX host performs an auto-configure function during initialization, and checks for newly installed devices on the SCSI interface, automatically configures the devices, and makes them **available**.

The following describes the SCSI configuration:

1. After the power-up initialization of the RISC/6000, verify that the OCC has been detected by displaying a list of installed devices, by entering the following command:

   lsdev -C

2. Verify the OCC device by its preselected SCSI ID address, and note the device names assigned to the OCC, as in the following example, where the preselected SCSI ID address of **4** is reflected by -40, -41,....-47.

   ```
   rmt0     Available 00-01-00-40   Other SCSI Tape Drive
   rmt1     Available 00-01-00-41   Other SCSI Tape Drive
   rmt2     Available 00-01-00-42   Other SCSI Tape Drive
   rmt3     Available 00-01-00-43   Other SCSI Tape Drive
   rmt4     Available 00-01-00-44   Other SCSI Tape Drive
   rmt5     Available 00-01-00-45   Other SCSI Tape Drive
   rmt6     Available 00-01-00-46   Other SCSI Tape Drive
   rmt7     Available 00-01-00-47   Other SCSI Tape Drive
   ```

   Under the OCC′s SCSI channel interface address, the OCC creates eight sub-addresses for Logical Units. In the above example, Logical Unit Numbers (LUNs) are represented by device names rmt0, rmt1, ..., rmt7.

   The definitions for the OCC′s eight Logical Units must be modified slightly using the RISC/6000 AIX utility program, the System Management Interface Tool (SMIT).

3. From the AIX prompt display the SMIT System Management screen by entering:

   SMIT

4. From the SMIT System Management screen, display the SMIT Devices by entering:

   Devices

5. From the SMIT Devices screen, display the SMIT Tape Drive screen by selecting:

   Tape Drive

6. From the SMIT Tape drive screen, display the SMIT Tape Drive Selection Window by selecting:

   Change/Show Characteristics of a Tape Drive

7. Select a Logical Unit Number (LUN) definition line, and press the

   enter key

   to display that LUN′s interactive Change/Show Tape Drive Characteristics Screen.

8. In the Change/Show Characteristics of a Tape Drive Screen, change the BLOCK size from **512** to **000**. This tells the system to run with variable block size.

9. In the Change/Show Characteristics of a Tape Drive Screen, change the **Use EXTENDED file marks** from **yes** to **no**.

10. Press the **enter** key to accept the changes and to redisplay the Tape Drive Selection Window.
11. Repeat steps 7 through 10 until all eight tape drives/LUNs have been reconfigured.

16.1.4 OCC Customization

In order to complete the OCC Installation, several customization steps have to be completed:

- Customize the OCS II Gateway
- Configure OCC with OCC Configuration file
- Configure the SNA host for OCS II Gateway

16.1.4.1 Customize OCS II Gateway

The gateway software must be configured to include OCC definitions. We recommend to use the configurator program ′occonfig′.

The following is a short description of these steps.

1. Enter the command:

   occonfig

   You will get the Main Menu as described in Figure 113 on page 158. Continue through the screens until you have the Specify Internet Addresses screen.

   In the following, not every screen is mentioned, it is assumed that you have gone through all the customization screens as described in 12.4, "Customization" on page 156.

2. In the Specify Internet Addresses screen, we entered the following address:

   192.61.100.81, and pressed **enter**

3. On the following screen, the Data Link Values, make sure the Datalink buffer size is **265**, and Datalink encoding is specified as **NRZ**
4. On the Physical Unit Definition screen, we specified the PU name as:

   IPFP2901

5. On the Display All Logical Units, we specified the LU Names as follows:

```
 OCS occonfig               Display All Logical Units
To configure LU¢s, enter the LU name and LU type. See Help for LU types.

    LU Name  Type       LU Name  Type       LU Name  Type       LU Name  Type
 1) IPFT2S9A   14    2) IPFT2S9B   14    3) IPFT2S9C   14    4) IPFT2S9D   14
 5) IPFT2S9E   14    6) IPFT2S9F   14    7) IPFT2S9G   14    8) IPFT2S9H   14
 9) IPFT2S9I   14   10) IPFT2S9J   14   11) IPFT2S9K   14   12) IPFT2S9L   14
13) IPFT2S9M   10   14) IPFT2S9N   1    15) IPFT2S9O   1    16) IPFT2S9P   14
17)                 18)                 19)                 20)
21)                 22)                 23)                 24)
25)                 26)                 27)                 28)
29)                 30)                 31)                 32)
33)                 34)                 35)                 36)
37)                 38)                 39)                 40)
41)                 42)                 43)                 44)
45)                 46)                 47)                 48)
49)                 50)                 51)                 52)
53)                 54)                 55)                 56)
57)                 58)                 59)                 60)
61)                 62)                 63)                 64)

ESC+1=Help      ESC+2=ExitPgm   ESC+3=PrevMenu        OCS II RISC/6000 V3.7.5
ESC+4=ConfMenu  ESC+5=MainMenu  Return=AcceptMods               Instance=ibm1
```

Figure 158. Logical Unit Configuration

6. On the Miscellaneous System Wide Parameters, the Instance Administration Port Number is specified as:

   2001

7. On the Data Link Configuration screen, the relative Link Station number was entered as default:

   1

8. On the Link Configuration screen, the Link Station Type was specified in our case as:

   SCA

9. On the SCA Link Configuration screen, the Host SubChannel Address was specified as:

   01

   And the Link Name as:

   IPFP2901

16.1.4.2 Configure OCC with OCC Configuration File

The OCC configuration file is an ASCII INI file used to drive the OCC Administration Daemon. The OCC configuration can be done in two different ways:

1. Using the interactive configuration program ('configurator').

2. Editing the configuration file **sca.ini** manually.

The first method is recommended as you are prompted to view and/or modify all OCC configuration parameters. To invoke the configuration program, enter the command:

configurator usr/occ/sca.ini

from the usr/oc/occ directory. Figure 159 shows the Main Menu.

```
root@aix320:/oc/occ >scacl
 .start OCC Admin Client session

ScaCmd:
ScaCmd:configurator/usr/occ/sca.ini
********************************************
* SCA Interactiv Configurator started.     *
* Enter ¢q¢ at any time to exit.           *
********************************************

Enter the SCA Resource ID to configure.
Choose from the following SCA Resources.
The default is ¢All SCA Resources¢.

ID Resource
1  SCA Server
2  SCA Startup
3  SCA Device Name
4  SCA Global
5  SCA LDH3274
6  SCA LDHGTO
7  SCA Channel
q  Quit SCA Interactive Configurator
```

Figure 159. OCC Configurator Main Menu

You can select any one of the submenu options or you can press the Enter key to begin step-by-step prompting. If you need more detailed information on the OCC configurator, please refer to Appendix G of the *OpenConnection for Channel Installation, Administration, and Operations Guide*, 350-0466-101-01.

We selected:

- SCA Server
- SCA Startup
- SCA Device name
- SCA Channel

Our configuration output, the sca.ini file is shown in Figure 160 on page 215.

```
PortNumber=500
DefaultDirectory=/usr/oc/occ
SCALoadImage=/usr/oc/occ/lsca.s2
SCADumpFile=/usr/oc/occ/core.sca
[SCAStartup]
LoadSCA=yes
StartSCA=yes
ConfigureSCA=yes
EnableSCA=yes
AutoDump=yes
[UnixDeviceMap]
LUN0=/dev/rmt0
LUN1=/dev/rmt1
LUN2=/dev/rmt2
LUN3=/dev/rmt3
LUN4=/dev/rmt4
LUN5=/dev/rmt5
LUN6=/dev/rmt6
LUN7=/dev/rmt7
[SCA]
MemPercent3274=90
MemPercentGTO=10
[LDH3274]
IBUFsize=265
OBUFsize=4120
IBUFpercent=50
LoMarkpercent=50
[LDHGTO]
IBUFsize=32768
OBUFsize=32768
IBUFpercent=50
[Channel]
Highspeed=no
DataStreaming=no
SelectPriorityHigh=no
```

Figure 160. Configuration File sca.ini

16.1.5 Configure the SNA Host for OCS II Gateway

On the mainframe SNA host, there are I/O and VTAM generation definitions that relate to the OCC and to the OCS II Gateway. The definitions on the SNA host must correlate with the corresponding definitions on the RISC/6000 host for OCC and OCS II Gateway.

In order to define the channel attached OCC to the operating system (VSE/ESA), it may be necessary to do an IOCP generation.

- Channel Path ID (CHPID) Definitions
  - TYPE=BL for Block Multiplexer
  - TYPE=BY for Byte Multiplexer
- Control Unit (CNTLUNIT) Definitions
  - CUNUMBER=280 for Control Unit Address 280
  - PROTOCOL=D for Direct Coupled (DC) Interlocked
  - PROTOCOL=S for Low Speed Datastreaming (up to 3.0 MB/s)
  - PROTOCOL=S4 for High Speed Datastreaming (up to 4.5 MB/s)

- UNIT=3705

- I/O Device (IODEVICE) Definitions
  - UNIT=3705

Our control unit definitions are shown in Figure 161.

```
CNTLUNIT CUNUMBR=2240,PATH=(22),UNIT=3705,UNITADD=((00,32)),SHARED=N, C
               PROTOCL=D
.
IODEVICE ADDRESS=(2200,32),CUNUMBR=(2240),UNIT=3705,MODEL=E8
```

Figure 161. Control Unit Definitions

For the VSE/ESA and VSE VTAM definitions, please refer to 5.3, "VSE/ESA with OSCII Gateway Attached via the OCC" on page 56.

Chapter 17. OpenConnection for Channel Operation

This chapter:

- provides instructions on how to start and operate the OCC hardware.

Please see 2.2, “OpenConnection for Channel (OCC)” on page 8, for a short overview of the OCC.

17.1 OCC Operation Samples

When the OCC is powered up, the channel status online enable LED is on, and the Front Panel Code ″-0001-″ is displayed.

17.1.1 Starting the OCC Administration Daemon

The **scaad** command is used to start the OCC Administration Daemon process. It must be directed to the background using the **&** sign. Figure 162 shows the start script of scaad and the corresponding ocsna process.

```
/usr/oc/bin/scaad&
sleep 30
/usr/oc/bin/ocsna -i occ1&
```

Figure 162. Startsca Script Sample

Figure 163 on page 218 shows how the processes are started using the startsca script.

```
root@aix320: /usr/oc/scr >startsca
Processing completed for INI configuration file /usr/oc/occ/sca.ini.
*****************************************************************
* SCA Admin Process started:  PID=8544  Fri Jul 19 09:52:56 1996
*          Version: 1  Release: 3    (04/03/96 16:28)
*****************************************************************
IB:Started PID=8289

OB:Started PID=9058

AD:SCA is in power-up-reset state
AD:SCA Firmware V1.R3
AD:SCA is setup for AIX
AD:Beginning SCA startup procedures
AD:Firmware compatibility for S2 file is V1.R3
AD:SCA startup procedures completed successfully
AD:Listening on port 500

OCSNA Process Started :PID=7824 Fri Jul 19 09:54:21 1996


0705 09:54:28 24 days remaining before expiration of the OCSNA
0705 09:54:28 OCS II RISC/6000 V3.7.3 (95/11/14 14:20)
0705 09:54:32 ==> Front Panel Code 001 <==
```

Figure 163. SCAAD and OCSNA Startup Sample

To stop scaad processing and end all sessions established you just need to ″kill″ the scaad process. After logging in as an authorized user, you may want to display the process id of scaad and use the command ′kill -9 ProcessID′ to stop all three processes. The shutdown process is shown in Figure 164 below.

```
root@aix320: / >ps -ef|grep ocsna
    root 7294    1   0 09:53:52      -  0:02 ocsna -i occ1
    root 8407 7879   4 10:21:59  hft/0  0:00 grep ocsna
root@aix320: / >kill 7294

root@aix320: / >ps -ef|grep scaad
    root 6694 6437   0 09:52:55  hft/0  0:03 usr/oc/bin/scaad
    root 7207 6694   0 09:53:02      -  0:00 usr/oc/bin/scaad
    root 7264 6694   0 09:59:02      -  0:00 usr/oc/bin/scaad
    root 8407 7879   5 10:21:59  hft/0  0:00 grep scaad
root@aix320: / >kill -9 6694
```

Figure 164. SCAAD and OCSNA Shutdown Sample

Chapter 18. OC/FTP Server Operation and Examples

This chapter:

- provides instructions on how to start and operate the OC/FTP Server at the VSE host side.
- explains how to initiate and execute file transfers from TCP/IP nodes to/from the VSE host.
- gives examples of file transfer operations that were tested in our environment.

18.1 Operation

This section describes how the OC/FTP Server is started and provides an overview of the commands.

18.1.1 VSE Host (FTP Server)

To start OC/FTP Server, the job described in Figure 47 on page 76 is used. Figure 165 below shows a sample console log after having started the OC/FTP Server.

```
Z1 0045 // JOB FTPS
        DATE 07/05/96,CLOCK 11/29/41
Z1 0045 FTPM010I : OPENCONNECT SYSTEMS  VSE FTP SERVER V2.R2.M4.T1(206)

Z1 0045 FTPC030I : CONFIG PROCESSING COMPLETE. 000 ERRORS.
Z1 0045 FTPI020I : CPU ID 70001 IS REGISTERED AS PRIMARY.
Z1 0045 FTPI040I : CPU ID 70001 IS ON TRIAL UNTIL 12/30/96.
Z1 0045 FTPM100I : SERVER USING PORT NUMBER 00021
Z1 0045 FTPN030W : ALL DEFAULT ALLOCATION VALUES WILL BE USED

Z1 0087 FTPS010I : VTAM SESSION ESTABLISH TO LU IPFT2S9A
Z1 0090 ftp> ACF verification for USR10 passed
```

Figure 165. OC/FTP Server Startup Sample

To communicate with the OC/FTP Server running in a dynamic partition enter the following command at the VSE console:

msg *fx*,**data=***cmd*

where *fx* is the partition name and *cmd* is one of the commands described in Table 9.

Command	Explanation
? or help	display help information on operation command.
status	display information about OC/FTP Server such as ACB name, trace status, logins.
quiesce	before shutting down the OC/FTP Server, operator lets current users complete their file transfer operation and disables further logins.
trace,yes	start tracing the OC/FTP Server global packets
trace,no	stop tracing the OC/FTP Server global packets
netstat	display the current users and connections status.
stop	immediate shutdown of the OC/FTP Server forcing the current users.

Table 9. OC/FTP Server Host Side Commands

Figure 166 below shows a sample console log after using the ′status′ operator command. On the status display, the VTAM ACB name can be identified to which the OC/FTP Server is connected, whether trace was started or not, as well as the TCP/IP port number we assigned to OC/FTP Server.

```
Z1-0045 FTP> ENTER COMMAND
45 STATUS
Z1 0045 FTP0130I : CURRENT FTP RUNTIME STATUS:
Z1 0045  ACB NAME: OCSFTPS           LOGINS: ENABLED
Z1 0045     TRACE: NO
Z1 0045      PORT: 00021
Z1 0045   VERSION: V2.R2.M4.T1(206)
Z1 0045 FTP0130I : STATUS   REQUEST COMPLETE.
```

Figure 166. Console Log of Status Operation Command

18.1.2 TCP/IP FTP Clients

This section describes the commands that an FTP client can use. The standard TCP/IP FTP commands (GET, PUT and so on) will not be described, only the particular commands related to OC/FTP Server. For standard FTP commands refer to the appropriate documentation for your FTP Client (for example, *TCP/IP V2.0 for OS/2 Command Reference,* SX75-0070).

OC/FTP Server lets users transfer files to three different subsystems within VSE/ESA:

- VSAM (ESDS, KSDS and SAM)

- Librarian
- POWER

This is controlled via the **SITE** command described below. For samples refer to section 18.2, “Operation Samples” on page 226.

18.1.2.1 SITE Command

This command is used to set the appropriate subsystem environment values for your FTP operation. The OC/FTP server site command is issued from an FTP Client and modifies the options of the OC/FTP Server SITE member ″SITE.TCPIP″. The command syntax is the following:

quote site *P1 P2 ....Pn*

where *P1 P2 ....Pn* are the parameter names followed by their values. For a short description of the SITE parameters, refer to Table 10 on page 222. For a detailed description of the SITE parameters, refer to Appendix I of the *OpenConnect/FTP Client and FTP Server, VSE Installation and User′s Guide* 350-0382-101.05. To enable the subsystem environment via the SITE command enter:

quote site verify

The *access* parameter is specified in a particular way. You only need to enter ′SITE′ followed by one of the following access modes (without specifying the *access* parameter):

quote site vsam/vsamesd/vsamksd/vsamsam/vsesam/libr/user

Note

The access mode ′user′ is for the User I/O exit routine. Please refer to 6.2.6, “User Exits” on page 79 for more information.

The access mode ′vsesam′ is for VSE **non** VSAM sequential operation.

Table 10 (Page 1 of 3). OC/FTP Server Client Site Commands Summary		
Subsystem	**Parameter**	**Explanation**
Generic	Access	Defines the access method for the current FTP operation. Acceptable values are: • VSAM/VSAMKSD/VSAMSAM/VSESAM • LIBR = for Librarian member • USER = User program (in this case you must specify an IOexit value)
	IOexit	Specifies the name of a user written I/O exit routine
	Xlate	Selects the used translation table
VSAM	Adisp	Specifies the VSAM abend disposition
	Blkfactor	The value used for max (record/block) size calculation (default = 1) VSAM and VSESAM mode only.
	Bufndata	Defines the VSAM buffer data number
	Bufnindex	Only for VSAMKSD mode; defines the VSAM buffer index number
	Bufsp	Defines the VSAM buffer space size
	Cafreespace	Only for VSAMKSD mode; defines the Control Area free space for the VSAM file
	Catalog	Specifies the filename of the used VSAM catalog
	Cdisp	Specifies the VSAM close disposition
	Cifreespace	Only for VSAMKSD mode; defines the Control Interval free space for the VSAM file
	CIsize	Defines the Control Interval size of the VSAM files
	Cisizedata	Defines the Control Interval size for the data part of VSAM file
	Cisizeindex	Defines the Control Interval size for the index part of a VSAM/KSDS file
	Dsn	Defines the data set name for receiving data set if none can be determined. VSAM mode only.
	Highkey	Specifies a high key value for a keyrange transfer of a portion of a VSAM/KSDS data set
	Keylength	Only for VSAMKSD mode; defines the VSAM KSDS file keylength used during file transfer

Table 10 (Page 2 of 3). OC/FTP Server Client Site Commands Summary		
Subsystem	**Parameter**	**Explanation**
VSAM	Keylocation	Only for VSAMKSD mode; defines the VSAM KSDS file key start location used during file transfer (begin from position 0)
	Lowkey	Specifies a low key value for a keyrange transfer of a portion of a VSAM/KSDS data set
	Modelentry	Defines the model data set used for current file transfer. Default is NULL (not used)
	Palloc	Defines the primary VSAM record allocation
	RDK	Only for VSAMKSD mode; defines the Replace Duplicate Keys option for KSDS file reinsertion. Usage is RDK ON or RDK OFF
	Recordcount	Provide the ability to transfer part of a file. Specifies the number of records to transfer. Only for VSAM and VSESAM modes.
	RTB	Defines the Remove Trailing Blanks option when VSE sends out a VSAM file. Usage is RTB ON or RTB OFF
	Salloc	Defines the secondary VSAM record allocation
	Skipcount	Defines the number of records skipped before doing the file transfer, only for VSAM and VSESAM.
	Share	Defines the VSAM shareoption for VSAM file
	Units	Defines the unit allocation used for VSAM file. The value can be CYL/TRK/REC
	VSAMrecfm	Defines the VSAM record format for VSAM SAM files. The value can be FIXBLK, FIXUNB, VARBLK, VARUNB or UNDEF, only for VSAM and VSESAM.
	VSAMrecl	Defines the maximum VSAM record length, only for VSAM and VSESAM.
Librarian	LIBRrecf	Defines the LIBR record format parameter. Only two values are possible: • ″R″ = fixed record format • ″B″ = byte string format
	MBRtype	Defines the default member type value if you don′t specify one with the FTP command
	LIBRrecl	Specifies the LIBR member record length
	Lib	The default VSE library name when no target library is specified
	Slib	The default VSE sublibrary name when no target sublibrary is specified

Table 10 (Page 3 of 3). OC/FTP Server Client Site Commands Summary		
Subsystem	**Parameter**	**Explanation**
POWER	Copies	Specifies the number of copies to create
	LSTclass	Defines the default LIST class
	LSTform	Specifies the LIST form number
	LSTpri	Specifies the LIST priority
	LSTrecl	Specifies the LIST record length
	PUNclass	Defines the default PUNCH class
	PUNform	Specifies the PUNCH form number
	PUNpri	Specifies the PUNCH priority
	PUNrecl	Specifies the PUNCH record length
	RDRclass	Defines the default READER class
	RDRpri	Specifies the READER priority
	RDRrecl	Specifies the READER record length
	SYSid	Defines the system id value (necessary when you have a shared spool environment)
	TABs	Specifies the number of blank characters to substitute when FTP receives an ASCII horizontal tab command
	Upcase	Converts the files sent to POWER queues to upper case

18.1.2.2 CD and DIR Commands

The Change Directory command establishes the Current Working Directory (CWD) and DIR displays its contents. These work similar to the UNIX or DOS commands, however, since VSE/ESA has a different file organization within VSAM and the Librarian, semantics and syntax of the above commands are different.

The OC/FTP Server identifies a Librarian member as follows:

library.sublibrary.membername.membertype

and a VSAM file with its ′fileid′ (the same format of the DLBL statement):

for example, **xxxxxxx.yyyyyyyy.zzzzzzz.wwww**.

Table 11 on page 225 contains the correct syntax and the results of these commands.

Subsystem	Client command syntax	OC/FTP Server response
Librarian	**cd** *ftp*	The library *ftp* becomes the Current Working Directory
	cd *ftp.serv*	The sublibrary *ftp.serv* becomes the CWD
	cd *..client*	The library *ftp.client* becomes the CWD
	dir	Displays the contents of the CWD
	dir *'ftp'*	Displays the sublibraries of the *ftp* library
	dir *'ftp.client'*	Displays the contents of the *'ftp.client'* sublibrary
	dir *test.client*	Displays the member *test.client* (only the entry, not the contents!)
	dir *test*.**	Displays the members of the CWD whose names begin with *test*
VSAM	**cd** *'goofie'*	The qualifier *goofie* becomes the CWD (the catalog is chosen with the SITE command!)
	cd *minnie*	The qualifier *minnie* is appended to the CWD
	cd *''*	The CWD is reset
	dir	Displays the contents of the CWD

Table 11. OC/FTP Server CD and DIR Commands

18.1.2.3 POWER Command

The OC/FTP Server **power** command allows an FTP Client to send commands to or obtain data from the VSE/POWER subsystem. The syntax is as follows:

quote power *xxxxxx*

where *xxxxxx* is the VSE/POWER command (for example, *d rdr*). For a complete list and more details of VSE/POWER commands please refer to *VSE/POWER Administration and Operation, SC33-6571*.

18.1.2.4 PUT and GET VSE/POWER Queue Files

The OC/FTP Server allows an FTP Client to send or retrieve data from VSE/POWER queues. The syntax of these commands is as follows:

put *filename $$RDR*
put *filename $$PUN.jobname(.class)*
put *filename $$LST.jobname(.class)*

get *$$RDR.jobname(.class)(.number)(.suffix) filename*
get *$$PUN.jobname(.class)(.number)(.suffix) filename*
get *$$LST.jobname(.class)(.number)(.suffix) filename*

where:

- *filename* is the name of the FTP Client local file
- *jobname* is the name of the VSE/POWER job

- *class* is the class of the VSE/POWER job
- *number* is the number of the VSE/POWER job
- *suffix* is the suffix used for duplicated VSE/POWER jobnames

Note

The parameters in parenthesis are optional, but if you omit one of them a period (″**.**″) must be entered. If you omit the *class* parameter, the default value from the SITE command is used.

18.2 Operation Samples

The following sections show a number of examples of how OC/FTP Server functions can be invoked from a TCP/IP client.

18.2.1 Establish Connection to the OC/FTP Server

Before you can use any FTP commands you need to connect to OC/FTP Server. Figure 167 shows how a connection procedure from an OS/2 2.3 FTP Client works.

```
IBM TCP/IP for OS/2 - FTP Client ver 07:13:23 on Jun 12 1995
ftp> open aix320
Connected to aix320.itsc.ibm.com.
220 VSE/ESA FTP Server ( V2.R2.M4.T1(206)  by OpenConnect Systems ) ready
Name (aix320): pram
331 Password required.
Password: ....
230 User logged in.
ftp>
```

Figure 167. OC/FTP Server Logon Sample

- **aix320** is the symbolic name assigned to the address of the OCS II Gateway. The name was resolved through the OS/2 2.3 Domain Name Server and a session is established with AIX. Refer to 7.2.4.11, "RESOLVER Member" on page 101 for resolving host names through a Domain Name Server.
- **user** and **password** must match one of the values specified in VSE library member **PASSWD.TCPIP** (see Figure 44 on page 72).

For your information

Since we are using the TCP/IP standard FTP listen port (21) for OC/FTP Server, we don′t need to specify a different port number on the FTP Open command (refer to 12.6, "Set up Alternative Ports for AIX TELNET and FTP Servers" on page 186).

18.2.2 Using SITE

Figure 168 gives some examples how to use the SITE command for OC/FTP Server. The first 'quote site' command shows the default values of OC/FTP Server. The second one ('quote site VSAM....') establishes access to VSAM and changes two related parameters. The following 'quote site' command shows the results of the previous 'quote site VSAM' command.

'quote site verify' is required to enable the subsystem environment set by the previous 'quote site' command.

> **Note**
>
> It is recommended that you enter the **quote site** command without any parameter in order to see the result of your changes before issuing the **quote site verify** command.

```
ftp> quote site
202-Current site dependent parameters.
Access       => LIBR     Xlate        => FTPSXLTE IOexit       =>
Catalog      => VSESPUC  Share        => 2        Units        => REC
CIsizedata   => 04096    Palloc       => 0001024  Salloc       => 0000512
VSAMrecl     => 00132    Bufsp        => 0000000  Bufndata     => 00000
Cdisp        => K        Adisp        => K        CIsizeindex  => 00000
Bufnindex    => 00001    Blkfactor    => 00001    Keylocation  => 00000
Keylength    => 00000    Cafreespace  => 00000    Cifreespace  => 00000
VSAMrecfm    => UNDEF    Skipcount    => 00000000 Recordcount  => 00000000
MBRtype      => TCPIP    LIBRrecl     => 00080    LIBRrecf     => R
RDRrecl      => 00080    LSTrecl      => 00132    PUNrecl      => 00080
RDRclass     => 0        LSTclass     => A        PUNclass     => A
RDRpri       => 3        LSTpri       => 3        PUNpri       => 3
LSTform      =>          PUNform      =>          TABs         => 08
Sysid        => N        Copies       => 001      Upcase       => YES
Restart      => OFF
Modelentry   => NULL
Lowkey       => NULL
Highkey      => NULL

202 End of parameters.

ftp>
ftp> quote site vsam vsamrecl80

201 SITE command ready for VERIFY.
ftp>
```

Figure 168. Quote SITE Command Sample

```
ftp> quote site
202-Current site dependent parameters.
Access       => VSAMESD  Xlate        => FTPSXLTE IOexit       =>
Catalog      => VSESPUC  Share        => 2        Units        => REC
CIsizedata   => 04096    Palloc       => 0001024  Salloc       => 0000512
VSAMrecl     => 00080    Bufsp        => 0000000  Bufndata     => 00000
Cdisp        => K        Adisp        => K        CIsizeindex  => 00000
Bufnindex    => 00001    Blkfactor    => 00001    Keylocation  => 00000
Keylength    => 00000    Cafreespace  => 00000    Cifreespace  => 00000
VSAMrecfm    => UNDEF    Skipcount    => 00000000 Recordcount  => 00000000
MBRtype      => TCPIP    LIBRrecl     => 00080    LIBRrecf     => R
RDRrecl      => 00080    LSTrecl      => 00132    PUNrecl      => 00080
RDRclass     => 0        LSTclass     => A        PUNclass     => A
RDRpri       => 3        LSTpri       => 3        PUNpri       => 3
LSTform      =>          PUNform      =>          TABs         => 08
Sysid        => N        Copies       => 001      Upcase       => YES
Restart      => OFF
Modelentry   => NULL
Lowkey       => NULL
Highkey      => NULL

202 End of parameters.
ftp> quote site verify
200 SITE command okay.
ftp>
```

Figure 169. Quote SITE Verify Command Sample

18.2.3 Switching/Displaying Libraries (Librarian)

Figure 170 on page 229 show how to switch between different libraries and display the entire library or parts of it. Before using librarian commands, establish access to Librarian using the **quote site libr** command.

```
ftp> quote site libr
201 SITE command ready for VERIFY.
ftp> quote site
202-Current site dependent parameters.
Access      => LIBR     Xlate       => FTPSXLTE IOexit      =
Catalog     => VSESPUC  Share       => 2        Units       =
CIsizedata  => 04096    Palloc      => 0001000  Salloc      =
 .
 .
 .
ftp> quote site verify
200 SITE command okay.
ftp> cd tcpocs
200 CWD command okay.
ftp> pwd
251 †TCPOCS† is current directory.
ftp> cd tcpocs.ftpc
200 CWD command okay.
ftp> pwd
251 †TCPOCS.FTPC† is current directory.
ftp> cd ..usr1
200 CWD command okay.
ftp> pwd
251 †TCPOCS.USR1† is current directory.
ftp> dir
200 PORT command okay.
150 Opening data connection.
SERVICES TCPIP    96-07-02 12:59.34      103 R        4       80
AIXCON   TXT      96-05-29 13:14.43       68 R        3       80
ftp> dir ¢tcpocs¢
200 PORT command okay.
150 Opening data connection.
 .
 .
TELC     96-05-29 13:14.16      622     3504        0        0 I
USR1     96-05-29 13:14.43        2        8        0        0 A
USR10    96-05-29 13:14.43        0        1        0        0 A
USR2     96-05-29 13:14.43        0        1        0        0 A
USR3     96-05-29 13:14.44        0        1        0        0 A
USR4     96-05-29 13:14.44        0        1        0        0 A
USR5     96-05-29 13:14.44        0        1        0        0 A
USR6     96-05-29 13:14.44        0        1        0        0 A
USR7     96-05-29 13:14.45        0        1        0        0 A
USR8     96-05-29 13:14.45        0        1        0        0 A
USR9     96-05-29 13:14.45        0        1        0        0 A
LPD      96-05-29 13:37.50       37      547        0        0 A
PARMLIB  96-06-17 13:37.50       11      185        0        0 A
SAMMAC   96-06-24 16:48.46      176      527        0        0 A
SAMOBJ   96-06-24 16:49.00      101      818        0        0 A
SAMSMP   96-06-24 16:49.19       96      647        0        0 A
226 Transfer complete.
remote: ¢tcpocs¢
1716 bytes received in 1.50 seconds (1 Kbytes/s)
```

Figure 170 (Part 1 of 2). CD and DIR Librarian Command Samples

```
ftp>
ftp> dir ¢tcpocs.usr1¢
200 PORT command okay.
150 Opening data connection.
 ..USR1.SERVICES TCPIP    96-07-02 12:59.34      103 R        4     80 N
 ..USR1.AIXCON   TXT      96-05-29 13:14.43       68 R        3     80 N
226 Transfer complete.
remote: ¢tcpocs.usr1¢
160 bytes received in 0.22 seconds (0 Kbytes/s)
ftp>
ftp> dir aixcon.txt
200 PORT command okay.
150 Opening data connection.
AIXCON   TXT      96-05-29 13:14.43       68 R        3       80 N -
226 Transfer complete.
remote: aixcon.txt
73 bytes received in 0.18 seconds (0 Kbytes/s)
ftp>
ftp> dir aix*.*
200 PORT command okay.
150 Opening data connection.
AIXCON   TXT      96-05-29 13:14.43       68 R        3       80 N -
226 Transfer complete.
remote: aix*.*
73 bytes received in 0.19 seconds (0 Kbytes/s)
ftp>
```

Figure 170 (Part 2 of 2). CD and DIR Librarian Command Samples

18.2.4 Switching/Displaying VSAM Data Sets

Figure 171 demonstrates the use of CD and DIR commands in the VSAM environment. Change access mode using the **quote site vsam** command.

```
ftp> cd ¢test¢
200 CWD command okay.
ftp> pwd
251 †TEST† is current directory.
ftp> cd ftp
200 CWD command okay.
ftp> pwd
251 †TEST.FTP† is current directory.
ftp> dir
200 PORT command okay.
150 Opening data connection.
CLUSTER ------- TEST.FTP.CLIENT
CLUSTER ------- TEST.FTP.SERVER
226 Transfer complete.
66 bytes received in 14.02 seconds
ftp> cd ¢¢
200 CWD command okay.
ftp> pwd
551 No valid working directory has been established.
```

Figure 171. CD and DIR VSAM Command Samples

18.2.5 Accessing POWER

Figure 172 shows how access to the POWER RDR queue is established.

```
ftp> quote power d rdr,pa*
214-1R46I  READER QUEUE   P D C S  CARDS
    1R46I  PAUSEPC  01846 3 H 0       4 FROM=(USR2)
    1R46I  PAUSEBG  02530 3 L 0       4 FROM=(SYSA)
    1R46I  PAUSEF1  00024 3 L 1       4 FROM=(SYSA)
    1R46I  PAUSEF2  00025 3 L 2       4 FROM=(SYSA)
    1R46I  PAUSEF3  00026 3 L 3       4 FROM=(SYSA)
    1R46I  PAUSEF4  00027 3 L 4       4 FROM=(SYSA)
    1R46I  PAUSEF5  00028 3 L 5       4 FROM=(SYSA)
    1R46I  PAUSEF6  00029 3 L 6       4 FROM=(SYSA)
    1R46I  PAUSEF7  00030 3 L 7       4 FROM=(SYSA)
    1R46I  PAUSEF8  00031 3 L 8       4 FROM=(SYSA)
    1R46I  PAUSEF9  00032 3 L 9       4 FROM=(SYSA)
    1R46I  PAUSEFA  00022 3 L S       4 FROM=(SYSA)
    1R46I  PAUSEFB  00023 3 L T       4 FROM=(SYSA)
214
```

Figure 172. POWER Command Sample

18.2.6 Submitting a Job to VSE/ESA

One of the standard requirements for customers is to manipulate VSE job streams using a workstation editor and to send them to VSE/ESA for execution.

Figure 173 on page 232 shows an example of an AIX shell script (′vsesub′) which we used to submit a VSE job file (′listlib′) to VSE/ESA. The shell script accepts one parameter (the jobname) as input and transfers that job to the POWER reader queue using the FTP ′put′ command. Figure 174 on page 232 shows the VSE/ESA console log after the job was sent from AIX user ′PRAM′.

Note

To avoid the FTP login procedure, we coded a TCP/IP ′.netrc′ file which contains the login user ID and password for the FTP session.

The ′.netrc′ file also includes an FTP macro definition called ′sendjob′. This macro can be executed by preceding it with the ′$′ sign, that is

ftp> $sendjob vsejob

If the macro should be executed automatically after FTP login its name must be changed to init.

For more information on FTP macros refer to the TCP/IP related chapter in *AIX Version 3.2 System User′s Guide: Communications and Networks, GC23-2523*.

```
/u/pram >cat .netrc
machine aix320 21 login pram password pram
macdef init
quote power d rdr,free

macdef sendjob
put $1 $$RDR

macdef listjob
quote power d rdr,free


/u/pram >ls -al vse*
-rwxr--r--   1 pram     staff        378 Jul 17 07:52 vsegetpr
-rwxr--r--   1 pram     staff        289 Jul 17 07:52 vsepr
-rwxr--r--   1 pram     staff        530 Jul 17 09:02 vsesub
/u/pram >
/u/pram >cat vsesub
*
* keep old VSESUB

*
/u/pram >ls -al listlib
-rw-r--r--   1 pram     staff        113 Jul 17 07:57 listlib

/u/pram >cat listlib

* $$ job jnm=liblib,class=a
// job listlib - listing library
// exec libr,parm=¢mshp¢
ld l=tcpocs
/*
/&
* $$ eoj
/u/pram >

/u/pram >vsesub listlib
Verbose mode on.
200 PORT command okay.
150 Opening data connection.
226 Transfer complete.
120 bytes sent in 0.002208 seconds (53.07 Kbytes/s)
221 Goodbye.
Sending job listlib to the VSE system !
/u/pram >
/u/pram >
```

Figure 173. Submitting a Job to VSE/ESA POWER from AIX

```
Z1 0088 ftp> ACF verification for PRAM passed
BG 0001 1Q47I   BG LIBLIB 04458 FROM (PRAM) , TIME=12:13:54
BG 0000 // JOB LISTLIB - LISTING LIBRARY
        DATE 07/17/96,CLOCK 12/13/54
BG 0000 // EXEC LIBR,PARM=¢MSHP¢
BG 0000 EOJ LISTLIB   MAX.RETURN CODE=0000
        DATE 07/17/96,CLOCK 12/14/04,DURATION   00/00/09
BG 0001 1Q34I   BG WAITING FOR WORK
```

Figure 174. VSE/ESA Console Log after Submitting a Job from AIX

18.2.7 Sending Print Lists to VSE/ESA

Another common customer requirement is to use the host printer from LAN attached workstations. This can also be achieved by using the TCP/IP FTP facilities, as shown in the example described below.

Figure 175 shows a shell script ('vsepr') which we used to send a file to VSE's POWER list queue for printing. The shell script accepts two parameters, the name of the file to be printed and the print class on VSE/ESA POWER. The shell script transfers the file into the POWER list queue using again the FTP 'put' command. The owner of the resulting POWER list queue entry is the user who initiated the FTP file transfer for the print file ('PRAM' in our example, see Figure 176).

Figure 176 shows the VSE/ESA console log after the print file was sent from AIX user 'PRAM'.

```
/u/pram >cat vsepr

#!/bin/ksh
ftp aix320 21 << eof       # accept inline command up to eof
verbose on                 # toggle on the display mode
put $1 \$\$LST.$1.$2       # send the listing to VSE
bye                        # quit the ftp session
eof
echo sent listing $1 to the VSE System for printing !

/u/pram >ls -al ocstst*

-rw-r-----   1 pram     staff       1825 Nov 18 14:25 ocstst1
-rw-r-----   1 pram     staff       2025 Jul 17 13:52 ocstst2

/u/pram >vsepr ocstst1 p

Verbose mode on.
200 PORT command okay.
150 Opening data connection.
226 Transfer complete.
2050 bytes sent in 0.006686 seconds (299.4 Kbytes/s)
221 Goodbye.
sent listing ocstst1 to the VSE System for printing !

/u/pram >
```

Figure 175. Sending a Print List to VSE/ESA POWER from AIX

```
d lst,*ocstst
AR 0015 1C39I COMMAND PASSED TO VSE/POWER
F1 0001 1R46I    LIST QUEUE NOTHING TO DISPLAY
Z1 0089 ftp> ACF verification for PRAM passed
d lst,*ocstst
AR 0015 1C39I COMMAND PASSED TO VSE/POWER
F1 0001 1R46I    LIST QUEUE   P D C S  PAGES  CC FORM
F1 0001 1R46I  OCSTST1  04460 3 D P       1   1      FROM=(PRAM)
```

Figure 176. VSE/ESA Console Log after Sending a Print List from AIX

18.2.8 Receiving Print Lists from VSE/ESA

The FTP ′get′ command can be used to transfer print lists from the VSE/ESA POWER list queue to remote TCP/IP nodes and print them at local printers.

Figure 177 on page 235 shows two shell scripts, ′pwrlst′ to display the POWER list queue entries and ′vsegetpr′ to transfer print files from the POWER list queue to TCP/IP nodes for local printing.

While ′pwrlst′ requires only one parameter (the name or part of the name for the list queue entry), ′vsegetpr′ needs three parameters (jobname, class and jobnumber). The shell script could also have asked for only two parameters, jobname and class, then FTP would always retrieve the oldest version of the entry. The class parameter is always required.

```
/u/pram >cat pwrlst

#!/bin/ksh
ftp aix320 21 << eof      # accept inline input up to end of file string
verbose on                # toggle on the display mode
quote power d lst,*$1     # display POWER LST Queue entries
bye                       # quit the ftp session
eof

/u/pram >cat vsegetpr

#!/bin/ksh
ftp aix320 21 << eof      # accept inline input up to end of file string
verbose on                # toggle on the display mode
get \$\$LST.$1.$2.$3 $1   # get the file from list queue of VSE POWER
bye                       # quit the ftp session
eof
lpr $1                    # print the file out to AIX local printer
echo Job $1 retrieved from the VSE System for printing !

/u/pram >ls -al oc*

-rw-r-----   1 pram     staff       2025 Oct 28 15:59 ocs.test.ksds
-rw-r-----   1 pram     staff       1825 Nov 18 14:25 ocstst1
-rw-r-----   1 pram     staff       2025 Jul 17 13:52 ocstst2
-rw-r-----   1 pram     staff       1168 Jul 17 14:07 ocstst3

/u/pram >pwrlst oc

Verbose mode on.
214-1R46I    LIST QUEUE   P D C S  PAGES  CC FORM
    1R46I  OCSTST1  14293 3 D P 1      1   1      FROM=(PRAM)
    1R46I  OCSTST1  14328 3 D P 1      1   1      FROM=(PRAM)
214
221 Goodbye.

/u/pram >vsegetpr ocstst1 p 14293

Verbose mode on.
200 PORT command okay.
150 Opening data connection.
226 Transfer complete.
1850 bytes received in 0.03341 seconds (54.08 Kbytes/s)
221 Goodbye.
Job ocstst1 retrieved from the VSE System for printing !

/u/pram >ls -al oc*
-rw-r-----   1 pram     staff       2025 Oct 28 15:59 ocs.test.ksds
-rw-r-----   1 pram     staff       1825 Nov 18 14:32 ocstst1
-rw-r-----   1 pram     staff       2025 Oct 25 13:52 ocstst2
-rw-r-----   1 pram     staff       1168 Nov 18 14:07 ocstst3

/u/pram >lpstat
Queue   Dev   Status    Job Files         User       PP %   Blks  Cp  Rnte
------- ----- --------- --- ------------- -------- ---- -- ----- --- -----
lp0     lp0   RUNNING     3 ocstst1       pram        0 100    2   1    1
bsh     bshde READY

/u/pram >
```

Figure 177. Receiving Print Lists from VSE/ESA POWER

18.2.9 Transferring Files to/from VSE/ESA

File transfers from an AIX FTP client using 'get'/'mget' and 'put'/'mput' commands follows.

Figure 178 shows the transfer from VSE Librarian members to the AIX client and back into the VSE library.

```
ftp> quote site libr
201 SITE command ready for VERIFY.
ftp> quote site verify
200 SITE command okay.
ftp> dir ¢tcpocs.usr1¢
200 PORT command okay.
150 Opening data connection.
 ..USR1.SERVICES TCPIP    96-07-02 12:59.34      103 R        4       80 N -
 ..USR1.AIXCON   TXT      96-05-29 13:14.43       68 R        3       80 N -
226 Transfer complete.
ftp> pwd
551 No valid working directory has been established.
ftp> cd tcpocs.usr1
200 CWD command okay.
ftp>
ftp> get services.tcpip
200 PORT command okay.
150 Opening data connection.
226 Transfer complete.
2853 bytes received in 0.3018 seconds (9.231 Kbytes/s)
ftp>
ftp> mget a*
mget AIXCON.TXT?
200 PORT command okay.
150 Opening data connection.
226 Transfer complete.
2210 bytes received in 0.09886 seconds (21.83 Kbytes/s)
ftp>
ftp> get services.tcpip services.test
200 PORT command okay.
150 Opening data connection.
226 Transfer complete.
2853 bytes received in 0.1311 seconds (21.26 Kbytes/s)
ftp>
ftp> !ls
AIXCON.TXT         ocstst1           services.test     trash
AixtermLog.IacBm3  ocstst2           smit.log          vsegetpr
Power.dt           ocstst3           smit.script       vsepr
aixdata            prereq            sublist           vsesub
cache.f            ptf               test              vsesub1
info               pwrlst            test.a            vsesub2
listlib            q                 test2
mbox               rmlpp             test3
ocs.test.ksds      services.tcpip    testmem.b
ftp>
```

Figure 178 (Part 1 of 2). Transfering Files to/from VSE/ESA from an AIX FTP Client

```
ftp> put test.a
200 PORT command okay.
150 Opening data connection.
226 Transfer complete.
27 bytes sent in 0.03296 seconds (0.8 Kbytes/s)
ftp>
ftp> mput s*
mput services.tcpip?
200 PORT command okay.
150 Opening data connection.
226 Transfer complete.
2853 bytes sent in 0.008451 seconds (329.7 Kbytes/s)
mput services.test?
200 PORT command okay.
150 Opening data connection.
226 Transfer complete.
2853 bytes sent in 0.007939 seconds (350.9 Kbytes/s)
mput smit.log? n
mput smit.script? n
mput sublist? n
ftp>
ftp> put services.test services.t
200 PORT command okay.
150 Opening data connection.
226 Transfer complete.
2853 bytes sent in 0.007916 seconds (352 Kbytes/s)
ftp>
ftp> dir
200 PORT command okay.
150 Opening data connection.
TEST     A        96-07-17 14:56.28       3 R       1       80 N -
SERVICES T        96-07-17 14:58.26     103 R       4       80 N -
SERVICES TCPIP    96-07-17 14:56.52     103 R       4       80 N -
SERVICES TEST     96-07-17 14:57.07     103 R       4       80 N -
AIXCON   TXT      96-05-29 13:14.43      68 R       3       80 N -
226 Transfer complete.
ftp>
```

Figure 178 (Part 2 of 2). Transfering Files to/from VSE/ESA from an AIX FTP Client

Chapter 19. OC/FTP Client Operation and Examples

There are two ways to initiate file transfer operations from the VSE host:

1. **Using a batch job**

   This requires that all FTP commands have to be imbedded, in logical sequence, in the job. As shown in Figure 50 on page 85 the OC/FTP Client program then manages, together with VTAM, all communications to the OCS II Gateway.

2. **Interactively using CICS transaction ′ftp′**

   This requires that the OC/FTP Client program has been started and is running in a dynamic partition. CICS ISC is used to pass all interactive FTP commands to the OC/FTP Client program (see Figure 192 on page 262).

19.1 FTP Client Command Summary

For both modes of operation the FTP commands available for VSE host users have the same format. The subsystem environment to which the file transfer operations apply is defined in the DYNALL library member (refer to Figure 53 on page 93) and can be adjusted via the defaults command.

For a full description of the OC/FTP Client commands, please refer to Appendix H of the *OpenConnect/FTP Client and FTP Server, VSE Installation and User′s Guide*, 350-0382-101.05.

Table 12 provides a summary of the OC/FTP Client commands.

Table 12 (Page 1 of 7). OC/FTP Client Commands Summary

Command	Parameters	Explanation
append	*vsefile* (*TCP/IPfile*)	The VSE file named *vsefile* is appended to the *TCP/IPfile*. If the second parameter is omitted, the file on the TCP/IP host is named *vsefile*
ascii		The file transfer will be in ASCII mode (default)
binary		The file transfer will be in binary mode
bye		The OC/FTP Client session with the TCP/IP FTP Server is closed and an exit from the OC/FTP Client is performed. It works in the same way as the **quit** command
cd	*TCP/IPdirectory*	The *TCP/IPdirectory* becomes the Current Working Directory (CWD)
close		The OC/FTP Client session with the TCP/IP FTP Server is closed, but no exit from the OC/FTP Client is performed
cls		The screen is cleared
debug		When you activate this option, OC/FTP Client prints every command sent to the TCP/IP host. The debug is set off by default

Table 12 (Page 2 of 7). OC/FTP Client Commands Summary		
Command	**Parameters**	**Explanation**
defaults	**Blkfactor** *number*	Used by the OC/FTP Client to calculate the maximum block or record size for a VSAM file depending on whether the record format is blocked or unblocked (default = 1)
	Bufndata *number*	*number* becomes the default VSAM buffer data (BUFND) for the new VSAM file
	Bufnindex *number*	*number* becomes the new default VSAM buffer index (BUFNI) for the new VSAM file (for VSAM KSDS only)
	Bufsp *number*	*number* becomes the default VSAM buffer space (BUFSP) for the new VSAM file
	Cafreespace *number*	*number* becomes the default VSAM CA free space (CAFREESPACE) for the new VSAM file (for VSAM KSDS only)
	Cifreespace *number*	*number* becomes the default VSAM CI free space (CIFREESPACE) for the new VSAM file (for VSAM KSDS only)
	Cisizedata *number*	*number* becomes the default VSAM data CISIZE (CISIZE.DATA) for the new VSAM file
	Cisizeindex *number*	*number* becomes the default VSAM index CISIZE (CISIZE.INDEX) for the new VSAM file (for VSAM KSDS only)
	clsdsp *disposition*	The *disposition* becomes the new default VSAM file close disposition value (the possible values are the same which are used for the **DISP=** parameter of the VSE **DLBL** statement)
	Dsn *datasetname*	The new default datasetname when the receiving data set name cannot be determined
	HALT	Used to vary support ON, or OFF for OC/FTP Client to produce return codes whenever an error is experienced during FTP execution
	Highkey	Specifies a high key value for a keyrange transfer of a portion of a VSAM/KSDS data set
	Keylength *number*	The keylength for VSAM KSDS file transfer (if not specified, the value will be taken from default model (for VSAM KSDS only))
	Keylocation *number*	The start position of the key counting from 0 (for VSAM KSDS only)
	lib *libname*	The *libname* becomes the new default Library name value

Table 12 (Page 3 of 7). OC/FTP Client Commands Summary		
Command	**Parameters**	**Explanation**
defaults	**list**	Displays the current *vsefile* allocation characteristics contained in the defaults member
	Lowkey	Specifies a low key value for a keyrange transfer of a portion of a VSAM/KSDS data set
	lrecl *number*	The *number* becomes the new default VSAM maximum record size value • **Mode LIBR** Lrecl=80 when Recfm=RECORD • **Mode VSAM** for VSAMSAM, Lrecl*Blkfactor=Maxrecl
	mbrn *mbrname*	The *mbrname* becomes the new default Librarian member name value
	mbrt *mbrtype*	The *mbrtype* becomes the new default Librarian member type value
	mode *libr vsam vsamesd vsamksd vsamsam*	Lets you choose the new subsystem environment for file transfers: • **LIBR** = ftp operations will affect the Librarian • **VSAMESD** = ftp operations will affect VSAMESDS files • **VSAMKSD** = ftp operations will affect VSAMKSDS files • **VSAMSAM** = ftp operations will affect VSAMSAM files
	Modelentry *datasetname*	The name of the model data set instead of using the default model
	opendsp *disposition*	The *disposition* becomes the new default VSAM file open disposition value (the possible values are the same ones that are used for the **DISP=** parameter of the VSE **DLBL** statement)
	palloc *number*	The *number* becomes the new default VSAM file primary allocation value
	RDK	Used only for VSAM/KSDS inbound transfers to VSE, this option denotes (R)eplace (D)uplicate (K)eys
	Recfm	Specifies the record format for transfer data set definition to VSE. Only for LIBR, VSAMSAM, and VSESAM modes
	recordcount *number*	The default threshold value for sending or receiving VSAM data set expressed in number of records
	retain *number*	The *number* becomes the new default VSAM file retention period value (the number format is the same one that is used for the **DATE=** parameter of the VSE **DLBL** statement)

Table 12 (Page 4 of 7). OC/FTP Client Commands Summary		
Command	**Parameters**	**Explanation**
defaults	**recfm** *type*	The default record format for files transferred in LIBR or VSAM mode • **MODE LIBR** – Recfm=RECORD - Fixed record – Recfm=BYTE - Variable byte • **MODE VSAMESD** Recfm is not used • **MODE VSAMKSD** Recfm is not used • **MODE VSAMSAM** Recfm default is UNDEF – Recfm=FIXBLK - Fixed blocked – Recfm=FIXUNB - Fixed unblocked – Recfm=VARBLK - Variable blocked – Recfm=VARUNB - Variable unblocked – Recfm=UNDEF - Undefined
	RTB	Used only for VSAM mode outbound transfers from VSE, denotes (R)emove (T)railing (B)lanks
	salloc *number*	The *number* becomes the new default VSAM file secondary allocation value
	share *number*	The default shareoptions for the VSAM file
	skipcount *number*	The default number of records to be skipped before starting the data transfer
	slib *slibname*	The *slibname* becomes the new default sublibrary name value
	units *cyl trk rec*	The default allocation unit during data transfer
	vsamcat *catname*	The *catname* becomes the new default VSAM catalog
delete	*TCP/IPdirectory*	The *TCP/IPdirectory* will be deleted
dir	(*TCP/IPdirectory*)	The list of the *TCP/IPdirectory* will be printed, if you omit the parameter the list of the CWD will be printed
get MODE=LIBR	*TCP/IPfile* (*mbrname.mbrtype*)	The OC/FTP Client program gets the *TCP/IPfile* and puts it in a VSE library with the specified member name and member type. If you omit the member attributes OC/FTP Client tries to use *TCP/IPfile* value. If the name isn′t compatible with the VSE/ESA file naming conventions, the OC/FTP Client will then change this name for compliance. The **get** command is equal to the **recv** command
get MODE=VSAM	*TCP/IPfile* (*vsefile*)	The OC/FTP Client program gets the *TCP/IPfile* and puts it in a VSAM data set created with the specified *vsefile*. If you omit the member attributes OC/FTP Client tries to use *TCP/IPfile* value. If the name isn′t compatible with the VSAM file naming conventions, the OC/FTP Client will then change this name for compliance

Table 12 (Page 5 of 7). OC/FTP Client Commands Summary		
Command	**Parameters**	**Explanation**
help	(*command*)	The OC/FTP Client program displays the help about the specified *command*. If you omit the argument the OC/FTP Client displays a list of the known commands. **help** command has the same meaning as the **?** command.
lappend	*TCP/IPfile* (*vsefile*)	The *TCP/IPfile* is appended at the end of the *vsefile*. This command is valid only for a sequential data set
ldel	(*vsefile*)	The named *vsefile* is deleted. If you omit the argument the CWD is deleted
ldir	(*vsedir*)	The directory listing of the named library/catalog is displayed. If you omit the argument the CWD is shown
ls	(*TCP/IPdir*)	Displays the contents of the *TCP/IPdir*. If you omit the argument the CWD is shown
mget	*a1 (a2)....(an)*	Retrieves and transfers multiple files from the TCP/IP to the VSE/ESA host (VSAM data set or sublibrary, depending on the MODE selected). *a1 (a2)....(an)* are the file names selected. Specifying ′*****′ means that you want to transfer all files of the CWD. Remember, for MODE=VSAM, the *dsn* value is considered the first qualifier of a VSAM name (for example, with dsn=ftp.test, mget * will retrieve all members named ftp.test.xxxxx.yyyyy)
mkdir	*TCP/IPdir*	Creates a new directory on the TCP/IP host
mkvsedir	*vsedir*	Creates a new sublibrary on the VSE/ESA host
mput	*a1 (a2)....(an)*	Copies multiple VSAM data sets or Librarian members from VSE/ESA to the TCP/IP host. *a1 (a2)....(an)* are the file names selected. Specifying ′*****′ means that you want to transfer all files of the CWD. Remember, for MODE=VSAM, the *dsn* value is considered the first qualifier of a VSAM name (for example, with dsn=ftp.test, mput * will select all VSAM files starting with ′ftp.test.′)
mputucase		This option disables the lower case option (default) of the file names used on an **mput** command
open	*host / IPaddress*	Opens a connection to the named *host* or the *IPaddress* of the TCP/IP host. If you use the *host* the specified name must be present in the VSE **HOSTS** PARMLIB member
pvsedir		Displays the current working directory (CWD) of the VSE host
prompt		Enables or disables the interactive prompting used during **mget/ mput** operations to allow users to individually select the files to be transferred. If prompting is off, all files of the CWD will be transferred

Table 12 (Page 6 of 7). OC/FTP Client Commands Summary		
Command	**Parameters**	**Explanation**
put MODE=LIBR	(*vsefile*) *TCP/IPfile*	The OC/FTP Client program gets and transfers the *vsefile* to the CWD of the TCP/IP host with the specified *TCP/IPname*. If you omit the member attributes, OC/FTP Client tries to use *TCP/IPfile* value. If the name isn′t compatible with the TCP/IP file naming conventions, the OC/FTP Client will then change this name for compliance. The **put** command is equal to the **send** command
put MODE=VSAM	(*vsefile*) *TCP/IPfile*	The OC/FTP Client program gets and transfers the *vsefile* from a VSAM data set to the TCP/IP host CWD with the specified *TCP/IPname*. If you omit the member attributes, OC/FTP Client tries to use *TCP/IPfile* value. If the name isn′t compatible with the TCP/IP file naming conventions, the OC/FTP Client will then change this name for compliance
pwd		Displays the name of the TCP/IP CWD
quit		See the **bye** command
quote	*item1 (item2)...(itemn)*	Send a command (*item1*) from VSE/ESA to the TCP/IP host
recv	*TCP/IPfile (vsefile)*	See the **get** command
rename	*oldfile newfile*	Not supported by the OC/FTP Server for VSE
rmdir	*TCP/IPdir*	Deletes the CWD on the TCP/IP host
rmvsedir	*vsedir*	Deletes the current sublibrary on the VSE/ESA host
send	*vsefile (TCP/IPfile)*	See the **put** command
setdfname	*(domain name suffix)*	Sets, if specified, the Domain name suffix that will be appended to the host name. If you omit the parameter the value will be cleared
setnet	*network name*	Sets the Network name of the OC Server. The name has to match one of the names contained in the VSE PARMLIB member **NETWORKS**
status		Displays the current status of OC/FTP Client
struct	*(t1)*	Sends the file type information to the FTP Server: • **f** = set the file structure to file • **r** = set the file structure to record If you omit the parameter the current structure will be displayed
tcp/iphelp	*(commandname)*	The OC/FTP Client program displays the help information about the specified *command*. If you omit the argument the FTP Server shows a list of available commands
tenex		The file transfer type is set to **tenex** to enable communication with Tenex machines
time		Displays the current time and date

Table 12 (Page 7 of 7). OC/FTP Client Commands Summary		
Command	**Parameters**	**Explanation**
trace		When you activate this option, OC/FTP Client records the packet tracing to the userid.syslog.data. The trace is set off by default
type	*(t1)*	Sends the file transfer type information to the FTP Server: • **ascii** = set the file transfer type to ASCII • **binary** = set the file transfer type to binary • **image** = set the file transfer type to image (same as binary) • **tenex** = set the file transfer type to TENEX If you omit the parameter the current file transfer type is displayed
user		OC/FTP Client prompts you for the user ID and password for the FTP Server. This is usually done when communication to the TCP/IP host is established, unless you start with autologin disabled
verbose		When verbose is ON (default) all responses sent by the TCP/IP host are displayed, including statistical values after file transfer operations
vsecd	*vsedirectory*	The *vsedirectory* becomes the VSE CWD
?	(*command*)	See the **help** command

19.2 Batch Operation Samples

The following examples show the jobs and output listings of file transfer operations from TCP/IP DOS and AIX nodes. An example of OS/2 2.1 directory operations is also provided.

19.2.1 Get a File from PC/DOS to VSE Library

The figures in this section show the FTP client batch job and its corresponding output listings when a file (′TEST.TXT′) is transferred from a workstation running Windows to VSE/ESA.

TCP/IP for PC/DOS Windows (node TCPCL3, see Figure 13 on page 34) has a FTP Daemon (Server) running as a background task.

Note that this example also shows the use of our Domain Name Server (TCPCL1). Figure 179 on page 246 shows the batch job we used to get a file from PC/DOS into the VSE library and Figure 180 on page 246 shows the output produced by the job.

```
* $$ JOB JNM=FTPCBAT,CLASS=Y,DISP=D
// JOB FTPCBAT
// OPTION NOLOG,NOSYSDUMP,JCANCEL
// LIBDEF *,SEARCH=(TCPOCS.FTPC)
// EXEC FTPC,SIZE=(FTPC,256K)
OPEN TCPCL3
pram
jack
PWD
PVSEDIR
LS
GET TEST.TXT
LDIR
BYE
/*
/&
* $$ EOJ
```

Figure 179. VSE/ESA FTP Client Batch Job Example: Get a File from Windows

```
// JOB FTPCBAT                              DATE 07/08/96,CLOCK 13/28/27
// OPTION NOLOG,NOSYSDUMP,JCANCEL
 OpenConnect Systems, VSE FTP Client V2.R2.M4.T1(206)     Task=21

 Cpu id 70001 is registered as primary.

 Cpu id 70001 is on trial until 12/30/96.

 ftp> ACF verification for ¢unknwn¢ passed

 ftp> Initializing LIBR     local directory, press ¢enter¢ to continue

 ftp> LIBR     path now: TCPOCS.USER     VSE.TCPOCS.LIBRARY
```

Figure 180 (Part 1 of 3). FTP Client Batch Job Example Output: Get a File from Windows

```
ftp> OPEN TCPCL3


Attempting to establish a connection with domain name server :
TCPCL1.ITSC.IBM.COM (192.61.100.83  )

Time out. Retrying connection.
Host name resolved through domain name server:
TCPCL1.ITSC.IBM.COM (192.61.100.83  )

TCP Trying 192.61.100.199
Vtam connection established to Lu/Port IPFT2S9B
Connected to TCPCL3.ITSC.IBM.COM

220 tcpcl3 FTP server (2.2.4.1) ready.
Name:
ftp> pram

Password:
331 Password required for pram.
230 User pram logged in.
ftp> ID=pram     IP=192.61.100.199  LU=IPFT2S9B

ftp> PWD
257 †C:\† is current directory.

ftp> PVSEDIR
ftp> LIBR     path is: TCPOCS.USER     VSE.TCPOCS.LIBRARY

ftp> LS

200 PORT command successful.
150 Opening ASCII mode data connection for C:\*.*.
C:\COMMAND.COM
C:\WINA20.386
C:\CONFIG.OLD
C:\AUTOEXEC.OLD
C:\CONFIG.SYS
C:\AUTOEXEC.BAT
C:\AUTOEXEC.BK
C:\AUTOEXEC.BK2
C:\CONFIG.BK4
C:\AUTOEXEC.BK3
C:\CONFIG.BK1
C:\AUTOEXEC.BK4
C:\AUTOEXEC.BK1
C:\CONFIG.BK2
C:\CONFIG.BK3
C:\TEST.TXT
226 Transfer complete.
252 BYTES received in 0 SECONDS (0.00 KBYTES/S)
```

Figure 180 (Part 2 of 3). FTP Client Batch Job Example Output: Get a File from Windows

```
ftp> GET TEST.TXT
200 PORT command successful.
150 Opening ASCII mode data connection for TEST.TXT (61 bytes).
ftp> ID=pram     IP=192.61.100.199  LU=IPFT2S9B
ftp> ID=pram     DN=TCPOCS.USER
ftp> ID=pram     MN=TEST.TXT          RQ=GET OD=N/A CD=N/A
226 Transfer complete.
60 BYTES received in 0 SECONDS (0.00 KBYTES/S)
ftp> ID=pram     BC==========60  RC===========1

ftp> LDIR
TEST     A        96-05-04 15:12.14                   0000003
TCPAPPL  B        95-07-06 09:52.20                   0000089
TESTMEM  B        95-07-06 09:19.22                   0000089
 .
 .
 .
OCSTESTV Z        96-05-04 15:00.50                   0000004
 directory request successful

ftp> BYE

ftp> ID=pram     IP=192.61.100.199  LU=IPFT2S9B
221 Goodbye.
```

Figure 180 (Part 3 of 3). FTP Client Batch Job Example Output: Get a File from Windows

19.2.2 Send a VSE/VSAM KSDS File to AIX

In this example, we first defined a VSAM KSDS file, loaded it with some sample data of 25 records and sent it to AIX.

Note

When this example was created, the OC/FTP Server was using port number 2071 and the AIX FTP Server the TCP/IP standard FTP port 21. Therefore, when no port number is specified in the FTP Open subcommand, the target FTP server will be the one of the AIX system. Also, no resolver file has been created for AIX to use the Domain Name system.

Figure 181 on page 249 shows the job we used to define a VSAM KSDS file. Figure 182 on page 250 shows the job to load and print the file. Finally, Figure 183 on page 251 shows the job and Figure 184 on page 252 its output for the VSE/ESA FTP Client file transfer operation.

```
* $$ JOB JNM=DEFKSDS,CLASS=0,DISP=D
// JOB  DEFKSDS - DEFINE A VSAM KSDS FILE
// EXEC IDCAMS,SIZE=AUTO
 DEFINE CLUSTER ( NAME (OCS.TEST.KSDS) -
        CYLINDERS (1 1) -
        SHAREOPTIONS (2) -
        RECORDSIZE (80  80) -
        VOLUMES (SYSWK4) -
        NOREUSE -
        INDEXED -
        FREESPACE (15 7) -
        KEYS (5  0) -
        TO (99366)) -
        DATA (NAME (OCS.TEST.KSDS.DATA) -
        CONTROLINTERVALSIZE (4096)) -
        INDEX (NAME (OCS.TEST.KSDS.INDEX)) -
       CATALOG (TCPIP.USER.CATALOG)
  IF LASTCC NE 0 THEN CANCEL JOB
/*
// DLBL OCSTST,¢OCS.TEST.KSDS¢,,VSAM,CAT=TCPCAT
/*
// EXEC IESVCLUP,SIZE=AUTO
A OCS.TEST.KSDS                               OCSTST TCPCAT
/*
/&
* $$ EOJ
```

Figure 181. Define and Initialize a VSAM KSDS File

The file's **record length** is 80 and **keylength** is 5 starting from **position** 0.

```
* $$ JOB JNM=LOADKSDS,CLASS=A,DISP=D,NTFY=YES
// JOB  LOADKSDS - LOAD FILE FROM ICCF MEMBER
// DLBL LOADFL,¢OCS.TEST.KSDS¢,,VSAM,CAT=TCPCAT
// EXEC IESVSMLD,SIZE=AUTO
80,K,LOADFL
00001PART1               000500100000000503000239AAAAAAAAAAAAAAAAAAAAAAAA
00002PART2               000600050000050002000239BBBBBBBBBBBBBBBBBBBBBBBB
00003PART3               000100030000000602000239CCCCCCCCCCCCCCCCCCCCCCCC
00004PART4               000200400000050003000239DDDDDDDDDDDDDDDDDDDDDDDD
00005PART5               000100056000056030002 39EEEEEEEEEEEEEEEEEEEEEEEE
00006PART6               000500100000070004000239FFFFFFFFFFFFFFFFFFFFFFFF
00007PART7               000400050000000601000239GGGGGGGGGGGGGGGGGGGGGGGG
00008PART8               000100200000000801000239HHHHHHHHHHHHHHHHHHHHHHHH
00009PART9               000050008000000701000239IIIIIIIIIIIIIIIIIIIIIIII
00010PART10              000050700000006701000239JJJJJJJJJJJJJJJJJJJJJJJJ
00011PART11              000300100000000701200039KKKKKKKKKKKKKKKKKKKKKKKK
00012PART12              000100060000057002000239LLLLLLLLLLLLLLLLLLLLLLLL
00013PART13              000200070000078003000239MMMMMMMMMMMMMMMMMMMMMMMM
00014PART14              000200030000000703000239NNNNNNNNNNNNNNNNNNNNNNNN
00015PART15              000100400000567040002390000000000000000000000000
00016PART16              00005005000000Q004000239PPPPPPPPPPPPPPPPPPPPPPPP
00017PART17              000600050000050040002 39QQQQQQQQQQQQQQQQQQQQQQQQ
00018PART18              000700080000056701000239RRRRRRRRRRRRRRRRRRRRRRRR
00019PART19              000030050000070001000239SSSSSSSSSSSSSSSSSSSSSSSS
00020PART20              000400500000000801000239TTTTTTTTTTTTTTTTTTTTTTTT
00021PART21              000200100000006701000239UUUUUUUUUUUUUUUUUUUUUUUU
00022PART22              000810100000035003003239VVVVVVVVVVVVVVVVVVVVVVVV
00023PART23              000060100000002001003239WWWWWWWWWWWWWWWWWWWWWWWW
00024PART24              000100100000021003000239XXXXXXXXXXXXXXXXXXXXXXXX
00025PART25              000010109900350010032 39YYYYYYYYYYYYYYYYYYYYYYYY
/*
// EXEC IDCAMS,SIZE=AUTO
  PRINT INFILE (LOADFL)  CHARACTER
/*
/&
* $$ EOJ
```

Figure 182. Load and Print the VSAM KSDS File

Notice that the first five bytes will be the key of each record.

```
* $$ JOB JNM=FTPCBAT,CLASS=0,DISP=D
// JOB FTPCBAT
// OPTION NOLOG,NOSYSDUMP,JCANCEL
// LIBDEF *,SEARCH=TCPOCS.FTPC
// EXEC FTPC,SIZE=(FTPC,256K)
OPEN AIX320
pram
jack
DEFAULT MODE VSAMKSD
PUT ocs.test.ksds
DIR
BYE
/*
/&
* $$ EOJ
```

Figure 183. Job to Transfer a VSAM KSDS File to AIX

If current working directory (CWD) is not set, it's going to use the VSAMCAT parm in the default list.

```
// JOB FTPCBAT                  DATE 07/09/96,CLOCK 15/53/33
// OPTION NOLOG,NOSYSDUMP,JCANCEL
 OpenConnect Systems, VSE FTP Client V2.R2.M4.T1(206)     Task=21
 Cpu id 70001 is registered as primary.
 Cpu id 70001 is on trial until 12/30/96.
 ftp> ACF verification for ¢unknwn¢ passed
 ftp> Initializing LIBR     local directory, press ¢enter¢ to continue
 ftp> LIBR     path now: TCPOCS.USER     VSE.TCPOCS.LIBRARY

 ftp> OPEN AIX320
 TCP Trying 192.61.100.81
 Vtam connection established to Lu/Port IPFT2S9B
 Connected to AIX320
 220 aix320 FTP server (Version 4.9 Thu Sep 2 20:35:07 CDT 1993) ready.
 Name:
 ftp> pram

 Password:
 331 Password required for pram.
 230 User pram logged in.
 ftp> ID=pram     IP=192.61.100.81   LU=IPFT2S9B

 ftp> DEFAULT MODE VSAMKSD
 Defaults command completed successfully

 ftp> PUT ocs.test.ksds

 200 PORT command successful.
 150 Opening data connection for ocs.test.ksds.
 ftp> ID=pram     IP=192.61.100.81   LU=IPFT2S9B
 ftp> ID=pram     DN=OCS.TEST.KSDS
 ftp> ID=pram     MN=N/A               RQ=PUT OD=OLD CD=KEEP
 226 Transfer complete.
 2050 BYTES sent in 0 SECONDS (0.00 KBYTES/S)
 ftp> ID=pram     BC=======2,050  RC==========25
 ftp> DIR
 200 PORT command successful.
 150 Opening data connection for /bin/ls.
 total 88
 -rw-r-----   1 pram     staff       2025 Jul 15 15:45 OCS.TEST.KSDS
 drwxr-xr-x   3 pram     staff        512 Jul 15 14:00 info
  .
  .
 -rw-r--r--   1 root     system        14 Jul 15 15:57 test3
 226 Transfer complete.
 707 BYTES received in 0 SECONDS (0.00 KBYTES/S)
 ftp> BYE
 ftp> ID=pram     IP=192.61.100.81   LU=IPFT2S9B
 221 Goodbye.
```

Figure 184. VSE/ESA FTP Client Batch Example: Send a VSAM KSDS File to AIX

The VSAM KSDS file is stored in AIX as a byte stream with a byte containing hexadecimal ′0a′ after each record. There are 25 records of 80 byte characters (including blanks = 2000 bytes), and the FTP file transfer shows that 2050 bytes were sent. An AIX file list shows only 2025 bytes. This is due to the fact that most hosts use two control characters for the end of record indication; AIX, however only uses one (′0a′).

The VSE host sends hexadecimal characters ′0d15′ for <CR> and <LF>. Personal computers use ′0d0a′ for <CR> and <NL> (see the EBCDIC translation table where a hexadecimal ′15′ is translated into hexadecimal ASCII ′0a′). When a session is established to a host, the OpenConnect/File Transfer Program distinguishes between workstations (PCs and UNIX-based) and the host and provides for the necessary translation of subject control characters.

Also keep in mind that VSE files are record oriented, that means, these control characters don′t have to be imbedded in your file.

PC and AIX files, on the other hand, are byte oriented and require subject control characters.

19.2.3 Sending a VSE/VSAM SAM File to AIX

In this example, we first defined a VSAM SAM file. Then, we loaded the file with four records before we sent it to AIX using FTP ′put′.

Note

When this example was created, our OC/FTP Server used port number 2071 and the AIX FTP Server used TCP/IP standard FTP port 21. Therefore, when no port number is specified in the FTP Open subcommand, the AIX FTP server is addressed. Also, the resolver file was not created in AIX to use the Domain Name system.

Figure 185 on page 254 shows the job to define and initialize a VSAM SAM file. Figure 186 on page 254 and Figure 187 on page 255 show the job and output to load and print the file. Figure 188 on page 256 and Figure 189 on page 257 show the job and output to send the VSAM SAM file to AIX.

```
* $$ JOB JNM=DEFSAM,CLASS=0,DISP=D,NTFY=YES
// JOB  PRAM  DEFINE FILE
// EXEC IDCAMS,SIZE=AUTO
 DEFINE CLUSTER ( -
        NAME (OCS.TEST.VSAMSAM) -
        CYLINDERS(1 1) -
        SHAREOPTIONS (2) -
        RECORDSIZE (80 80) -
        RECORDFORMAT (FIXBLK (80)) -
        VOLUMES (SYSWK4) -
        NOREUSE -
        NONINDEXED -
        FREESPACE (15 7) -
        TO (99366)) -
        DATA (NAME (OCS.TEST.VSAMSAM.@D@) -
        CONTROLINTERVALSIZE (4096)) -
        CATALOG (TCPIP.USER.CATALOG)
  IF LASTCC NE 0 THEN CANCEL JOB
/*
// DLBL OCSSAM,¢OCS.TEST.VSAMSAM¢,,VSAM,                              C
               CAT=TCPCAT
/*
// EXEC IESVCLUP,SIZE=AUTO
A OCS.TEST.VSAMSAM                             OCSSAM  TCPCAT
/*
/&
* $$ EOJ
```

Figure 185. Define and Initialize a VSAM SAM File

The file's **record length** is 80 bytes and the **record format** is FIXBLK.

```
* $$ JOB JNM=LOADSAM,CLASS=A,DISP=D,NTFY=YES
// JOB  LOADSAM - LOAD DATA FROM SYSIPT INTO VSAMSAM FILE
// DLBL LOADFL,¢OCS.TEST.VSAMSAM¢,,VSAM,CAT=TCPCAT
// EXEC IDCAMS,SIZE=AUTO
   REPRO INFILE(SYSIPT) -
         OUTFILE(LOADFL)
PRAMOT HEMSTAPAT
KLAUS DIETER WACKER
KLAUS MARUSCHKA
WERNER STIEBER
/*
// EXEC IDCAMS,SIZE=AUTO
 PRINT INFILE (loadFL) -
       CHARACTER
/*
/&
* $$ EOJ
```

Figure 186. Load and Print the VSAM SAM File

```
// JOB  LOADSAM                   DATE 07/08/96,CLOCK 08/46/51
// DLBL LOADFL,¢OCS.TEST.VSAMSAM¢,,VSAM,CAT=TCPCAT
// EXEC IDCAMS,SIZE=AUTO

IDCAMS  SYSTEM SERVICES

   REPRO INFILE(SYSIPT) -
         OUTFILE(LOADFL)
IDC0005I NUMBER OF RECORDS PROCESSED WAS 4
IDC0001I FUNCTION COMPLETED, HIGHEST CONDITION CODE WAS 0

IDC0002I IDCAMS PROCESSING COMPLETE. MAXIMUM CONDITION CODE WAS 0
1S55I  LAST RETURN CODE WAS 0000
// EXEC IDCAMS,SIZE=AUTO

IDCAMS  SYSTEM SERVICES

 PRINT INFILE (LOADFL) -
       CHARACTER
IDCAMS  SYSTEM SERVICES
LISTING OF DATA SET -OCS.TEST.VSAMSAM
RBA OF RECORD - 0
PRAMOT HEMSTAPAT
RBA OF RECORD - 80
KLAUS DIETER WACKER
RBA OF RECORD - 160
KLAUS MARUSCHKA
RBA OF RECORD - 240
WERNER STIEBER
IDC0005I NUMBER OF RECORDS PROCESSED WAS 4
IDC0001I FUNCTION COMPLETED, HIGHEST CONDITION CODE WAS 0
IDCAMS  SYSTEM SERVICES

IDC0002I IDCAMS PROCESSING COMPLETE. MAXIMUM CONDITION CODE WAS 0
1S55I  LAST RETURN CODE WAS 0000
EOJ LOADSAM    DATE 07/08/96,CLOCK 08/47/09,DURATION   00/00/17
```

Figure 187. Load and Print VSAM SAM File Output

```
* $$ JOB JNM=FTPCBAT,CLASS=0,DISP=D
// JOB FTPCBAT
// OPTION NOLOG,NOSYSDUMP,JCANCEL
// LIBDEF *,SEARCH=TCPOCS.FTPC
// EXEC FTPC,SIZE=(FTPC,256K)
OPEN AIX320
pram
jack
DEFAULT MODE VSAMSAM RECFM FIXBLK LRECL 80 RTB ON
VSECD TCPCAT
LDIR
DIR
STATUS
PUT ocs.test.vsamsam
DIR
BYE
/*
/&
* $$ EOJ
```

Figure 188. Send a VSAM SAM File from a VSE/ESA FTP Client to AIX

When sending the VSAM SAM file, we had to change the default mode to 'VSAMSAM', and recfm and lrecl had to be specified. For VSAM outbound transfers we enabled the Remove Trailing Blanks (RTB) option to eliminate trailing blanks in each record.

```
// JOB FTPCBAT                    DATE 07/08/96,CLOCK 13/59/45
// OPTION NOLOG,NOSYSDUMP,JCANCEL
 OpenConnect Systems, VSE FTP Client V2.R2.M4.T1(206)     Task=21


 Cpu id 70001 is registered as primary.

 Cpu id 70001 is on trial until 12/30/96.


 ftp> ACF verification for ¢unknwn¢ passed

 ftp> Initializing LIBR     local directory, press ¢enter¢ to continue

 ftp> LIBR     path now: TCPOCS.USER     VSE.TCPOCS.LIBRARY

 ftp> OPEN AIX320


 TCP Trying 192.61.100.81
 Vtam connection established to Lu/Port IPFT2S9B
 Connected to AIX320


 220 aix320 FTP server (Version 4.9 Thu Sep 02 20:35:07 CDT 1993) ready.
 Name:
 ftp> pram

 Password:
 331 Password required for pram.
 230 User pram logged in.
 ftp> ID=pram     IP=192.61.100.81   LU=IPFT2S9B

 ftp> DEFAULT MODE VSAMSAM RECFM FIXBLK LRECL 80 RTB ON

 Defaults command completed successfully

 ftp> VSECD TCPCAT

 ftp> VSAMSAM  path now: TCPCAT  TCPIP.USER.CATALOG

 ftp> LDIR
 AAAAAAAA.BBBBBB
 AIXDATA.KSDS
 AIXDATA.VSAMSAM
  .
  .
 VTMBK.B1
 VTMBK.B2
 ¢TCPCAT¢ directory request successful
```

Figure 189 (Part 1 of 2). Output of FTP Client Send VSAM SAM File to AIX Operation

```
ftp> DIR
200 PORT command successful.
150 Opening data connection for /bin/ls.
226 Transfer complete.
total 104
-rw-r-----   1 staff       2025 Jul 15 15:45 OCS.TEST.KSDS
-rw-r-----   1 staff          0 Jul 15 17:48 OCS.TEST2.KSDS
-rw-r-----   1 staff       2025 Jul 15 17:50 aixdata
 .
 .
-rw-r--r--   1 system        27 Jul 15 15:57 test2
-rw-r--r--   1 system        14 Jul 15 15:57 test3
777 BYTES received in 0 SECONDS (0.00 KBYTES/S)

ftp> STATUS
Connected to AIX320
Vse execution - Partition(BG) - Task(21)
Type: ASCII ; Structure: FILE    ; Verbose: ON
Prompting: ON ; Debugging: OFF; Tracing: OFF
Current VSE working directory: TCPCAT
Current Domain name server default name suffix:

ftp> PUT ocs.test.vsamsam

200 PORT command successful.
150 Opening data connection for ocs.test.vsamsam.
ftp> ID=pram     IP=192.61.100.81   LU=IPFT2S9B
ftp> ID=pram     DN=OCS.TEST.VSAMSAM
ftp> ID=pram     MN=N/A               RQ=PUT OD=OLD CD=KEEP
226 Transfer complete.
72 BYTES sent in 0 SECONDS (0.00 KBYTES/S)
ftp> ID=pram     BC==========72  RC===========4

ftp> DIR

200 PORT command successful.
150 Opening data connection for /bin/ls.
total 112
-rw-r-----   1 staff       2025 Jul 15 15:45 OCS.TEST.KSDS
-rw-r-----   1 staff          0 Jul 15 17:48 OCS.TEST2.KSDS
-rw-r-----   1 staff       2025 Jul 15 17:50 aixdata
 .
 .
-rw-r--r--   1 system        27 Jul 15 15:57 test2
-rw-r--r--   1 system        14 Jul 15 15:57 test3
226 Transfer complete.
840 BYTES received in 1 SECONDS (0.82 KBYTES/S)
ftp> BYE

ftp> ID=pram     IP=192.61.100.81   LU=IPFT2S9B
221 Goodbye.
```

Figure 189 (Part 2 of 2). Output of FTP Client Send VSAM SAM File to AIX Operation

The FTP file transfer shows that 72 bytes were transferred and appear as 68 bytes in AIX due to the control character elimination (that is, one byte less for each of the four records). Also, since RTB (Remove Trailing Blanks) was set to ′ON′ all trailing blanks were eliminated in each record.

19.2.4 Remote Directory Operations

The example shown in this section demonstrates how to use the OC/FTP Client for manipulating directories on remote TCP/IP nodes from VSE/ESA. Figure 190 shows the job we used for directory operations on an OS/2 2.1 node and Figure 191 on page 260 lists the corresponding output of the OC/FTP Client batch job.

The example also shows how host names are resolved using our OS/2 2.1 Domain Name Server TCPCL1.

```
* $$ JOB JNM=FTPCBAT,CLASS=0,DISP=D
// JOB FTPCBAT
// OPTION NOLOG,NOSYSDUMP,JCANCEL
// DLBL OCS,¢VSE.TCPOCS.LIBRARY¢,,VSAM,CAT=IJSYSCT
// LIBDEF *,SEARCH=(TCPOCS.FTPC)
// EXEC FTPC,SIZE=(FTPC,256K)
OPEN TCPCL1
pram
jack
VSECD
DEFAULTS MODE LIBR LIB TCPOCS
VSECD .USR1
PWD
MKDIR C:\TCPIP\FTPC
RMDIR C:\TCPIP\FTPC
MKVSEDIR TCPOCS.FTPB
RMVSEDIR TCPOCS.FTPB
STATUS
LDIR
BYE
/*
/&
* $$ EOJ
```

Figure 190. VSE/ESA OC/FTP Client Batch Job for OS/2 2.1 Directory Operations

```
// JOB FTPCBAT                          DATE 07/10/96,CLOCK 14/34/52
// OPTION NOLOG,NOSYSDUMP,JCANCEL
 OpenConnect Systems, VSE FTP Client V2.R2.M4.T1(206)     Task=21

 Cpu id 70001 is registered as primary.

 Cpu id 70001 is on trial until 12/30/96.

 ftp> ACF verification for ¢unknwn¢ passed

 ftp> Initializing LIBR     local directory, press ¢enter¢ to continue

 ftp> LIBR     path now: TCPOCS.USER     VSE.TCPOCS.LIBRARY

 ftp> OPEN TCPCL1

 Attempting to establish a connection with domain name server :
 TCPCL1.ITSC.IBM.COM (192.61.100.83 )

 Host name resolved through domain name server:
 TCPCL1.ITSC.IBM.COM (192.61.100.83 )

 TCP Trying 192.61.100.83
 Vtam connection established to Lu/Port IPFT2S9B
 Connected to TCPCL1.ITSC.IBM.COM

 220 tcpcl1 IBM TCP/IP for OS/2 - FTP Server ver 12:58:07 on Mar 16 1994
 Name:
 ftp> pram

 Password:
 331 Password required for pram.
 230 User pram logged in.
 ftp> ID=pram     IP=192.61.100.83   LU=IPFT2S9B

 ftp> VSECD
 You now have no host LIBR     working directory

 ftp> DEFAULTS MODE LIBR LIB TCPOCS
 Defaults command completed successfully

 ftp> VSECD .USR1
 ftp> LIBR     path now: TCPOCS.USR1     VSE.TCPOCS.LIBRARY

 ftp> PWD
 257 †C:\† is current directory.

 ftp> MKDIR C:\TCPIP\FTPC
 257 MKD command successful.

 ftp> RMDIR C:\TCPIP\FTPC
 250 RMD command successful.

 ftp> MKVSEDIR TCPOCS.FTPB
 ¢TCPOCS.FTPB¢ definition successful
```

Figure 191 (Part 1 of 2). Output from OC/FTP Client Batch Job for OS/2 2.1 Directory Operations

```
ftp> RMVSEDIR TCPOCS.FTPB
¢TCPOCS.FTPB¢ removal successful

ftp> STATUS
Connected to TCPCL1.ITSC.IBM.COM
Vse execution - Partition(BG) - Task(21)
Type: ASCII ; Structure: FILE    ; Verbose: ON
Prompting: ON ; Debugging: OFF; Tracing: OFF
Current VSE working directory: TCPOCS.USR1
Current Domain name server default name suffix: ITSC.IBM.COM

ftp> LDIR
TEST001  B        96-07-21 11:23.46                     0001133
TEST     L        96-07-07 09:38.03 96-07-11 13:55.30 0000012
TEST1    L        96-07-07 09:45.24 96-07-11 13:55.50 0001133
TEST     LB       96-07-21 11:23.50                     0000100
TEST     LL       96-07-21 11:23.29                     0000100
TEST     PC       96-07-11 13:55.56                     0000012
CONFIG   TCP      96-07-04 14:35.55                     0000094
PPPPP    TCPIP    96-07-11 13:54.27 96-07-12 08:26.59 0001133
PPPPPP   TCPIP    96-07-12 08:27.37                     0001133
PPPPP22  TCPIP    96-07-12 08:28.35                     0001133
P2       TCPIP    96-07-12 08:26.03 96-07-10 12:56.33 0001133
CONF928  TEXT     96-07-28 10:59.35                     0000068
 directory request successful

ftp> BYE

ftp> ID=pram     IP=192.61.100.83   LU=IPFT2S9B
221 Goodbye.
```

Figure 191 (Part 2 of 2). Output from OC/FTP Client Batch Job for OS/2 2.1 Directory Operations

19.3 Online Operation Samples

Before FTP commands can be entered interactively via the ′ftp′ transaction, the OC/FTP Client must have been started in a dynamic partition. The startup job shown in Figure 67 on page 104 can be used to do this, Figure 192 shows the console log after startup.

```
Y2 0047 // JOB FTPC
        DATE 07/11/96,CLOCK 11/11/23
Y2 0047 *  FTPC EXECUTION IN ONLINE MODE
Y2 0047 OCS000 INITIALIZATION IN PROGRESS
Y2 0047 OCS000 INITIALIZATION IS COMPLETE
Y2 0047 OCS003 PHASE FTPC     ATTACHED AS SUBTASK FTPC0001
Y2 0090 ftp> ONLINE  initialization started
Y2 0090 ftp> ACF verification for STEF passed
Y2 0090 ftp> ID=pram     IP=192.61.100.81   LU=IPFT2S9B
Y2 0047 OCS010 DETACH COMPLETE FOR SUBTASK FTPC0001
```

Figure 192. OC/FTP Client Startup Sample

The commands shown in Table 13 can be passed to OC/FTP Client from the VSE console, using the ′msg′ operator command:

msg *fx*,**data=***cmd*

where *fx* is the partition name and *cmd* is one of the commands shown in Table 13.

Command	**Explanation**
status	displays information about OC/FTP Client such as user names, subtask
quiesce	before shutting down the OC/FTP Client the operator lets current users complete their file transfer operation and disables further logins.
stop	shuts down immediately the OC/FTP Client, thereby forcing off all current users.

Table 13. OC/FTP Client Job Commands

The following two examples show multiple file transfer operations to/from an AIX node.

19.3.1 Obtaining Multiple Files from AIX

The examples shown in Figure 193 on page 264 and Figure 194 on page 265 demonstrate the usage of the DEFAULT and MGET commands affecting the Librarian subsystem.

Note

When this example was created, our OC/FTP Server used port number 2071 and the AIX FTP Server used TCP/IP standard FTP port 21. Therefore, when no port number is specified in the FTP Open subcommand, the AIX FTP server is addressed. Also, the resolver file was not created in AIX to use the Domain Name system.

```
ftp> ACF verification for PRAM passed
OpenConnect Systems, VSE FTP Client V2.R2.M4.T1(206)     Task=5A


Cpu id 70001 is registered as primary.

Cpu id 70001 is on trial until 12/30/96.


ftp> Initializing LIBR     local directory, press ¢enter¢ to continue

ftp> LIBR   path now: TCPOCS.USER      VSE.TCPOCS.Library
ftp>
ftp> default list


Vsamcat      => TCPCAT   Palloc       => 00000256 Restart      => OFF
Salloc       => 00000256 Lrecl        => 00000080 logging      => OFF
Recfm        => RECORD   Cisizedata   => 00002048
Bufsp        => 00000000 Bufndata     => 00000010
Opndsp       => NEW      Clsdsp       => KEEP
Retain       => 00000007 Mode         => LIBR
Lib          => TCPOCS   Slib         => USR1
Mbrn         => FTPOBJ   Mbrt         => TCPIP
Bufnindex    => 00000004 Blkfactor    => 00000001
Keylocation  => 00000000 Keylength    => 00000010
Share        => 00000002 Units        => REC
Cafreespace  => 00000000 Cifreespace  => 00000000
Cisizeindex  => 00000000 Skipcount    => 00000000
Recordcount  => 00000000 HALT         => OFF
Dsn          => OCS.FTP.TESTING
Modelentry   => NULL
Lowkey       =>
Highkey      =>

ftp>
ftp> default slib usr6 mbrt z dsn ftp

Defaults command completed successfully
ftp>
ftp> default list


Vsamcat      => TCPCAT   Palloc       => 00000256 Restart      => OFF
Salloc       => 00000256 Lrecl        => 00000080 logging      => OFF
Recfm        => RECORD   Cisizedata   => 00002048
Bufsp        => 00000000 Bufndata     => 00000010
Opndsp       => NEW      Clsdsp       => KEEP
Retain       => 00000007 Mode         => LIBR
Lib          => TCPOCS   Slib         => USR6
Mbrn         => FTPOBJ   Mbrt         => Z
Bufnindex    => 00000004 Blkfactor    => 00000001
Keylocation  => 00000000 Keylength    => 00000010
Share        => 00000002 Units        => REC
Cafreespace  => 00000000 Cifreespace  => 00000000
Cisizeindex  => 00000000 Skipcount    => 00000000
Recordcount  => 00000000 HALT         => OFF
Dsn          => FTP
Modelentry   => NULL
Lowkey       =>
Highkey      =>
```

Figure 193. DEFAULT Command Sample

```
ftp> open aix320

TCP Trying 192.61.100.81
Vtam connection established to Lu/Port IPFT2S9B
Connected to AIX320

220 aix320 FTP server (Version 4.9 Thu Sep 02 20:35:07 CDT 1993) ready.
Name:
ftp> pram

Enter FTP Password>

ftp> pram


331 Password required for pram.
230 User pram logged in.

ftp> pwd

257 †/home/pram† is current directory.

ftp> pvsedir

ftp> LIBR     path is: TCPOCS.USR1     VSE.TCPOCS.LIBRARY

ftp> ls

200 PORT command successful.
150 Opening data connection for /bin/ls.
OCS.TEST.KSDS
OCS.TEST2.KSDS
aixdata
info
mbox
ocs.test.ksds
ocs.test.vsamsam
ocstst1
ocstst2
prereq
smit.log
smit.script
test
test2
test3
226 Transfer complete.
154 BYTES received in 0 SECONDS (0.00 KBYTES/S)
ftp> mget ocs*.*


Mget: ocs.test.ksds        invalid name, new name OCSTESTK.Z
               retrieve(y/n/r/q)?
ftp> y

200 PORT command successful.
150 Opening data connection for ocs.test.ksds (2025 bytes).
226 Transfer complete.
2050 BYTES received in 0 SECONDS (0.00 KBYTES/S)
Mget: ocs.test.vsamsam     invalid name, new name OCSTESTV.Z
               retrieve(y/n/r/q)?

ftp> y
```

Figure 194 (Part 1 of 2). MGET Command Sample

```
200 PORT command successful.
150 Opening data connection for ocs.test.vsamsam (68 bytes).
226 Transfer complete.
72 BYTES received in 0 SECONDS (0.00 KBYTES/S)

ftp> mget a*
Mget: aixdata              invalid name, new name AIXDATA.Z
               retrieve(y/n/r/q)?

ftp> r

Mget: aixdata              invalid name, new name

ftp> aix.data

200 PORT command successful.
150 Opening data connection for aixdata (2025 bytes).
226 Transfer complete.
2050 BYTES received in 0 SECONDS (0.00 KBYTES/S)

ftp> mget t*

Mget: test                 invalid name, new name TEST.Z
               retrieve(y/n/r/q)?

ftp> n

Mget: test.a               retrieve(y/n/q)?

ftp> y
200 PORT command successful.
150 Opening data connection for test.a (24 bytes).
226 Transfer complete.
27 BYTES received in 0 SECONDS (0.00 KBYTES/S)
Mget: test2                invalid name, new name TEST2.Z
               retrieve(y/n/r/q)?

ftp> q
```

Figure 194 (Part 2 of 2). MGET Command Sample

19.3.2 Transferring Multiple Files to AIX

This example shows the usage of the DEFAULT and MPUT commands affecting the VSAM subsystem.

First we set our default mode to ′VSAM′ and the default data set name (dsn) to be ′*′. The MPUT command uses dsn for searching files. For example, if dsn is ″ABC″, it searches for filenames starting with ″ABC″. In our case, all files are transferred using the ′MPUT *′ command.

In contrast to the FTP client batch operation where all FTP transfer messages appear in the corresponding job output list, the messages now are displayed on the VSE console. On the FTP console log shown below (Figure 197 on page 268) you can verify, looking at BC (Byte Count) and RC (Record Count), that all data has been transferred.

Figure 195 on page 267 shows the DEFAULT command we used, Figure 196 on page 267 the MPUT command and Figure 197 on page 268 the VSE console log.

```
ftp> default mode vsam dsn *

ftp> default list


Vsamcat      => TCPCAT   Palloc       => 00000256 Restart      => OFF
Salloc       => 00000256 Lrecl        => 00000080 logging      => OFF
Recfm        => RECORD   Cisizedata   => 00002048
Bufsp        => 00000000 Bufndata     => 00000010
Opndsp       => NEW      Clsdsp       => KEEP
Retain       => 00000007 Mode         => VSAM
Lib          => TCPOCS   Slib         => USR1
Mbrn         => FTPOBJ   Mbrt         => TCPIP
Bufnindex    => 00000004 Blkfactor    => 00000001
Keylocation  => 00000000 Keylength    => 00000010
Share        => 00000002 Units        => REC
Cafreespace  => 00000000 Cifreespace  => 00000000
Cisizeindex  => 00000000 Skipcount    => 00000000
Recordcount  => 00000000 HALT         => OFF
Dsn          => *
Modelentry   => NULL
Lowkey       =>
Highkey      =>
```

Figure 195. DEFAULT Command Sample

```
ftp> mput *


Mput: ONEMB.DAT                                    send(y/n/q)?
ftp> y
200 PORT command successful.
150 Opening data connection for onemb.dat.
226 Transfer complete.
1040457 BYTES sent in 1926 SECONDS (0.52 KBYTES/S)

Mput: TCPCAT.TEST.PC                               send(y/n/q)?
ftp> y

200 PORT command successful.
150 Opening data connection for tcpcat.test.pc.
226 Transfer complete.
1991 BYTES sent in 0 SECONDS (0.00 KBYTES/S)
Mput: TEST.FTP.CLIENT                              send(y/n/q)?
```

Figure 196. MPUT Command Sample

```
Y2 0088 ftp> ID=pram     IP=192.61.100.81   LU=IPFT2S9B
Y2 0088 ftp> ID=pram     DN=ONEMB.DAT
Y2 0088 ftp> ID=pram     MN=DAT               RQ=PUT OD=OLD CD=KEEP
Y2 0088 ftp> ID=pram     BC===1,040,457  RC======16,149
Y2 0088 ftp> ID=pram     IP=192.61.100.81   LU=IPFT2S9B
Y2 0088 ftp> ID=pram     DN=TCPCAT.TEST.PC
Y2 0088 ftp> ID=pram     MN=PC                RQ=PUT OD=OLD CD=KEEP
Y2 0088 ftp> ID=pram     BC=======1,991  RC==========67
```

Figure 197. Console Log of MPUT Command

19.3.3 Moving an AIX Data File into a VSAM KSDS

This example demonstrates how to move an AIX data file to a VSAM KSDS using the OC/FTP Client ′get′ command.

First we set the default mode to ′VSAMKSD′ and deleted our original file ″OCS.TEST.KSDS″. Then we transferred the file from AIX using FTP ′get′. The result is a new file ″OCS.TEST.KSDS″ with a keylength of 10 since the implicit define used the keylength of the default parameter. For transferring data to/from VSAM KSDSs, the keylength and keyposition have to be considered and specified.

Next we demonstrate the use of the RDK (Remove Duplicate Keys) option. In order for this parameter to work, you also need to set the Opndsp (Open Disposition) of the file to ′OLD′.

Figure 198 on page 269 shows the online example to transfer an AIX data file into a VSAM KSDS and Figure 199 on page 270 shows the VSE II (Interactive Interface) screen containing the VSAM data set information we received.

Note

When this example was created, our OC/FTP Server used port number 2071 and the AIX FTP Server used TCP/IP standard FTP port 21. Therefore, when no port number is specified in the FTP Open subcommand, the AIX FTP server is addressed. Also, the resolver file was not created in AIX to use the Domain Name system.

```
OpenConnect Systems, VSE FTP Client V2.R2.M4.T1(206)     Task=5A
Cpu id 70001 is registered as primary.
Cpu id 70001 is on trial until 12/30/96.

ftp> LIBR   path now: TCPOCS.USER      VSE.TCPOCS.Library
ftp>
ftp> default list

Vsamcat      => TCPCAT   Palloc       => 00000256 Restart      => OFF
Salloc       => 00000256 Lrecl        => 00000132 logging      => OFF
Recfm        => RECORD   Cisizedata   => 00002048
Bufsp        => 00000000 Bufndata     => 00000010
Opndsp       => NEW      Clsdsp       => KEEP
Retain       => 00000007 Mode         => LIBR
Lib          => TCPOCS   Slib         => USR1
Mbrn         => FTPOBJ   Mbrt         => TCPIP
Bufnindex    => 00000004 Blkfactor    => 00000001
Keylocation  => 00000000 Keylength    => 00000010
Share        => 00000002 Units        => REC
Cafreespace  => 00000000 Cifreespace  => 00000000
Cisizeindex  => 00000000 Skipcount    => 00000000
Recordcount  => 00000000 HALT         => OFF
Dsn          => OCS.FTP.TESTING
Modelentry   => NULL
Lowkey       =>
Highkey      =>

ftp> default mode vsamksd
Defaults command completed successfully

ftp> vsecd tcpcat
ftp> VSAMKSD  path now: TCPCAT  TCPIP.USER.CATALOG

ftp> open aix320
TCP Trying 192.61.100.81
Vtam connection established to Lu/Port IPFT2S9B
Connected to AIX320
220 aix320 FTP server (Version 4.9 Thu Sep 2 20:35:07 CDT 1993) ready.
Name:
ftp> pram
Enter FTP Password>
ftp> pram
331 Password required for pram.
230 User pram logged in.

ftp> ocs.test.ksds
¢OCS.TEST.KSDS¢ removal successful

ftp> get ocs.test.ksds
200 PORT command successful.
150 Opening data connection for ocs.test.ksds (2025 bytes).
226 Transfer complete.
2050 BYTES received in 1 SECONDS (2.00 KBYTES/S)

ftp> default opndsp old rdk on

ftp> get ocs.test.ksds
200 PORT command successful.
150 Opening data connection for ocs.test.ksds (2025 bytes).
226 Transfer complete.
2050 BYTES received in 1 SECONDS (2.00 KBYTES/S)
```

Figure 198. Transferring a File from AIX into a VSAM KSDS

```
IESFILDETA                  SHOW FILE INFORMATION

File Attributes:        Attribute Values:

FILE ID:                OCS.TEST.KSDS
FILE NAME:              *NONE*
CATALOG NAME:           TCPCAT

FILE ORGANIZATION:      2          1=Non keyed (ESDS)  3=Numbered   (RRDS)
                                   2=Keyed     (KSDS)  4=Numbered   (VRDS)
                                                       5=Sequential (SAM ESDS)

FILE ACCESS:            2          1=Multiple Read OR Single Write
                                   2=Multiple Read AND Single Write
                                   3=Multiple Read AND Write (no integrity)
                                   4=Multiple Read AND Write (with integrity)

FILE USAGE:             2          1=File is used as a Data File (NOREUSE)
                                   2=File is used as a Work File (REUSE)

CREATION DATE:             94316
EXPIRATION DATE:           00000

ALLOCATION UNIT:           2         1=Cylinder, 2=Track, 3=Block
PRIMARY ALLOCATION:        1
SECONDARY ALLOCATION:      1

AVERAGE RECORD SIZE:       132
MAXIMUM RECORD SIZE:       132

KEY LENGTH:                10
KEY POSITION:              0         Position 0 starts at the beginning

                           Data Component:             Index Component:

NUMBER OF CI SPLITS:       0                           0
NUMBER OF CA SPLITS:       0                           0
NUMBER OF PHYSICAL I/Os:   4                           4

CURRENTLY ALLOCATED SPACE: 36864          Bytes        23552          Bytes
CURRENTLY USED SPACE:      36864          Bytes        512            Bytes
AVAILABLE SPACE:           0   % free                  97    % free

CI SIZE:                   2048           Bytes        512            Bytes
NUMBER OF CIs PER CA:      18                          46
```

Figure 199. Display VSAM KSDS after FTP File Transfer

19.3.4 Transferring an AIX Data File into a VSAM SAM File

Before we transferred the file we set the default mode to 'VSAMSAM', the record format to 'FIXBLK' and deleted the original file "OCS.TEST.VSAMSAM". Then we transferred the file from AIX using FTP 'get'.

Figure 200 shows the online example of subject file transfer and Figure 201 on page 272 lists the information from the corresponding VSE II.

Note

When this example was created, our OC/FTP Server used port number 2071 and the AIX FTP Server used TCP/IP standard FTP port 21. Therefore, when no port number is specified in the FTP Open subcommand, the AIX FTP server is addressed. Also, the resolver file was not created in AIX to use the Domain Name system.

```
ftp> default mode vsamsam recfm fixblk


Defaults command completed successfully

ftp> ldel ocs.test.vsamsam


¢OCS.TEST.VSAMSAM¢ removal successful

ftp> get ocs.test.vsamsam


200 PORT command successful.
150 Opening data connection for ocs.test.vsamsam (68 bytes).
226 Transfer complete.
72 BYTES received in 0 SECONDS (0.00 KBYTES/S)

ftp> status


Connected to AIX320



Vse execution - Partition(Z1) - Task(55)
Type: ASCII ; Structure: FILE    ; Verbose: ON
Prompting: ON ; Debugging: OFF; Tracing: OFF
Current VSE working directory: TCPCAT
Current Domain name server default name suffix:

ftp> bye


221 Goodbye.
```

Figure 200. Transferring a File from AIX into a VSAM SAM

```
IESFILDETA                 SHOW FILE INFORMATION

File Attributes:        Attribute Values:

FILE ID:                OCS.TEST.VSAMSAM
FILE NAME:              OCSSAM
CATALOG NAME:           TCPCAT

FILE ORGANIZATION:      5          1=Non keyed (ESDS)  3=Numbered   (RRDS)
                                   2=Keyed     (KSDS)  4=Numbered   (VRDS)
                                                       5=Sequential (SAM ESDS)

FILE ACCESS:            2          1=Multiple Read OR Single Write
                                   2=Multiple Read AND Single Write
                                   3=Multiple Read AND Write (no integrity)
                                   4=Multiple Read AND Write (with integrity)

FILE USAGE:             2          1=File is used as a Data File (NOREUSE)
                                   2=File is used as a Work File (REUSE)

CREATION DATE:             94307
EXPIRATION DATE:           00000

ALLOCATION UNIT:           2         1=Cylinder, 2=Track, 3=Block
PRIMARY ALLOCATION:        2
SECONDARY ALLOCATION:      1
                                     1=Fixed unblocked,    2=Fixed blocked,
RECORD FORMAT:             2         3=Variable unblocked, 4=Variable blocked,
                                     5=Undefined, 6=No control interval format
RECORD SIZE:               132
BLOCK SIZE:                132

NUMBER OF CI SPLITS:       0
NUMBER OF CA SPLITS:       0
NUMBER OF PHYSICAL I/Os:   0

CURRENTLY ALLOCATED SPACE: 73728          Bytes
CURRENTLY USED SPACE:      2048           Bytes
AVAILABLE SPACE:           97  % free

CI SIZE:                   2048           Bytes
NUMBER OF CIs PER CA:      18
```

Figure 201. Display VSAM SAM after FTP File Transfer

Chapter 20. OC/Line Printer Daemon Operation and Examples

This chapter:

- provides instructions on how to start and operate the OC/Line Printer Daemon at the VSE host side.
- explains how to check the status of OC/Line Printer Daemon print queues from TCP/IP nodes.
- explains how to print files from TCP/IP nodes on the VSE host.
- gives examples of Line Printer operations that we tested in our environment.

20.1 Operation

This section describes how the OC/Line Printer Daemon is started and provides an overview of the commands available at both, the client and server side.

20.1.1 VSE Host (Line Printer Server)

To start OC/Line Printer Daemon, the job described in Figure 75 on page 117 is used. Figure 202 below shows a sample console log after having started the OC/Line Printer Daemon.

```
Y3 0048 // JOB LPD START LINE PRINTER DAEMON
        DATE 07/17/96,CLOCK 16/20/19
Y3 0048 OpenConnect Systems, LPD Server V1.R1.M0.T0(8)      Ready for client.
```

Figure 202. OC/Line Printer Daemon Startup Sample

To stop the OC/Line Printer Daemon, cancel the partition in which it is running using the following command at the VSE console:

cancel *fx*,**nodump**

where *fx* is the partition name that is running OC/Line Printer Daemon.

Figure 203 below shows a sample console log after cancelling the partition.

```
cancel y3,nodump
AR 0015 1I40I  READY
Y3 0048 0S01I THE OPERATOR CANCELED THE JOB
Y3 0048 0S00I JOB LPD      CANCELED
Y3 0048 0S07I PROBLEM PROGRAM  PSW = 07DD0000 00434A9C
F3 0003 IST804I CLOSE IN PROGRESS FOR OCSSMP01 OPENED BY LPD
F3 0003 IST400I TERMINATION IN PROGRESS FOR APPLID OCSSMP01
F3 0003 IST805I VTAM CLOSE COMPLETE FOR OCSSMP01
```

Figure 203. Stopping OC/Line Printer Daemon by Cancelling the Partition

Note

There is no other possibility to stop OC/Line Printer Daemon operation except by cancelling the partition in which it is running.

20.1.2 Displaying the Status of OC/Line Printer Daemon Queues

To check the status of the OC/Line Printer Daemon Queues, use the ′lpstat′ command on the OCS II Gateway.

Figure 204 shows a sample response to the lpstat command

```
root@aix320: /etc >lpstat
Queue   Dev   Status    Job Files              User         PP %   Blks  Cp Rnk
------- ----- --------- --- ------------------ ---------- ---- -- ----- --- ---
lp0     lp0   DOWN
bsh     bshde READY
vseprtm vsepr READY
vseprtm: Queue(VSEPRTM ) Type(Power   ) Ready. H=Y, M=Y, C=A, F=STD
lpdfile lpdfi READY
lpdfile: Queue(LPDFILE ) Type(DSN     ) Ready. H=Y, M=Y, C=A, F=STD
sysout  sysou READY
sysout: Queue(SYSOUT  ) Type(Power   ) Ready. H=N, M=Y, C=A, F=STD
vseprt  vsepr READY
vseprt: Queue(VSEPRT  ) Type(Power   ) Ready. H=Y, M=N, C=A, F=STD
root@aix320: /etc >
```

Figure 204. lpstat Sample Output

lpstat returns the status of all printer queues defined to AIX. The sample output contains the local queues lp0 and bsh and the OC/Line Printer Daemon queues vseprtm, lpdfile, sysout, and vseprt as described in 8.2.2.1, “LPDCONF Member” on page 112.

For these remote queues the lpstat output shows two entries:

- **vseprtm** shows the status of the local AIX queue.

  Note

  If the status of this local queue is **DOWN**, the status of the remote queue will not be displayed and no print requests will be forwarded to OC/Line Printer Daemon.

  To correct this, the local queue must be started. This can be done by using the following SMIT command sequence:

  - Spooler (Print Jobs)
  - Manage ***Local*** Printer Subsystem
  - Local Printer Queues
  - Start a Queue

- **vseprtm:** shows the status of the OC/Line Printer Daemon queue on VSE/ESA. Below is a short description of the status values:

  – **Type(DSN)** Output will be spooled to data set. Relates to **DDNAME=** parameter of LPDCONF.LPDPARM

- **Type(Power)** Output will be spooled to POWER LST queue. Relates to **SYSOUT=** parameter of LPDCONF.LPDPARM
- **H=** specifies if output will be proceeded with Print Banner. Relates to **H=** parameter of LPDCONF.LPDPARM
- **M=** specifies if the Line Printer requester will be notified by E-mail. Relates to **MAIL=** parameter of LPDCONF.LPDPARM
- **C=** specifies the Power LST queue class the print output will be spooled in (meaningless in combination with Type(DSN)). Relates to **SYSOUT=** parameter of LPDCONF.LPDPARM
- **F=** specifies the Power Form used for print output to this OC/Line Printer Daemon queue (meaningless in combination with Type(DSN)). Relates to **F=** parameter of LPDCONF.LPDPARM

To query the status of a particular queue, you may use the **-v** parameter of the lpstat command.

Another command to check the status of a particular queue is the standard line printer command **lpq**. For details about the parameters of this command refer to the appropriate documentation of your Line Printing Client. Figure 205 shows a sample response to the OS/2 LPQ command :

```
[C:\]lpq -l -s aix320 -p vseprtm
Queue(VSEPRTM ) Type(Power   ) Ready. H=Y, M=Y, C=A, F=STD
```

Figure 205. LPQ Sample Output

20.1.3 Printing from Remote TCP/IP Clients

To print a file from remote TCP/IP clients, use the lpr command. For details about the parameters for this command refer to the appropriate documentation of your Line Printing Client. Figure 206 shows a sample response to the OS/2 lpr command.

```
[C:\]lpr config.sys -s aix320 -p vseprt

Printing C:\CONFIG.SYS:
Trying LPD print server aix320.itsc.ibm.com(192.61.100.81), device vseprt.
Sent 5501 bytes.
The entire document was sent.
```

Figure 206. LPR Sample Output

Attention

Most LPR clients provide a -f parameter which indicates that the remote LPD should format the print output using the UNIX pr command.

If this parameter is specified on requests to the OC/Line Printer Daemon, the partition running OC/Line Printer Daemon will enter a loop with heavy I/O.

If this occurs, you have to cancel the partition and resubmit the OC/Line Printer Daemon startup job.

20.1.4 Operation Examples

20.1.4.1 Printing a File from OS/2 to a POWER Queue

The following file (containing the standard ASCII characters 32 to 126), is sent to the OC/Line Printer Daemon:

```
ASCII CHAR SET
0123456789ABCDEF

 !†#$%&¢()*+,-./  20
0123456789:;<=>?  30
@ABCDEFGHIJKLMNO  40
PQRSTUVWXYZ[|]^_  50
abcdefghijklmno  60
pqrstuvwxyz{|}~   70
```

Figure 207. Test File with Standard ASCII Characters 32-126 (X'20-X'7E).

This file is sent to OC/Line Printer Daemon using the following OS/2 command:

lpr TEST.DATA -s aix320 -p sysout

> **Note**
>
> We always got the following error message on the VSE/ESA console, when using the OS/2 lpr command:
>
> ```
>    .
> Y1 0045 LPD:00              &                 Command record ignored.
> ```
>
> *This message can be ignored.*

After completion of the lpr command the following POWER LST queue entry can be found:

```
d lst,lpd*
AR 0015 1C39I COMMAND PASSED TO VSE/POWER
F1 0001 1R46I    LIST QUEUE   P D C S  PAGES  CC FORM
F1 0001 1R46I  LPD      04562 3 D A        1   1 STD  TO=(DIRK) FROM=(DIRK)
```

The name of this entry will always be the same as the VSE/POWER job name of the OC/Line Printer Daemon startup job. This entry will display a screen similar to the following in VSE Interactive Interface:

```
ASCII CHAR SET
0123456789ABCDEF

 !†#$%&¢()*+,-./  20
0123456789:;<=>?  30
@ABCDEFGHIJKLMNO  40
PQRSTUVWXYZ•\“ _  50
abcdefghijklmno  60
pqrstuvwxyz{|}   70
```

Note

The following characters are not standard EBCDIC characters and may therefore not be displayed and printed correctly:

- [left bracket
-] right bracket
- ~ tilde
- ^ caret
- right prime

Since SENDMAIL is not started on our OS/2 system, we will not get an E-mail notification for completion of our line printer request.

20.1.4.2 Printing a File from AIX to a VSAM File

The same file as shown in Figure 207 on page 277 is sent to the OC/Line Printer Daemon using the following AIX command:

lpr -Plpdfile test.data

After completion of the lpr command the VSAM SAM ESDS ′LPDFILE′ will appear as the following:

```
                            OOOOOOO  CCCCCCC  SSSSSS     /     LL      PPPPPPPP  DDDDDDDD
                           OO   OO  CC       SS    SS   /     LL      PP    PP  DD    DD
                          OO   OO  CC         SS       /     LL      PPPPPPPP  DD    DD
                         OO   OO  CC      SS   SS     /     LL      PP        DD    DD
                        OOOOOOO  CCCCCCCC  SSSSSSS   /     LLLLLL  PP        DDDDDDDD
                    OpenConnect Systems,LPD/Server V1.R1.M0.T0(8)
*************************************************************************************************
GATEWAY    :        VSE
HOST NAME  :        aix320
DELIVER TO :        root
JOB NAME   :        test.data
LOGON ID   :        root
DATE QUEUED:        01:10:30, Fri Jul 19, 1996
COPIES     :        001
*************************************************************************************************
                            OOOOOOO  CCCCCCC  SSSSSS     /     LL      PPPPPPPP  DDDDDDDD
                           OO   OO  CC       SS    SS   /     LL      PP    PP  DD    DD
                          OO   OO  CC         SS       /     LL      PPPPPPPP  DD    DD
                         OO   OO  CC      SS   SS     /     LL      PP        DD    DD
                        OOOOOOO  CCCCCCCC  SSSSSSS   /     LLLLLL  PP        DDDDDDDD
ASCII CHAR SET
0123456789ABCDEF

 !†#$%&¢()*+,-./  20
0123456789:;<=>?  30
@ABCDEFGHIJKLMNO  40
PQRSTUVWXYZ•\“ _  50
abcdefghijklmno  60
pqrstuvwxyz{|}   70
```

Figure 208. VSAM Print Output of LPDFILE

Since the mailer daemon is started on AIX, we will get the following E-mail notification.

```
From lpd@vse Thu Jul 18 08:49:50 1996
Date: Thu, 18 Jul 1996 08:49:46 +0100
From: lpd@vse
Subject: Your print.
Apparently-To: root@aix320

Your print has been routed to the queue
you selected.

Thank you for using OpenConnect Systems products.
```

Figure 209. E-mail Notification from OC/Line Printer Daemon

Chapter 21. OC/RSH Client Operation and Examples

The OC/RSH Client is invoked in the VSE environment via a CICS transaction or a VSE batch job.

Before executing commands on remote TCP/IP nodes with the OC/RSH Client you need to make sure that:

- There is an active RSH server on the remote node
- You are authorized for remote command execution on the remote node

The OC/RSH Client program is designed to invoke programs on remote TCP/IP hosts which obtain their input from commands and write results to the standard output device for commands, that is, the terminal. This means that you cannot invoke interactive programs such as the **vi** editor under AIX.

There are two ways to initiate remote command execution from the VSE host:

1. **Interactively using CICS transaction ′rsh′**

   This requires that the OC/RSH Client program has been started and is running in a dynamic partition. CICS ISC is used to pass all interactive RSH commands to the OC/RSH Client program (see Figure 9 on page 25).

2. **Using a batch job**

   This requires that all RSH commands have to be imbedded in the job. As shown in Figure 79 on page 126 the OC/RSH Client program then manages, together with VTAM, all communications to the OCS II Gateway.

21.1 OC/RSH Client Online Operation Examples

Before RSH commands can be entered interactively via the ′rsh′ transaction, OC/RSH Client must have been started in a dynamic partition. Since the RSH functions have been implemented in the OC/FTP Client program, we use its startup job shown in Figure 67 on page 104. Figure 210 shows the VSE console log after OC/FTP Client startup.

```
Y2 0047 // JOB FTPC
        DATE 07/11/96,CLOCK 11/11/23
Y2 0047 *  START FTP AND RSH CLIENT
Y2 0047 OCS000 INITIALIZATION IN PROGRESS
Y2 0047 OCS000 INITIALIZATION IS COMPLETE
```

Figure 210. OC/RSH Client Startup Sample

The commands shown in Table 14 can be passed to the OC/RSH Client from the VSE console, using the ′msg′ operator command:

msg *fx***,data=***cmd*

where *fx* is the partition name and *cmd* is one of the commands shown in Table 13 on page 262. The commands are the same as for the OC/FTP Client program.

Command	Explanation
status	displays information about OC/RSH Client such as user names, subtask.
quiesce	before shutting down the OC/RSH Client the operator lets current users complete their command execution.
stop	shuts down immediately the OC/RSH Client, thereby forcing off all current users.

Table 14. OC/RSH Client Job Commands

Figure 211 on page 283 and Figure 212 on page 283 demonstrate the usage of the OC/RSH Client.

```
RSH/Client for VSE.  Version 2 Release 1.5.1
Default Userid: WACKER   enter alternate replacement, for change

rsh> parmlib TCPOCS.FTPC

Parmlib is now set to: TCPOCS.FTPC

rsh> rsh aix320 -l wacker -n echo $HOME; ls -l

/u/wacker
total 40
-rwxr-----   1 wacker  staff        254 Sep 30 17:35 .profile
-rw-r--r--   1 wacker  system       482 Oct  6 17:29 birthday
-rw-r--r--   1 wacker  system        36 Sep 30 16:02 cde
-rw-r--r--   1 wacker  system        40 Oct  6 12:07 ls.file
-rw-r--r--   1 root    system      1268 Oct  7 16:10 masterplan
drwxr-xr-x   2 wacker  system       512 Sep 30 15:30 puss.1
-rw-r--r--   1 wacker  system        19 Sep 30 15:08 wacktest
-rw-r--r--   1 wacker  system    280436 Oct  6 17:21 tcptrace
```

Figure 211. List AIX Directory Using OC/RSH Client

After having entered the 'rsh' transaction, the program displays the default user ID to be used for communication with the remote RSH server. Press enter to accept that user ID, then enter the RSH 'parmlib' command to display the default library where the OC/FTP Client administration files reside (refer to 7.2.4, "FTP Client Customization" on page 92).

Then the remote command can be typed in and sent to the RSH server ('aix320' in our case) as shown in the example below.

```
rsh> aix320 -l wacker -n grep Nov birthday

Nov 03          James McMiller
Nov 12          Mary
Nov 17          Steve
Nov 25          Connie
Nov 30          Iren

rsh> aix320 -l wacker -n rm masterplan

No data was returned from the TCP/IP host for the command entered.

rsh> bye
```

Figure 212. File Manipulation Sample

21.2 Batch Operation Samples

This section contains a batch job and its corresponding output listing for remote shell command execution.

21.2.1 RSH Batch Example

Figure 213 shows the job we used to do remote command execution and Figure 214 shows the output produced by the job.

```
* $$ JOB JNM=RSHBATCH,DISP=D,PRI=3,                                    C
* $$ NTFY=YES,                                                         C
* $$ LDEST=(,RSCS),                                                    C
* $$ CLASS=Y
* $$ LST CLASS=Z
// JOB RSHBATCH
// OPTION NOLOG,SYSDUMP,JCANCEL
// LIBDEF *,SEARCH=TCPOCS.FTPC,TEMP
// EXEC PGM=RSHC,SIZE=(RSHC,256K)
PARMLIB TCPOCS.FTPC
RSH AIX320 -l wacker -n date; echo $HOME; ls -l
RSH AIX320 -l wacker -n cat /etc/hosts
RSH AIX320 -l maru -n ls -l   grep Oct
RSH 192.61.100.81 -l maru -n ls -l   grep Oct
BYE
/*
/&
* $$ EOJ
```

Figure 213. RSH Batch Example

```
// JOB RSHBATCH          DATE 07/10/96,CLOCK 11/04/26
// OPTION NOLOG,NOSYSDUMP,JCANCEL
RSH/Client for VSE.  Version 2 Release 1.5.1
PARMLIB TCPOCS.FTPC
Parmlib is now set to: TCPOCS.FTPC

RSH AIX320 -l wack -n date; echo $HOME; ls -l
Wed Jul 10 12:00:03 CDT 1996
/u/wack
total 40
-rwxr-----   1 wacker  staff        254 Sep 30 17:35 .profile
-rw-r--r--   1 wacker  system       482 Oct  6 17:29 birthday
-rw-r--r--   1 wacker  system        36 Sep 30 16:02 cde
-rw-r--r--   1 wacker  system        40 Oct  6 12:07 ls.file
-rw-r--r--   1 root    system      1268 Oct  7 16:10 masterplan
drwxr-xr-x   2 wacker  system       512 Sep 30 15:30 puss.1
-rw-r--r--   1 wacker  system        19 Sep 30 15:08 wacktest
-rw-r--r--   1 wacker  system    280436 Oct  6 17:21 tcptrace
```

Figure 214 (Part 1 of 3). RSH Batch Example Output Listing

```
RSH AIX320 -l wacker -n cat /etc/hosts
# looback network (for testing on a local site)
#
127.0.0.1               localhost
#
#   COMPONENT NAME: cmd_tcpip
#   FUNCTIONS: none
#   Licensed Materials - Property of IBM
#   This Module is †Restricted Materials of IBM†
#   Product#5756-112 (C) Copyright IBM Corp. 1992
#   See Copyright Instruction (Pamphlet G1202083)
#
9.164.182.11    mega1  t1mega1   # Added by BOS Install
#
# AIXESA hosts via ESCON
25.0.0.1  emega1 esa1 escon1
25.0.0.2  emega2 esa2 escon2                    # Added by BOS Install
192.61.100.83 tcpcl1 #ps/2 client johnny
192.61.100.84 tcpcl2 #ps/2 client francesco
192.61.100.82 tcpgtw2 #ps/2 server ip/2 yuki
192.61.100.14 9221ctr
192.61.200.14 9221c
192.61.200.60 mac
192.61.200.91 sun2
192.61.200.55 rs6tec
192.61.100.55 rs6tectr
192.61.100.54 rs6tdbtr
192.61.100.81 aix320
111.111.111.111 aix320
```

Figure 214 (Part 2 of 3). RSH Batch Example Output Listing

```
RSH AIX320 -l maru -n ls -l   grep Oct
-rw-r--r--   1 root     system         0 Oct  6 12:21 news.data
RSH 192.61.100.81 -l maru -n ls -l   grep Oct
-rw-r--r--   1 root     system         0 Oct  6 12:21 news.data
BYE
LISTLOG UTILITY
COUNT     MESSAGE TEXT          CLOCK    DATE  PAGE      1
    1  BG 000 // JOB RSHBATCH                  11:04:26 07/10/96
    2     DATE 07/10/96,CLOCK 11/04/26         11:04:26 07/10/96
   END OF LISTLOG UTILITY
```

Figure 214 (Part 3 of 3). RSH Batch Example Output Listing

Chapter 22. OC/TELNET FS Operation and Examples

As explained in chapter 3.6, “OC/TELNET FS Functional Overview” on page 26 and illustrated in Figure 10 on page 27, OC/TELNET FS runs in a dynamic partition as a VTAM application. This means that OC/TELNET FS must have been started before you can log on to a TCP/IP Telnet Server. Figure 215 shows the console log of the OC/TELNET FS startup job.

```
Y1 0046 // JOB TNFS   START OC/TELNET CLIENT FULL SCREEN
        DATE 07/05/96,CLOCK 14/11/05
Y1 0046 TAI000I OC/TELNET FS VERSION 4.1.1 (12/19/94) IS STARTING
Y1 0046 TAI007I *** NOTE - PROGRAM NOW EXECUTING ON CPU 70001
Y1 0046 TAI008I *** NOTE - PROGRAM LOADED AT ADDRESS 00420078
Y1 0046 TAI220I *** NOTE - MAJOR ACBNAME IS VSTELNET
Y1 0046 TAI222I *** NOTE - MINOR ACBNAMES ARE VSNETNNN
Y1 0046 TAI000I OC/TELNET FS VERSION 4.1.1 (12/19/94) IS ACTIVE
Y1 0046 TAI000I USE MSG COMMAND TO COMMUNICATE WITH THIS PROGRAM
Y1 0046 TAI300I D08001   LOGON      IN PROGRESS
Y1 0046 TAI302I D08001   LOGON      ACCEPTED (VSNET001/00000000) - NUM
USERS IS 1
Y1 0090 TAI000I D08001   CONNECTED TO       LU IPFT2S9J HOST 192.61.10
PORT 23
Y1 0046 PARTITION WAITING FOR CONSOLE INPUT - REPLY STOP TO TERMINATE

Y1-0046
46 STOP
Y1 0046 TAI000I STOP COMMAND ACCEPTED
Y1 0046 EOJ TNFS
```

Figure 215. OC/TELNET FS Startup Job Sample

The commands shown in chapters 26 and 27 of *OpenConnect/TELNET Client Full Screen User and Administration Guide, 350-0145-101* can be passed to OC/TELNET FS from the VSE console using the ’msg’ operator command:

msg *fx*

where *fx* is the partition name. After you receive a response from the partition, type the command as shown in the sample Figure 215 above.

22.1 Operation Samples

To use the TELNET functions in order to log on to a remote TCP/IP node, you need first to log on to the appropriate VTAM application (VSTELNET) which connects your terminal to the OCS II Gateway (refer to section 3.6, “OC/TELNET FS Functional Overview” on page 26 and chapter Chapter 10, “OC/TELNET FS Installation and Customization” on page 133 for details):

logon applid(vstelnet)

22.2 Working with Telnet Panels

After connection with OC/TELNET FS has been established, the TELNET Primary Option Menu shown in Figure 216 is presented.

The Primary Option Menu lets you define your terminal environment and overwrite the default values specified in the DFLTUSR file (refer section 10.2.2, "TELNET Client Customization" on page 137).
Figure 216 to Figure 219 on page 292 show the options available for terminal, translation table and keyboard definition.

```
------------------------  TELNET PRIMARY OPTION MENU  -------------------------
OPTION ===>
                                                             TERMINAL - D0810016
   1  SETUP      - Specify terminal setup options             APPL ID - VSNET001
   2  TRANSLATE  - Specify translation table information
   3  KEYBOARD   - Specify keyboard definitions
   O  OPEN       - Connect to another host within a TCP/IP Network
   C  CLOSE      - Terminate the current connection
   T  TERMINAL   - Begin terminal mode
   Q  QUIT       - Terminate TELNET session

Enter QUIT command to terminate your session with TELNET.




                     OC/Telnet FS Version 4.1.1 (12/19/94)
  Copyright (c) 1984-1994 by Teubner & Associates, Inc., All Rights Reserved
                   Distributed by OpenConnect Systems, Inc.



PF 1=         2=          3=QUIT      4=          5=          6=
PF 7=         8=          9=         10=         11=         12=QUIT
```

Figure 216. TELNET Primary Menu

- **1 SETUP.** The terminal setup menu (Figure 217 on page 289) is displayed
- **2 TRANSLATE.** The translation table setup menu (Figure 218 on page 291) is displayed
- **3 KEYBOARD.** The keyboard setup menu (Figure 219 on page 292) is displayed
- **O OPEN.** The host selection panel (Figure 220 on page 293) is displayed
- **C CLOSE.** The current session with your Telnet partner is closed
- **T TERMINAL.** After a TN connection is established, you can enter remote host commands as if your terminal were attached to that host. The remote host welcome menu is displayed as a result of the TN connection. An example of connecting to an AIX node is shown in Figure 220 on page 293
- **Q QUIT.** The session with OC/TELNET FS is closed

```
------------------------  TERMINAL SETUP OPTIONS PANEL  -----------------------
  ENTER/VERIFY OPTIONS BELOW:
    TERMINAL TYPE                ===> VT100      (LINE,PAGE,VT100,VT200)
    KEYBOARD PROFILE             ===> VT100      (DEFAULT IS TERMINAL TYPE)
    DISPLAY MODE                 ===> BLOCK      (BLOCK,PAGE,LINE,CONT)
    LOCAL ECHO                   ===> NO         (YES,NO)
    NEW LINE                     ===> YES        (YES,NO)
    AUTO WRAP                    ===> NO         (YES,NO)
    SHIFT LOCK                   ===> NO         (YES,NO)
    COMMAND PREFIX               ===>              (SINGLE CHAR. OR FIELD MARK)
    AUTO KEYBOARD UNLOCK         ===> DATA       (YES,NO,DATA)
    VERTICAL CURSOR MOVEMENT     ===> DATA       (NEW,LEFT,DATA)
    HORIZONTAL CURSOR MOVEMENT   ===> DATA       (NEW,DATA)
    CURSOR MOVEMENT SEQUENCE     ===> SINGLE     (SINGLE,EDT,VI)



  SPECIFY HORIZONTAL TAB STOPS BELOW:
         1         2         3         4         5         6         7         8
234567890123456789012345678901234567890123456789012345678901234567890123456789 0
       T       T       T       T       T       T       T       T       T

PF 1=          2=          3=END       4=          5=          6=
PF 7=          8=          9=         10=         11=         12=END
```

Figure 217. Terminal Setup Sample

- **TERMINAL TYPE.** You can choose the desired terminal emulation. The allowed values are shown on the right side of the screen
- **KEYBOARD PROFILE.** You can choose the desired keyboard mapping. The value selected must match one of the names in the VSAM **TEST.PARMLIB.***type* data set, where *type* is the keyboard profile name (see Figure 97 on page 143)
- **DISPLAY MODE.** The value selected depends on the application program you want to use at the remote TCP/IP node. If it is a full-screen application, BLOCK is the best choice
- **LOCAL ECHO.** If you choose YES, the OC/TELNET FS will display input data at your terminal
- **NEW LINE.** Defines the meaning of the <NL> control code sent on the remote system:
  - **YES.** OC/TELNET FS will send the ASCII carriage return and line feed character
  - **NO.** OC/TELNET FS will send only the ASCII carriage return character
- **AUTOWRAP.** You can enable the autowrap facility on your terminal
- **SHIFT LOCK.** Determines the automatic translation from lower case to upper case when transmitting data to the remote system
- **COMMAND PREFIX.** Defines the prefix character of input data that the OC/TELNET FS has to interpret as a terminal mode command
- **AUTO KEYBOARD UNLOCK.** Specifies whether the system will automatically unlock your keyboard after an input operation
- **VERTICAL CURSOR MOVEMENT.** Defines the vertical cursor movement to be used:

- **NEW.** Cursor will be moved to the column corresponding to the new cursor position
- **DATA.** Cursor will be moved to the column on the new row nearest the new cursor position that contains some data
- **LEFT.** Cursor will always be moved to column one

• **HORIZONTAL CURSOR MOVEMENT.** Defines the horizontal cursor movement to be used:

- **NEW.** Cursor will be moved to the column corresponding to the new cursor position
- **DATA.** Cursor will be moved to the column on the new row nearest the new cursor position that contains some data

• **CURSOR MOVEMENT SEQUENCE.** This value depends on the editor you want to use on the remote system:

- **SINGLE.** Normal cursor movement
- **EDT.** Cursor will have to interact with DEC′s EDT editor
- **VI.** Cursor will have to interact with a VI editor on a UNIX system (for example, AIX or SUN)

```
----------------------- TRANSLATION TABLE INFORMATION -----------------------
        TRANSLATION TABLE  ===> ASCII           FIRST ENTRY  ===> 00

             DEFINITION       ASCII       EBCDIC       DEFINITION
        --------------------  ------      ------  --------------------
                         NUL      00  =>  00      NUL
                         SOH      01  =>  01      SOH
                         STX      02  =>  02      STX
                         ETX      03  =>  03      ETX
                         EOT      04  =>  37      EOT
                         ENQ      05  =>  2D      ENQ
                         ACK      06  =>  2E      ACK
                         BEL      07  =>  2F      BEL
                          BS      08  =>  16      BS
                          HT      09  =>  05      HT
                          LF      0A  =>  25      LF
                          VT      0B  =>  0B      VT
                          FF      0C  =>  0C      FF
                          CR      0D  =>  0D      CR
                          SO      0E  =>  0E      SO
                          SI      0F  =>  0F      SI

PF 1=          2=          3=END       4=          5=          6=
PF 7=UP        8=DOWN      9=         10=         11=         12=END
```

Figure 218. Translation Table Sample

This panel lets you modify the active translation table simply by typing over the EBCDIC predefined values. The translation between ASCII and EBCDIC characters is made automatically by OC/TELNET FS.

```
-------------------------  KEYBOARD DEFINITION PANEL  ----------------------

   The following key definitions are currently active during terminal mode:

ENTER ===> SEND * <NL>
CLEAR ===> CLEAR
PA1   ===> BREAK
PA2   ===> RESHOW
PA3   ===>
PF1   ===> SEND <ESC> OP *
PF2   ===> SEND * <ESC> OQ
PF3   ===> SEND * <ESC> OR
PF4   ===> SEND * <ESC> OS
PF5   ===> SEND <ESC>
PF6   ===> CONTROL
PF7   ===> SEND * <HT>
PF8   ===> SEND ¢ ¢
PF9   ===> SEND  Z
PF10  ===> SEND <ESC> Op
PF11  ===> SEND *
PF12  ===> END

PF 1=         2=          3=END       4=          5=          6=
PF 7=UP       8=DOWN      9=         10=         11=         12=END
```

Figure 219. Keyboard Definition Sample

This panel allows for modification of the keyboard mapping chosen in the *Terminal Setup Options Panel* (see Figure 217 on page 289). Just enter the new values required and press the ′enter′ key.

Once your environment is set up according to your needs, you can initiate a TELNET connection to a TCP/IP node by specifying its IP address or hostname as defined in the HOST file (refer to 10.2.2.3, “HOSTS File” on page 138).

```
--------------------------- HOST SELECTION PANEL -------------------



       This Panel allows you to specify the name or address of the
       host system to be accessed via the TCP/IP network.  Once the
       connection has been established, you must log on to the host
       system as if your terminal were physically connected to it.


   ENTER PARAMETER BELOW:

     HOST       ===> aix320

     PORT       ===>                                  (Optional)







 PF 1=          2=          3=END       4=          5=          6=
 PF 7=          8=          9=         10=         11=         12=END
```

Figure 220. Host Selection Sample

- **HOST.** Specifies the name or the IP address (in dotted decimal format) of the remote host to be accessed.
- **PORT.** Specifies the port number of the remote TELNET server. This parameter is optional; if you omit it, OC/TELNET FS will take the default value (23).

Figure 221 shows the result of a TELNET connection to the AIX TCP/IP Telnet Server. You are now in a position to log on to AIX as if your terminal was an ASCII terminal emulating a DEC VT100 terminal (refer to the terminal setup in Figure 217 on page 289).

```
IBM AIX Version 3 for RISC System/6000
(C) Copyrights by IBM and by others 1982, 1991.
login: root
*******************************************************************************
*                                                                             *
*                                                                             *
*  Welcome to IBM AIX Version 3.2!                                            *
*                                                                             *
*                                                                             *
*  Please see the README file in /usr/lpp/bos for information pertinent to    *
*  this release of the AIX Operating System.                                  *
*                                                                             *
*                                                                             *
*******************************************************************************

Last login: Fri Jul 26 08:05:26 NFT 1996 on hft/0
root@aix320: / >
```

Figure 221. AIX Welcome Menu Sample

After having logged in as the ′root′ user, the contents of the ′/usr/oc/sna/etc′ directory are displayed as in the figure below.

```
root@aix320: /usr/oc/sna/etc >ls -l
total 560
-rwxr-xr-x   1 root     system     15910 Sep  5 1994  cfgtext
-rw-r--r--   1 root     system        42 May  8 12:45 ockeys
-rw-r--r--   1 root     system        42 May  8 12:43 ockeys.080596
-rw-r--r--   1 root     system     53654 Jul 12 09:10 octermcap
-rw-r-----   1 root     system       508 Jul 22 08:02 opt.err
-rw-r-----   1 root     system       508 Jul 31 07:26 opt.err.occ1
-rw-r-----   1 root     system       508 Jul 25 09:46 opt.err.vm_via3174
-rwxr-xr-x   1 root     system     15328 Jul 12 09:10 options
-rw-r--r--   1 root     system       444 Jul 31 07:20 options.occ1
-rw-r--r--   1 root     system       399 Oct  5 1995  options.ocs95.cfg
-rwxr-xr-x   1 root     system     11143 Sep  5 1994  options.org
-rw-r--r--   1 root     system       311 Jul 15 11:54 options.vm_via3174
-rw-r--r--   1 root     system     53591 Nov 24 1994  saveoctermcap
-rw-r--r--   1 root     system     15379 Nov 24 1994  savetstext
-rw-r--r--   1 root     system     20555 Jul 12 09:00 tstext
-rw-r--r--   1 root     system     18108 Jul 12 08:37 tstextp
```

Figure 222. OC/TELNET FS AIX Emulation Sample

Chapter 23. OCS II Telnet Server Operation and Examples

23.1 General Considerations

The OCS II Telnet Server allows you to log on to VSE from a TCP/IP terminal, thereby emulating a 3278 terminal. This emulation is strictly controlled by the **octermcap** file, usually resident in the */usr/oc/sna/etc* directory of your RISC/6000 (see 12.4.4, “System Wide Configuration Options” on page 168). This file contains some terminal profiles describing the characteristics of a number of popular ASCII terminals. If required, new entries for additional terminal types can be added to this file. For more details refer to *OCS II Telnet Server Manager, 350-0193-101*.

Before you can log on to VSE/ESA, the OCS II Telnet Server needs to be active, that is, running as a process on the RISC/6000 OCS II Gateway (please refer to Chapter 13, “OCS II Gateway Operation” on page 189).

23.2 Operation Examples

To establish connection to the VSE host, the standard TELNET command can be used:

telnet *host port*

where:

- *host* = host name or IP address of the OCS II Telnet Server. This value **must match** the IP addresses and the host names defined in AIX TCP/IP and the parameter contained in the **HOSTS** file (see Figure 91 on page 139). Or you have provided a Domain Name Server to resolve the host name.
- *port* = OCS II Telnet Server port number. This value **must match** the number specified in OCS II Telnet Server customization panel (see Figure 127 on page 171). We have used the TCP/IP standard TELNET port number (23) for the VSE TELNET. The AIX TELNET port number has been changed to (2023), please refer to 12.6, “Set up Alternative Ports for AIX TELNET and FTP Servers” on page 186 for details. Also refer to Figure 127 on page 171 for the matching value in OCS II configuration.

The figure below shows a Telnet session from a TCP/IP station to VSE using OCS II Telnet Server.

```
[C:\]telnet aix320
```

Figure 223. Example of Connecting to the OCS II Telnet Server

23.2.1 Telnet Server Welcome Panel

The panel in Figure 224 shows that OCS II Telnet Server is ready for connecting to VSE. Only the 'enter' key needs to be pressed at this stage.

```
OpenConnect (R) TELNET Server - OCS II RISC/6000, V3.7.5
******************************************************************************
*     Welcome to the ITSO version of OpenConnect Systems TELNET Server      *
*                                                                            *
*                          OpenConnect Systems                              *
*                  2711 LBJ Freeway, Dallas, Texas 75234                    *
*                                                                            *
* OpenConnect Systems offers a complete line of hardware and software       *
* products that provide connectivity between IBM Mainframe/Midrange systems  *
* and TCP/IP networks.                                                       *
*                                                                            *
* For technical assistance contact Customer Support:                        *
*     Tel: (214)888-0678  Fax: (214)888-0680  Internet: support@oc.com      *
*     BBS: (214)888-0676                                                     *
*                                                                            *
* For information about our other connectivity solutions contact Marketing:  *
*     Tel: (214)484-5200  Fax: (214)888-0688  Internet: info@oc.com         *
*                                                                            *
* NOTE:  This display screen can be modified by your Systems Administrator   *
*        to contain site specific information.  A command called ¢info¢ is   *
*        provided to redisplay this screen for subsequent viewing.          *
******************************************************************************
Press RETURN to accept command, ESC to erase command, or ? for help.
TS> ! vt220,3278
```

Figure 224. OCS II Telnet Server Welcome Panel

The prompt line **TS> !vt220,3278** has the following meaning:

- '!' is the Telnet Server execute character
- **vt220** is the negotiated terminal type. This parameter correlates to the setting of the **NEGTERM** value in OCS II Gateway customization (see Figure 127 on page 171). If this value is set to **Y** the negotiated terminal type is displayed and this entry is used to access the corresponding entry in the **octermcap** file. If you have set it to **N**, the first entry of the **octermcap** file is taken.
- **3278** is the emulation requested. This version of OCS II Telnet Server emulates only two host terminals, **3278** and **5250**.

23.2.2 Telnet Server Commands Panel

If '?' or **cmds** is entered after the welcome panel, the list of available commands as shown in Figure 225 is displayed.

```
TS>

TS execute character is ¢!¢ and erase character is ¢ESC¢.
bye | exit | quit - Exit TELNET Server Manager.
clients           - Display information about current sessions.
cmds | ?          - Display TELNET Server Manager commands (this screen).
config            - Display configuration information.
default           - Prompt with default command line.
hat               - Display Host Access Table entries.
help              - Display TELNET Server Manager help screen.
info              - Display TELNET Server Manager banner screen.
keycode           - Enter key interpret mode.
ocadmin           - Enter OCADMIN administration mode.
reboot            - Gateway remote reboot facility.
refresh           - Refresh Session Access Control configuration.
reread            - Re-read TELNET Server support files.
sac               - Display Session Access Control information.
service           - Display available services.
showterm          - Display terminal names defined in octermcap file.
spy | ¢¢          - Spy Trace facility.
status            - Display TELNET Server status.
termcap           - Display octermcap file entry or mnemonic.
trace             - OCADMIN trace facility.
```

Figure 225. OCS II Telnet Server Commands

A short description of the commands follows; for more details refer to the *OCS II Telnet Server Manager, 350-0193-101*.

- **bye.** Closes the connection with OCS II Telnet Server.
- **clients.** Gives a list of clients connected to OCS II Telnet Server with their attributes such as IP address, port number, as shown in Figure 226 on page 299.
- **config.** Reads the configuration values of the OCS II Telnet Server in the files:
  - **features.** Internal OCS II Telnet Server file. Its contents change according to the feature of the TELNET product used.
  - **sna_bin.** ASCII file created automatically when saving the values entered in the **occonfig** exec (see 12.4.8, "Save Configuration File Set" on page 183).
  - **options.** This file can be customized by just editing and changing the values inside. It contains parameters about OCS II Telnet Server and is an alternative method to customizing OCS II Telnet Server without using the occonfig utility but was not used or tested in our implementation.

  Figure 227 on page 299 shows an example.
- **default.** OCS II Telnet Server prompts the default command line (for example, TS> !vt100,3278).

- **hat.** Reads the contents of the Host Access Table file (**tstext**). This option was tested in our environment for two functions. One is to let TCPCL3 bypass the Welcome Panel, and the other is to deny access to all host except TCPCL3. The HAT table is shown in Figure 228 on page 299.
- **help.** Displays OCS II Telnet Server help about emulation service as shown in Figure 235 on page 303.
- **info.** Displays OCS II Telnet Server welcome panel.
- **keycode.** Enters the terminal in keycode mode. When you press a key, ASCII, octal, decimal and hexadecimal code are displayed. To exit and return to the normal mode just press Shift+q. Figure 229 on page 300 shows an example.
- **ocadmin.** This command is used to enter the password protected administration (ocadmin) mode. The kill, reboot,refresh, reread, spy, and trace commands are restricted for use under ocadmin mode.
- **reboot.** Reboots the OCS II Gateway from your terminal. The ocadmin mode must be set to perform the reboot.
- **refresh.** The refresh command allows to change and reread the Session Access Control (SAC) configuration file without rebooting the gateway. The ocadmin mode must be set to perform the refresh.
- **reread.** Rereads the OCS II Gateway configuration files without the necessity to reboot the client. This is useful when you change these files (for example, ctermcap) when the Telnet session is already established.
- **sac.** The current SAC groups, pools, or TN3270E terminal and printer LU associations are displayed.
- **service.** Displays the services available in OCS II Telnet Server as shown in Figure 231 on page 300.
- **showterm.** Displays the terminal type entries in the octermcap file as shown in Figure 232 on page 301.
- **spy.** SPY activates a TELNET Server trace facility. The trace facility performs debugging of communication between OC/TELNET Server and
  1. Any TELNET client operating on a TCP/IP host.
  2. OC/TN3270 terminal emulation client operating on a TCP/IP host.
- **status.** Displays the OCS II Telnet Server status as shown in Figure 233 on page 301.
- **termcap.***term* Displays the contents of the *term* entry in octermacp file as shown in Figure 234 on page 302.
- **trace.** The trace command provides an interface to the OCSNA trace facility to initiate and/or terminate an OCSNA trace from the TELNET Server Manager command line. If entered with parameter -x a panel with the code level is displayed as shown in Figure 230 on page 300. For more details on this argument please refer to *OCS II Telnet Server Manager, 350-0193-101*.

23.2.3 Display OCS II Telnet Server Clients

```
TS> clients
Service IP Address      Port# Socket# PU Name  LU Name  LU# State IT  IL  Spy
------- --------------- ----- ------- -------- -------- --- ----- --- --- ---
Tsmgr   192.61.100.83    1026       6 ******** ******** ***     1 *** *** ***
tn3270  192.61.100.81    1024       4 IPFP2209 IPFT2S9M  13     1 *** *** ***
3278    192.61.100.199   1026       7 IPFP2209 IPFT2S9N  14     1 *** *** ***
    Configured sessions:  8    Active sessions:  3
```

Figure 226. Display Clients Example

23.2.4 Display OCS II Telnet Server Configuration

```
TS> config
Service       Feature Option Config Result
------------  ------- ------ ------ ------
USERS_TELNET       16      8    ***      8
USERS_3278         16     16      3      3
USERS_5250          0      0      0      0
USERS_TN3270(E)    16      8      3      3
USERS_TN5250       16    ***      0      0
USERS_TSPASS       16    ***    ***      8

Buffer    size count  cac  hac
------  ------ ----- ---- ----
Type0      128     8    1    3
Type1      256    12    0    5
Type2     1024     4    0    1
Type3     4096     2    0    1
Type4     8188     8    1    1

Memory  cac  hac
------ ---- ----
    12    2    2
   264    1    1

Termcap RAM Disk size = 65536, used = 29329
```

Figure 227. Display Configuration Example

23.2.5 Display OCS II Telnet Server HAT Table

```
TS> hat
Name         IP address      IP mask          Port  Pmask
------------ --------------- --------------- ------ ------
tcpcl3       192.61.100.199  0xFF.FF.FF.FF        0 0x0000
allhosts     0.0.0.0         0.0.0.0              0      0
```

Figure 228. Display HAT Table Example

23.2.6 Display OCS II Telnet Server Keycodes

```
TS> keycode
You entered key detect mode. Type Q to exit.
Char: <^[> Decimal:  27  Hex: 0x1B  Octal: \033
Char: < [> Decimal:  91  Hex: 0x5B  Octal: \133
Char: < 2> Decimal:  50  Hex: 0x32  Octal: \062
Char: < 8> Decimal:  56  Hex: 0x38  Octal: \070
Char: < ~> Decimal: 126  Hex: 0x7E  Octal: \176

Char: < Q> Decimal:  81  Hex: 0x51  Octal: \121
```

Figure 229. Display Keycodes Example

23.2.7 Display OCS II Telnet Server ′trace′ Facility

```
TS> trace -x
        -> PID: 6237  Start Time: Thu Jul 19 08:11:50 1996
        -> Current Trace Log Filename: L0421152625
        -> Last 2 trace files being kept.
        -> Maximum byte size of Trace Files is: 1046576
  Lvl  Description                        Lvl  Description
  ---  --------------------------------   ---  --------------------------
  100  TCP/IP Control Trace Level         101  Dump ALL TCP/IP Packets
  102  Dump PS TCP/IP Packets             103  Dump Telnet TCP/IP Packets
  104  Dump FTP TCP/IP Packets            105  Dump IP Route TCP/IP Packets
  106  Dump RSH TCP/IP Packets            107  Dump Internal IFBUF Packets
  108  Dump Short Format IFBUF Packets    199  Full TCP/IP Trace Level
  200  DLC Control Trace Level            201  DLC Read/Writes Trace Level
  202  Dump DLC Packets (Long format)     203  Dump DLC Packets (Short format)
  299  Full DLC Trace Level               300  SUP Malloc Trace Level
```

Figure 230. Display ′trace′ Example

23.2.8 Display OCS II Telnet Server Services

```
TS> service
5250 is default, available services are:
    3278      - TELNET Server/IBM 3278 display station emulation.
    5250      - TELNET Server/IBM 5270 display station emulation.
    tn3270    - TN3270 client support.
    tn3270E   - TN3270E client support.
    tn5250    - TN5250 client support.
    local     - local TELNET redirector support.
```

Figure 231. Display Services Example

23.2.9 Display OCS II Telnet Server Terminal Types

```
TS> showterm
Terminals known to TELNET Server:
ANSI                FALCO-925-32        IBM-3477-FC-E       TEKTRONIX-4115
APOLLO              FALCO-925-32W       IBM-3477-FC-P       TEKTRONIX-4115N
ATT-4418            FALCO-925W          IBM-3477-FG         TEKTRONIX-4205
ATT-4425            HP-2640             IBM-3477-FG-E       TEKTRONIX-4224
ATXENIX             HP-2640C            IBM-3477-FG-P       TEKTRONIX-4316
DATA-GENERAL-210    HP-9000             IBM-5251-11         TEKTRONIX-4316X
DATA-GENERAL-410    HP-9000C            IBM-5251-11-E       TEKTRONIX-4324
DATA-GENERAL-410A   IBM-3151            IBM-5251-11-P       TELEVIDEO-925
DEC-VT100           IBM-3164            IBM-5292-1          VISENTECH-VS230-8
DEC-VT100-32        IBM-3179-2          IBM-5292-1-E        WYSE-50
DEC-VT100-C         IBM-3179-2-E        IBM-5555-B01        WYSE-60
DEC-VT100C          IBM-3179-2-P        IBM-5555-B01-E      XTERM
DEC-VT100G          IBM-3196-A1         IBM-5555-C01        allhosts
DEC-VT102           IBM-3196-A1-E       IBM-5555-C01-E      dgbasic
DEC-VT220           IBM-3196-A1-P       LINK-125            tekbasic
DEC-VT220-8         IBM-3278-2          LSI-ADM-3A          vt100
DEC-VT300           IBM-3278-3          LSI-ADM-3A+         vt220
DEC-VT300-8         IBM-3278-4          LSI-ADM-5           vt220basic
DEC-VT320           IBM-3278-5          MACPLUS             vtbasic
DEC-VT320-132       IBM-3279-2          PCENH
DEC-VT320-8         IBM-3287-1          PCXTC
DEC-VT52            IBM-3477-FC         SUN
```

Figure 232. Display Terminal Types Example

23.2.10 Display OCS II Telnet Server Status

```
TS> status
TELNET Server finds everything ok
```

Figure 233. Status Display Example

23.2.11 Display Terminal Characteristics

```
TS> termcap vt100
vt100:
  tc=DEC-VT100:
d1|DEC-VT100|VT100|vt100|DEC vt100 capabilities:
  ku=\E[A¢Up Arrow¢:
  kd=\E[B¢Down Arrow¢:
  kr=\E[C¢Right Arrow¢:
  kl=\E[D¢Left Arrow¢:
  k0=\EOP¢PF1¢:
  k1=\EOQ¢PF2¢:
  k2=\EOR¢PF3¢:
  k3=\EOS¢PF4¢:
  ta=^I¢Tab¢:
  cs=|:
  f0=\EOP¢PF1¢|\E1¢Esc 1¢:
  f1=\EOQ¢PF2¢|\E2¢Esc 2¢:
  f2=\EOR¢PF3¢|\E3¢Esc 3¢:
  f3=\EOS¢PF4¢|\E4¢Esc 4¢:
  tc=vtbasic:
d0|vtbasic|DEC vt basic capabilities:
  cr=^M¢Return¢:
  nl=^J:
  cl=\E[;H\E[2J:
  cm=\E[%i%d;%dH:
  ce=\E[K:
  cd=\E[J:
  se=\E[m:
  hl=\E[1m:
  so=\E[7m:
  uc=\E[4m:
  bk=\E[5m:
```

Figure 234. Display Terminal Characteristics Example

23.2.12 OCS II Telnet Server Help Information

After having started the TELNET session, help information is available by pressing the appropriate keys as defined in the 'octermcap' file. For example, a PS/2 client, using termtype vt220 and a 3278 emulation, gets help information by pressing **Shift+PF5**; the result will be the screen shown in Figure 236 on page 303.

```
                    >>> TELNET Server 3278 Help Screen <<<
          3278 Key  Key Cap                         3278 Key  Key Cap
             ENTER  Return               DOUBLE LEFT ARROW  ^c
            RETURN  ^J                     RIGHT ARROW  Right Arrow
             RESET  ^a              DOUBLE RIGHT ARROW  ^b
       HELP TOGGLE  Help                            UP ARROW  Up Arrow
        TERM RESET  Esc tty                       DOWN ARROW  Down Arrow
      EXIT SESSION  Ctrl-3 xx                         INSERT  Ins Here
                    Esc xx                            DELETE  Remove
EXIT TO TS MANAGER  Ctrl-3 xe                     FIELD MARK  ^k
                    Esc xe                        L25 TOGGLE  ^y
             CLEAR  Ctrl-3 z                         SYS REQ  Select
           REPAINT  Ctrl-3 r                            ATTN  Find
         ERASE EOF  Prev Scr                         CUR SEL  Esc c Cr
       ERASE INPUT  Next Scr                             DUP  ^d
         BACKSPACE  Rubout                               PA1  Ctrl-3 a1 Cr
               TAB  Tab                                  PA2  Ctrl-3 a2 Cr
          BACK TAB  ^e                               PA3  Ctrl-3 a3 Cr
              HOME  ^                                PF1  PF1
        LEFT ARROW  Left Arrow                           PF2  PF2
Server: OCS II RISC/6000, V3.7.1        Socket#: 5    PU: IPFP2209  LU: IPFT2S90
Terminal: vt220
Attributes: Blink Underline Reverse Highlight        Colors: Not Supported
Press Space Bar for more help .....[screen 1 of 2[.....
```

Figure 235. Telnet Server Sample

```
TS> help
OpenConnect (R) TELNET Server --- Help

To start a TELNET Server 3278 session:
   TS> !term_type,3278,auto_detect,LU,PU_name

To start a TELNET Server 5250 session:
   TS> !term_type,5250,auto_detect

Where: term_type    - Terminal entry name defined in the octermcap file.
       3278 or 5250 - Emulation (service) type.
       auto_detect  - Auto-detect Carriage Return flag (default=Y).
       LU           - LU name or number, 3278 only (default=first available).
       PU_name      - PU name, 3278 only (default=first available).

To start a 3278 session using a Wyse 50 terminal, with auto-detect CR,
through LU 4 of a PU named PU2 enter:  !wyse50,3278,,4,PU2

To start a 5250 session using a vt220 terminal, without auto-detect CR,
enter:  !vt220,5250,N

No other services are currently supported by this version of the TELNET
Server.  Contact your system administrator for more detailed documentation.
```

Figure 236. Telnet Server Help Sample

Chapter 24. OC/SAM Programming

The OC/SAM software packet allows a VSE system to communicate with a local area network using an OpenConnect Server. The program on the VSE system communicates with a partner program on the LAN, using either the User Datagram Protocol (UDP) or Transmission Control Program (TCP). Program development using UDP can be either single or multi-user. Program development using TCP can be either single or, if using VTAM LU0, multi-user.

The Socket Access Method provides the entry point into the network, the so-called socket, which is used by the application program to communicate with the partner program. The network complexity is handled by the OC/SAM software product. The only programming required in the VSE program is for the protocol established by the partner program.

The interface between the mainframe application and the OpenConnect Server is the APPL definition (VTAM), the LU definition (VTAM), or channel address (GTO). All transactions flow through this single path. There can be any number of partners on the LAN interfacing to the VSE application through this single path. This eliminates the tying up of more than one resource (LU) on the OpenConnect Server to run a multi-user configuration.

Before writing your own socket programs you should study the sample programs provided in TCPOCS.SAMSMP.

24.1 TCP/IP Application Programming Tutorial

24.1.1 Network Programming Overview

Each process that uses TCP/IP communication services at a given node is assigned a unique **port** number. This process is called a **TCP/IP server** and can, from a conceptual point of view, be ′called′ by any TCP/IP client using that port number. The server uses a port number that all potential clients know. A TCP/IP client process typically uses one port to establish a communication with the server and another one to request services and transfer data.

The TCP/IP communication software from OCS supports the sockets network programming API. Sockets provide a network programming API that is very similar to do input/output to local devices. The sockets API is currently by far the most commonly used API for network programming in the TCP/IP environment.

24.1.2 Socket Programming

The socket access method supplied by OCS supports two different types of sockets:

1. Datagram sockets: These are sockets that can be used to communicate using the User Datagram Protocol (UDP). Datagram sockets support an unreliable, datagram form of data transfer in which individual user datagrams can be sent from one socket to another.

2. Stream Sockets: These are sockets that can be used to communicate using the Transmission Control Protocol (TCP). Stream sockets support a connection-oriented form of data transfer where streams of data can be sent from one socket to another over a TCP connection.

24.1.3 Socket Addresses

To establish a TCP/IP session, the sockets communication API functions need address information about the communication partners. A socket address is comprised of:

1. An Internet address: a four-byte integer value containing the internet address, in binary format, of the host in which the process is running. A client process must know the internet address of the host on which the server is running.
2. A Port: a two-byte integer value identifying the port number assigned to the process. A client must know the port number that the server is using. A client typically asks the communication software to supply it.

24.1.4 Application Protocols

24.1.4.1 Connectionless Application Protocol

A client/server application that implements a connectionless application protocol uses datagram sockets to communicate using UDP. Figure 237 on page 307 shows the sequence in which the API functions are typically issued in an application that uses a connectionless protocol.

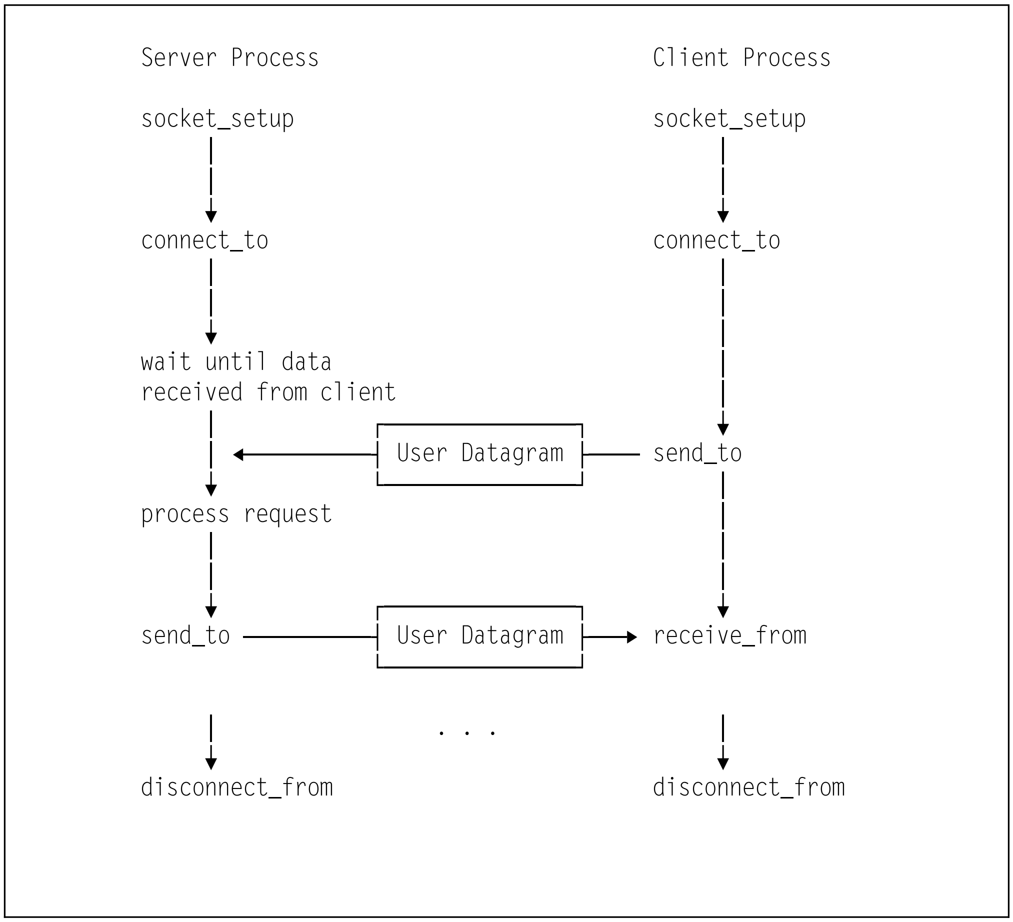


Figure 237. Typical Sockets API Functions for UDP

Steps performed by a connectionless server:

1. The server issues several SAM functions to provide protocol information that will eventually be required to support communication between the client and the server.
2. The server issues a ′connect_to′ function, to provide his own port number. The internet address of the own host is returned by this function.
3. The server then issues a ′receive_from′ function that allows the server to accept incoming data from a client. The ′receive_from′ function causes the server to wait until data arrives from any client that wishes to use the services of the server process.
4. When an incoming user datagram arrives from a client, the server performs any required processing. The protocol control information included in the incoming user datagram includes the client host′s internet address and the port number used by the client process on that host.
5. The server can then reply to the client by issuing one or more ′send_to′ functions. The ′send_to′ function references the client′s socket address that was received in the previous step.
6. To terminate the connection with the port number that was provided in the ′connect_to′ request, the ′disconnect_from′ function is used.

Steps performed by a connectionless client:

1. The client issues the same setup functions as the server.

2. The client issues the ′connect_to′ function providing his own port number.
3. The client issues a ′send_to′ function to send a user datagram to the server. The client must know the socket address of the server.
4. If the client expects to receive data from the server, he issues a ′receive_from′ function that causes the client to wait until data arrives from the server.
5. To end the connection a ′disconnect_from′ is issued.

The **socket setup** is comprised of several SAM functions issued in sequence to initialize the interface with the OpenConnect server:

1. Get work area: allocate storage areas that will be used by this session.
2. Set socket options: set up the options that will be used during program execution (for example, request tracing).
3. Set net by LU: identifies the LU that will be used to communicate with the OpenConnect server.
4. Set ACB by name: set the ACB name that will be used to identify your application to VTAM.
5. Set protocol by name: Set the type of protocol (UDP or TCP) that will be used to communicate with the partner on the network.
6. Session start: establish a connection from the ACB to the LU of the OpenConnect server.

   Note: This SAM function does not establish sessions to TCP partner programs, only to the OC server.

When the network resources are no longer required by the program, the program terminates the socket in an orderly fashion:

1. Session end: terminate the session with the LU on the OpenConnect server.
2. Close all: this function closes any internal files and file buffers.
3. Put work area: all storage areas allocated by the ′Get work area′ function are released.

24.1.4.2 Connection-oriented Application Protocols

A client/server application that implements a connection-oriented application protocol uses stream sockets to communicate using TCP. Figure 238 on page 309 shows the sequence in which the API functions are typically issued in an application that uses a connection-oriented protocol.

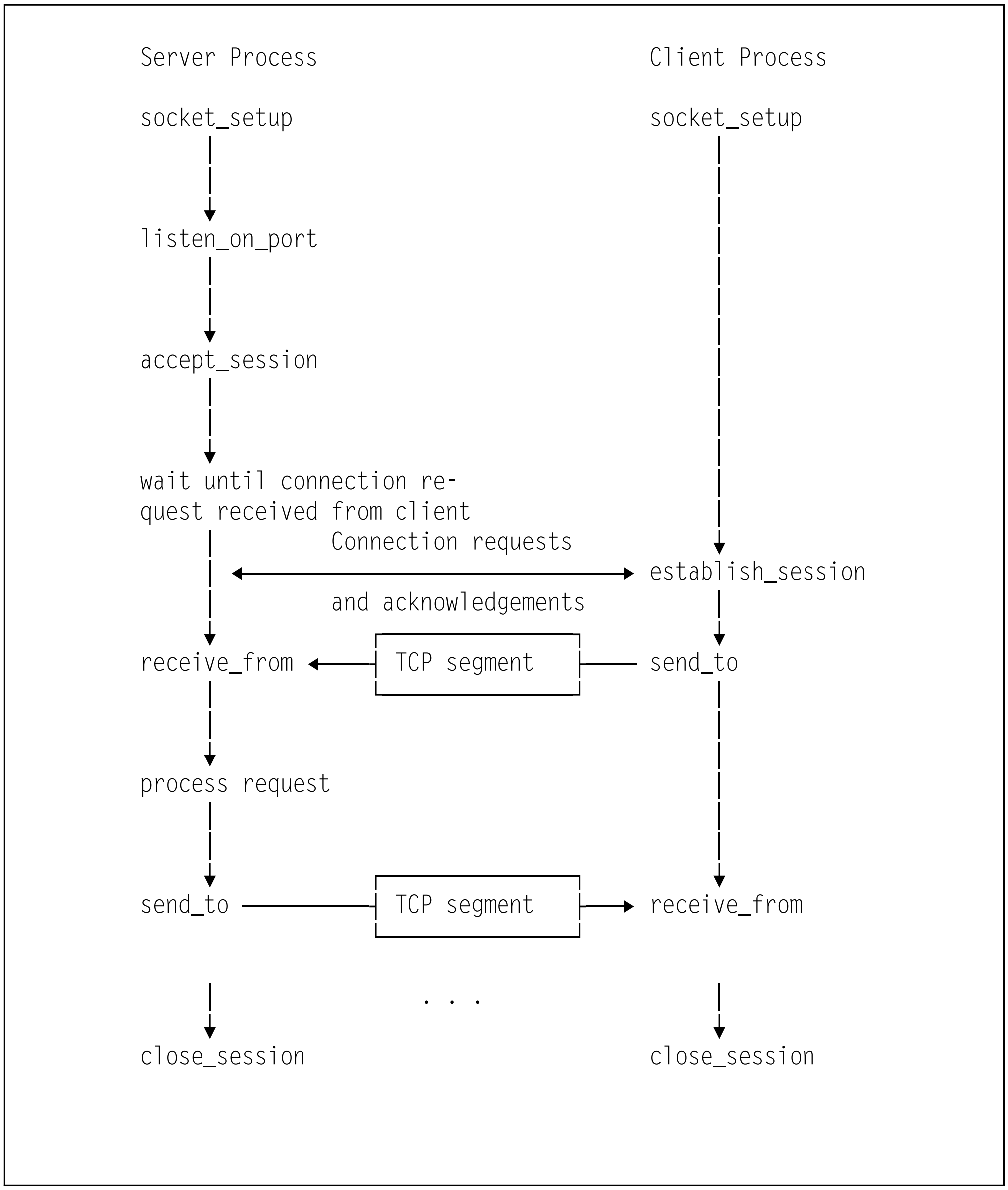


Figure 238. Typical Sockets API Functions for TCP

Steps performed by a connection-oriented server:

1. Server issues functions required for setting up the interface.
2. The server issues a listen function that links the server application to a specific port number.
3. The server issues an ′accept_session′ to indicate that the server is willing to accept a connection request from a client.
4. The ′wait until connection′ request can be done either on the ′accept_session′ or the ′receive_from′ function, depending on the options specified.
5. The server performs the processing required by the application and issues a ′send_to′ function that allows the server to send data to the client.
6. Before the server terminates, he issues the ′close_session′ request.

Steps performed by a connection-oriented client:

1. Set up sockets to initialize the communication interface.
2. The client issues an ′establish_session′ function to set up a session to the server. The client has to provide the socket address information of the server in this function.
3. The client uses a ′sent_to′ function to send data to the server over the established TCP connection.
4. To end the session created by ′establish_session′ the client issues the ′close_session′ request.

24.2 Running the Sample Programs

To run the sample programs provided by OC/SAM some preparatory work is required:

- Customize VTAM
- Adapt the source programs
- Create a job to execute the programs. All sample programs are assembled with the VSE/High Level Assembler (HLASM).

24.2.1 VTAM Customization

24.2.1.1 VTAM APPL Definitions

Before the sample programs can be run, the application names have to be defined in the APPL major node.

Figure 239 is an extract from the complete VTAM APPL provided in the sample library member ′$VTAMGEN′.

```
UDPTEST1 APPL  AUTH=(ACQ),EAS=2,ACBNAME=UDPTEST1
UDPTEST2 APPL  AUTH=(ACQ),EAS=2,ACBNAME=UDPTEST2
TCPTEST1 APPL  AUTH=(ACQ),EAS=2,ACBNAME=TCPTEST1
TCPTEST2 APPL  AUTH=(ACQ),EAS=2,ACBNAME=TCPTEST2
```

Figure 239. APPL Definitions for OC/SAM Samples

The key definitions and parameters are:

- **UDPTESTn / TCPTESTn APPL entries.** APPL entries for the UDP/TCP test programs. The key parameters are:
  - **ACBNAME.** Should be the same as the APPL label. The names must match the values specified in the sample programs.
  - **EAS=2.** Allows two concurrent sessions.

24.2.1.2 VTAM LU Definition

Figure 24 on page 47 lists all LUs we defined in our setup.

The two LU definitions we used for the OC/SAM sample programs are shown in the following figure.

```
IPF2S9I  LU    LOCADDR=9,DLOGMOD=OCSLU0,                          C
               ISTATUS=ACTIVE,MODETAB=OCSBIND,                    C
               MDLTAB=VTMMDL,MDLENT=VSELU2A,                      C
               SSCPFM=USSSCS,USSTAB=VTMUSSTR
IPF2S9P  LU    LOCADDR=16,DLOGMOD=OCSLU0,                         C
               ISTATUS=ACTIVE,MODETAB=OCSBIND,                    C
               MDLTAB=VTMMDL,MDLENT=VSELU2A,                      C
               SSCPFM=USSSCS,USSTAB=VTMUSSTR
```

Figure 240. LU Definitions for the OC/SAM Samples

The key parameters are:

- **The LU name.** Must match the:
  - LU name in the OCS II Gateway customization (see 5.1.2.4, "VTAM Local SNA Major Node" on page 46)
  - LU name in the OC/SAM sample programs
- **LOCADDR.** Must match the number used in the OCS II Gateway customization (see 5.1.2.4, "VTAM Local SNA Major Node" on page 46)
- **MODETAB and DLOGMOD.** Identify the appropriate logmode table entry. We used the same table as the OC/FTP Server (see 6.2.2, "VTAM Customization" on page 66).

24.2.2 OCS Sample Programs

Sublibrary TCPOCS.SAMSMP contains program samples for socket programming. In order to adapt the programs to your network environment, punch them into the ICCF library. To assemble and run the programs, use the sample job $VSEJCL.A provided in library TCPOCS.SAMSMP and modify it accordingly.

24.2.2.1 UDP Sample Programs

We ran sample programs UDPSBV0 and UDPCBV0 which use SNA LU0 to communicate with the OpenConnect Server.

UDPSBV0 is a TCP/IP server program using UDP to respond to the **UDPCBV0** client.

UDPCBV0 is a TCP/IP client program using UDP to call the **UDPSBV0** server.

Both programs are running on VSE/ESA and communicate with each other via the OCS II Gateway.

Some of the SAM function calls in the source code have to provide environment specific parameters. We modified that code as follows:

1. For the server UDPSBV0:

- GET_WORK_AREA function must use parameter value ″DOS″ instead of ″MVS″.
- SET_NET_BY_LU function must specify LU name ″IPFT2S9I″.
- SET_ACB_BY_NAME function must specify VTAM application program name ″UDPTEST1″.

2. For the client UDPCBV0:
   - GET_WORK_AREA function must use parameter value ″DOS″ instead of ″MVS″.
   - SET_NET_BY_LU function must specify LU name ″IPFT2S9P″.
   - SET_ACB_BY_NAME function must specify VTAM application program name ″UDPTEST2″.

The modified source code is then cataloged back into library TCPOCS.SAMSMP. Then we assembled and ran the UDP server program using the following job:

```
* $$ JOB JNM=UDPSBV0,CLASS=0,DISP=D
* $$ LST DISP=D,CLASS=A,DEST=(,WSTIEBER)
// JOB UDPSBV0
// LIBDEF SOURCE,SEARCH=TCPOCS.SAMMAC
// LIBDEF OBJ,SEARCH=TCPOCS.SAMOBJ
// LOG
// OPTION LINK
// EXEC PGM=ASMA90,SIZE=(ASMA90,50K),                                  C
               PARM=SYSPARM(DOS)¢
* $$ SLI MEM=UDPSBV0.A
/*
// EXEC PGM=LNKEDT
/*
// EXEC  ,SIZE=AUTO
/*
/&
* $$ EOJ
```

Figure 241. Assemble and Run the UDP Server Program

When the server program UDPSBV0 is ready to accept client requests, the following message appears on the VSE system console:
′SOCK$AM/UDPS READY FOR CLIENT′
Now the client program UDPCBV0 can be assembled and started.

```
* $$ JOB JNM=UDPCBV0,CLASS=5,DISP=D
* $$ LST DISP=D,CLASS=A,DEST=(,WSTIEBER)
// JOB UDPCBV0
// LIBDEF SOURCE,SEARCH=TCPOCS.SAMMAC
// LIBDEF OBJ,SEARCH=TCPOCS.SAMOBJ
// LOG
// OPTION LINK
// EXEC PGM=ASMA90,SIZE=(ASMA90,50K),                                  C
               PARM=SYSPARM(DOS)¢
* $$ SLI MEM=UDPCBV0.A
/*
// EXEC PGM=LNKEDT
/*
// EXEC  ,SIZE=AUTO
/*
/&
* $$ EOJ
```

Figure 242. Assemble and Run the UDP Client Program

The programs exchange short notices to test the UDP link and complete without further messages. Check the trace entries in the job output to verify that the programs communicated successfully.

24.2.2.2 TCP Sample Programs

We ran sample programs TCPSBV0, TCPCBV0 and TCPSMTP which use SNA LU0 to communicate with the OpenConnect Server.

TCPSBV0 is a TCP/IP server program using TCP to respond to the **TCPCBV0** client.

TCPCBV0 is a TCP/IP client program using TCP to call the **TCPSBV0** server.

TCPSMTP is a TCP/IP client program using TCP to send mail to a standard TCP/IP SMTP (Simple Mail Transfer Protocol) server.

TCPSBV0 and TCPCBV0 are running on VSE/ESA and communicate with each other via the OCS II Gateway.

Some of the SAM function calls in the source code have to provide environment specific parameters. We modified that code as follows:

1. For the server TCPSBV0:
   - GET_WORK_AREA function must use parameter value ″DOS″ instead of ″MVS″.
   - SET_NET_BY_LU function must specify LU name ″IPFT2S9I″.
   - SET_ACB_BY_NAME function must specify VTAM application program name ″TCPTEST1″.
2. For the client TCPCBV0:
   - ESTABLISH_SESSION function must specify the server IP address, in our case X′C03D6451′ (192.61.100.81).
   - GET_WORK_AREA function must use parameter value ″DOS″ instead of ″MVS″.

- SET_NET_BY_LU function must specify LU name ″IPFT2S9P″.
- SET_ACB_BY_NAME function must specify VTAM application program name ″TCPTEST2″.

The modified source code is then cataloged back into library TCPOCS.SAMSMP. Then we assembled and ran the TCP server program using the following job:

```
* $$ JOB JNM=TCPSBV0,CLASS=0,DISP=D
* $$ LST DISP=D,CLASS=A,DEST=(,WSTIEBER)
// JOB TCPSBV0
// LIBDEF SOURCE,SEARCH=TCPOCS.SAMMAC
// LIBDEF OBJ,SEARCH=TCPOCS.SAMOBJ
// LOG
// OPTION LINK
// EXEC PGM=ASMA90,SIZE=(ASMA90,50K),                                  C
               PARM=SYSPARM(DOS)¢
* $$ SLI MEM=TCPSBV0.A
/*
// EXEC PGM=LNKEDT
/*
// EXEC  ,SIZE=AUTO
/*
/&
* $$ EOJ
```

Figure 243. Assemble and Run the TCP Server

When the server program TCPSBV0 is ready to accept client requests, the following message appears on the VSE system console:
′SOCK$AM/TCPS READY FOR CLIENT′
Now the client program can be assembled and started.

```
* $$ JOB JNM=TCPCBV0,CLASS=5,DISP=D
* $$ LST DISP=D,CLASS=A,DEST=(,WSTIEBER)
// JOB TCPCBV0
// LIBDEF SOURCE,SEARCH=TCPOCS.SAMMAC
// LIBDEF OBJ,SEARCH=TCPOCS.SAMOBJ
// LOG
// OPTION LINK
// EXEC PGM=ASMA90,SIZE=(ASMA90,50K),                                  C
               PARM=SYSPARM(DOS)¢
* $$ SLI MEM=TCPCBV0.A
/*
// EXEC PGM=LNKEDT
/*
// EXEC  ,SIZE=AUTO
/*
/&
* $$ EOJ
```

Figure 244. Assemble and Run the TCP Client

The programs exchange short notices to test the TCP link and complete without further messages. Check the trace entries in the job output to verify that the programs communicated successfully.

TCPSMTP is a batch program which communicates with an SMTP mail server program.

Some of the SAM function calls in the source code have to provide environment specific parameters. We modified that code as follows:

- ESTABLISH_SESSION function must specify the server IP address, in our case X′C03D6451′ (192.61.100.81).
- GET_WORK_AREA function must use parameter value ″DOS″ instead of ″MVS″.
- SET_NET_BY_LU function must specify LU name ″IPFT2S9P″.
- SET_ACB_BY_NAME function must specify VTAM application program name ″TCPTEST1″.

The modified source code is then cataloged back into library TCPOCS.SAMSMP.

The TCPSMTP program reads mail to be sent from SYSIPT, the input must also include the standard SMTP control information. A more advanced mail client function for VSE is described in the next section.

To run the TCPSMTP program we submitted the following job:

```
* $$ JOB JNM=TCPSMTP,CLASS=0,DISP=D
* $$ LST DISP=D,CLASS=A,DEST=(,WSTIEBER)
// JOB TCPSMTP
// LIBDEF SOURCE,SEARCH=TCPOCS.SAMMAC
// LIBDEF OBJ,SEARCH=TCPOCS.SAMOBJ
// LOG
// OPTION LINK
// EXEC PGM=ASMA90,SIZE=(ASMA90,50K),                                  C
               PARM=SYSPARM(DOS)¢
* $$ SLI MEM=TCPSMTP.A
/*
// EXEC PGM=LNKEDT
/*
// EXEC ,SIZE=AUTO
 MAIL FROM:<wacker@aix320>
 RCPT TO:<pram@aix320>
 DATA
 SUBJECT: VSE/aix
 This is a line of data from an SMTP client on the VSE system.
/*
/&
 * $$ EOJ
```

Figure 245. Assemble and Run the TCPSMTP Mail Client

24.3 SMTP Mail Example

Based on the TCPSMTP sample program discussed in the previous section, we developed an extension to implement a simple but efficient mail facility for VSE/ESA based on TCP/IP SMTP.

The example described in this chapter works as a standard SMTP mail client which delivers mail to standard Internet SMTP mail servers, in order to send mail to other Internet users.

In our test environment we sent mail from the VSE system to users on AIX, OS/2 and Windows and where the mail can be retrieved using the standard TCP/IP-based mail programs on these platforms.

> **Note**
>
> The programs referred to in this chapter are available on the IBMVSE tools disk (managed by VMTOOLS at BOEVM3).

24.3.1 Mail Example Overview

A mail system is typically divided in two parts, the ″Message Transfer Agent″ and the ″Message Entry Function″. The basic structure of our implementation is illustrated in the following figure.

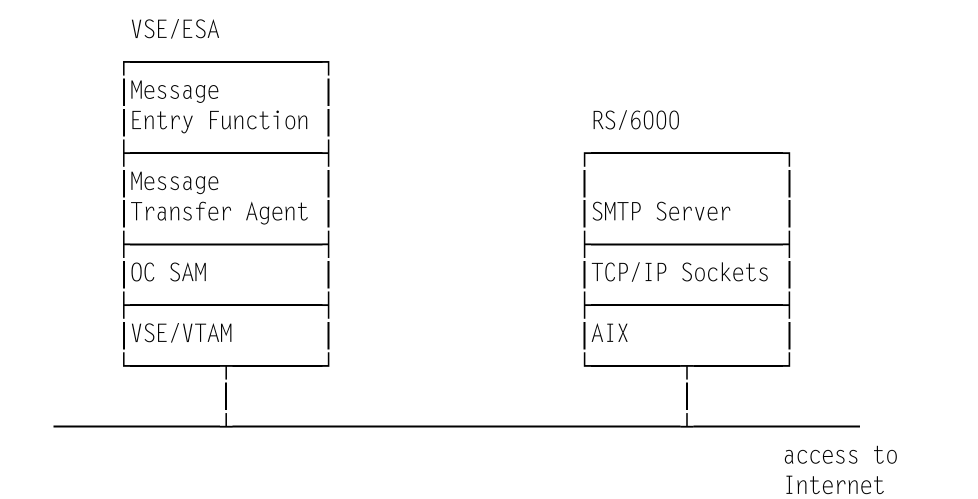


Figure 246. VSE/ESA SMTP Mail Client Structure

The SMTP mail function works as follows:

1. The Message Transfer Agent (MTA) receives the mail text and sends it to the SMTP server after having added the necessary SMTP protocol information. The MTA is implemented on VSE/ESA using the OC/SAM programming API.

2. The Message Entry Function (MEF) allows users to enter a mail text and send it to the desired recipients by using a PF-key.

   The MEF was implemented as a simple dialog function using the VSE/ESA II (Interactive Interface). Figure 247 on page 317 shows an example of how to enter mail using the MEF.

```
 IESADMMAIL                  Send a Note
 To: wacker@aix320 pramot@tcpcl3
 Cc: wstieber@aix320
 Subject: VSE/mail
                                                                 <
Hello,                                                           =
this is a mail entered via the VSE message entry function.       =
The VSE message transfer agent generates the required SMTP       =
control cards and sends them to the SMTP server on AIX.          =
  From there on the SMTP mail servers handle the delivery of the =
mail in the TCP/IP network.                                      =
                                                                 =
See you later on OS/2.                                           =
                                                                 =
                                                                 =
                                                                 =
 Enter receiver address and your mail.
 PF1=HELP   2=DEL LINE 3=END      4=RETURN   5=SEND     6=ADD LINE
```

Figure 247. Entering SMTP Mail on VSE/ESA

Note

The Message Entry Function (MEF) is not described further in this book; a detailed description is available on the IBMVSE tools disk (managed by VMTOOLS at BOEVM3).

24.3.2 Message Transfer Agent

The Message Transfer Agent (MTA) receives messages via the MEF and sends them to an SMTP server program.
A message consists of:

1. A message header, containing address information of the sender and the recipients and other mail information.
2. A message body, containing the mail text.

Our implementation of the MTA supports the following Message Header (MH) components (note that all MH component identifiers have to start in column 1):

MH-Component Comment

From: Specifies the sender address of the message. Has to be the first MH component. In our implementation this is inserted by the MEF and corresponds to the VSE/ESA user ID.

To: Specifies the address(es) of the primary recipient(s). At least one user has to be specified. Multiple users can be specified separated by blanks.

Cc: Specifies the address(es) of secondary recipient(s), that is the ′carbon copy′ list.

Subject: Provides for a comment line which will be displayed by the receiver′s mail system.

The message header is ended by a line of dashes (″-------″).
The message body ends with a ″.″ (dot) on the first character position.

Taking our example from Figure 247, the following lines are passed to the MTA:

```
To: wacker@aix320 pramot@tcpcl3
Cc: wstieber@aix320
Subject: VSE/mail
-------
Hello,
this is a mail entered via the VSE message entry function.
The VSE message transfer agent generates the required SMTP
control cards and sends them to the SMTP server on AIX.
  From there on the SMTP mail servers handle the delivery of the
mail in the TCP/IP network.

See you later on OS/2.
.
```

Figure 248. Message Input to the VSE/ESA Message Transfer Agent

All necessary SMTP control cards are generated by the Message Transfer Agent.

To handle the communication with the calling program the message is prefixed by an 80-byte Control Buffer, mainly to signal error conditions to the caller.

The contents of this Mail Control Buffer are listed in the following figure.

```
MAILDS   DSECT                  MAIL CB PASSED BY CALLER
MAILHD   DS    0CL80            VSE/ESA MAIL HEADER
MLTAG    DS    CL6              CONSTANT ¢MAILCB¢
MLRC     DS    H¢0¢             RETURN CODE TO CALLER
MLRCOK   EQU   X¢00¢            OK
MLRCSMTP EQU   X¢0A¢            SMTP ERROR
MLRCSOCK EQU   X¢0C¢            SOCK$AM ERROR
MLRSC    DS    H¢0¢             REASON CODE
MLSRC    DS    CL1              MAIL SOURCE (=M MEANS COMMAREA)
MLTROPT  DS    CL1              TRACE OPT (=T MEANS TRACE)
MLOPTFIL DS    CL6              EMPTY SLOTS
MLSRCQN  DS    CL8              NAME OF TS QUEUE OF SOURCE
MLTRQN   DS    CL8              NAME OF TS QUEUE FOR TRACE
MLDATA   DS    0CL13            DIAGNOSTIC: STRING DATA
MLSMMSG  DS    CL3              SMTP: MSG NUMBER/ SAM: REQ.
MLSMUSR  DS    CL10             SMTP: UNKNOWN USER
*
```

Figure 249. Layout of VSE MTA Control Buffer

24.3.3 SMTP Server

The VSE/ESA MTA transmits the mail to the AIX SMTP server. The message header is scanned and control records are generated which are sent to the SMTP server prior to the actual mail data.

These are the control records sent to the SMTP server:

Control Card Comments

MAIL FROM: The MAIL FROM card is generated out of the ″From:″ MH-component. The ″From:″ MH-component has to be the first entry in the message header.

RCPT TO: For each user specified in the ″To:″ and ″Cc:″ MH-components, a ′RCPT TO:′ record is generated.

DATA When the end-of-message indication is detected, the MTA sends a DATA record, indicating to the SMTP server that all following lines, up to the ″.″ (dot) line, are part of the data to be sent. The MTA sends all message header lines after the DATA record.

QUIT After all message lines are sent the MTA disconnects from the SMTP server using the SMTP ′QUIT′ command.

The VSE/ESA MTA supports a TRACE function, which writes all message lines and SMTP control records sent as well as the messages returned by the SMTP server to CICS Temporary Storage (using a default TS Queue name of ″IESXMAIL″).

The following TRACE lines were written to subject TS Queue when we sent the sample mail message shown in Figure 247 on page 317:

```
IESXMAILDATE=00094340TIME=00164628
MAILCB....MT              IESXMAIL
220 aix320.itsc.ibm.com Sendmail AIX 3.2/ UCB 5.64/4.03 ready at Tu
From:wacker
MAIL FROM:wacker
250 wacker... Sender is valid
To:wacker@aix320 pramot@tcpcl1
RCPT TO:wacker@aix320
250 wacker@aix320... Recipient is valid
RCPT TO:pramot@tcpcl1
250 pramot@tcpcl1... Recipient is valid
Subject: VSE/mail
Cc:wstieber@aix320
RCPT TO:wstieber@aix320
250 wstieber@aix320... Recipient is valid
-------
354 Enter mail. End with the . character on a line by itself
From:wacker
To:wacker@aix320 pramot@tcpcl1
Subject: VSE/mail
Cc:wstieber@aix320
-------
Hello,
this is a mail entered via the VSE message entry function.
The VSE message transfer agent generates the required SMTP
control cards and sends them to the SMTP server on AIX.
  From there on the SMTP servers handle the delivery of the
mail in the TCP/IP network.

See you later on OS/2.
.
Ok
221 aix320.itsc.ibm.com: closing the connection
MAILCB....MT              IESXMAIL
```

Figure 250. Communication between the VSE/ESA MTA Client and the SMTP Server

24.3.4 Receiving the Mail

To test our OC/SAM mail client we sent mail to AIX, OS/2 and Windows workstations. These platforms use different facilities to receive incoming mail from other TCP/IP nodes:

- AIX

  The ′mail′ command is used to display the contents of the mailbox. From the mailbox prompt mailbox commands can be entered to manage the contents of the mailbox.

- OS/2

  Uses the ′LaMail′ Presentation Manager application.

- Windows

  Uses the ′Mail′ Windows application.

Using these facilities the mail systems allow you to read the incoming mail, print it and store it into personal mail folders. As an example Figure 251 on page 321 and Figure 252 on page 321 show how our VSE/ESA mail was received and viewed using OS/2′s LaMail facility.

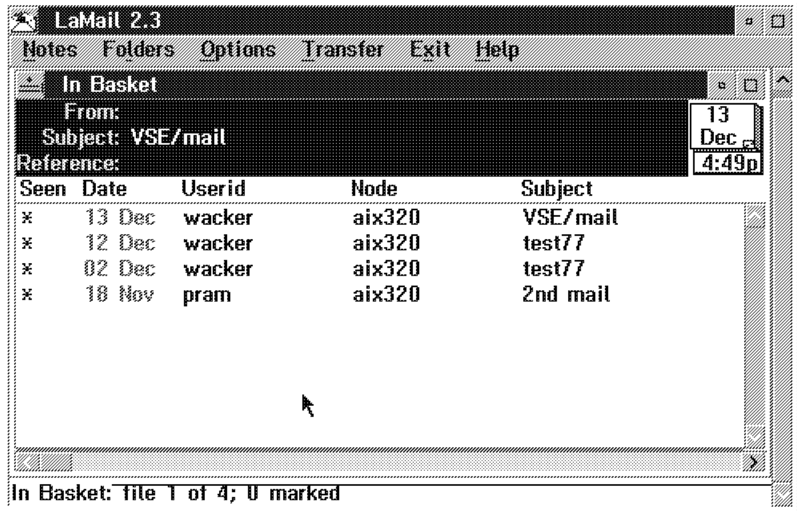


Figure 251. Receiving VSE/ESA SMTP Mail Using OS/2 LaMail

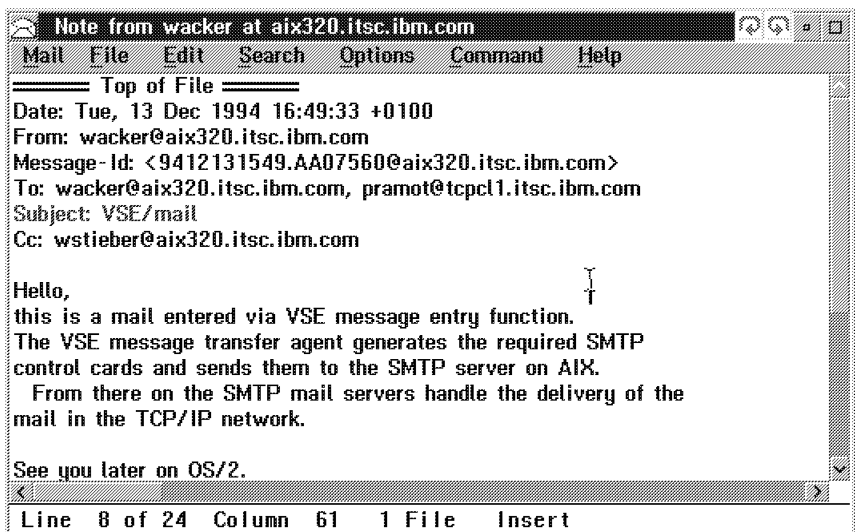


Figure 252. Reading VSE/ESA SMTP Mail Using OS/2 LaMail

24.3.5 Message Transfer Agent Program Implementation and Setup

The MTA implementation of the MTA is based on the sample program IESMTA which is available on the IBMVSE tools disk (managed by VMTOOLS at BOEVM3).

The following steps are required to set up IESMTA:

- Customize Assembler program IESMTA
- Customize CICS

- Customize OC/SAM

No further customization is required for VTAM, since we already provided the necessary APPL statements and the VTAM LU definitions (see 24.2, "Running the Sample Programs" on page 310).

24.3.5.1 Customize IESMTA

The following adaptations have to be made:

1. Define the IP address of the TCP/IP host running the SMTP server.
2. Define the communication type to be used for I/O to the OpenConnect server. The sample program uses SNA LU0 communication by default.
3. Define the LU name to be used for OCS II Gateway communication.

We used the job listed below to assemble the IESMTA program.

```
* $$ JOB JNM=COMMTA,DISP=D,CLASS=A,NTFY=YES
* $$ LST DISP=D,CLASS=Q,PRI=3
* $$ PUN DISP=I,PRI=9,CLASS=A
// JOB COMMTA TRANSLATE PROGRAM IESMTA
* VSE/MAIL INSTALL STEP: TRANSLATE IESMTA PROGRAM
// LIBDEF SOURCE,SEARCH=TCPOCS.SAMMAC
// LIBDEF OBJ,SEARCH=TCPOCS.SAMOBJ
// ASSGN SYSIPT,SYSRDR
// EXEC IESINSRT
$ $$ LST DISP=D,CLASS=Q,PRI=3
// JOB COMMTA COMPILE PROGRAM IESMTA
* VSE/MAIL INSTALL STEP: ASSEMBLE IESMTA PROGRAM
// LIBDEF SOURCE,SEARCH=TCPOCS.SAMMAC
// LIBDEF OBJ,SEARCH=TCPOCS.SAMOBJ
// SETPARM CATALOG=1
// IF CATALOG = 1 THEN
// GOTO CAT
// OPTION ERRS,SXREF,SYM,LIST,NODECK
// GOTO ENDCAT
/. CAT
// LIBDEF PHASE,CATALOG=PRD2.CONFIG
// OPTION ERRS,SXREF,SYM,CATAL,NODECK
   PHASE IESMTA,*
   INCLUDE DFHEAI
/. ENDCAT
// EXEC ASMA90,SIZE=(ASMA90,50K),PARM=¢EXIT(LIBEXIT(EDECKXIT))¢
* $$ END
// ON $CANCEL OR $ABEND GOTO ENDJ2
// OPTION NOLIST,NODUMP,DECK
// EXEC DFHEAP1$,SIZE=512K
*ASM XOPTS(CICS)
* $$ SLI ICCF=(IESMTA),LIB=(0029)
/*
/. ENDJ2
// EXEC IESINSRT
/*
// IF CATALOG NE 1 OR $MRC GT 4 THEN
// GOTO NOLNK
// EXEC LNKEDT,SIZE=256K
/. NOLNK
#&
$ $$ EOJ
* $$ END
/&
* $$ EOJ
```

Figure 253. Assemble Message Transfer Agent Program IESMTA

24.3.5.2 CICS Customization

This section covers the definitions required in CICS/VSE to enable VSE/ESA users to use the MTA. The CICS CEDA transaction was used to define the following resources to CICS/VSE:

- Program IESMTA

We filed this resource under group 'VSEGROUP'.

The screen image from our CEDA session is shown in the figure below.

```
CEDA  DEFINE
 PROGram        : IESMTA
 Group          : VSEGROUP
 Language       : Assembler         CObol | Assembler | C | Pli | Rpg
 RELoad         : No                No | Yes
 RESident       : No                No | Yes
 RSl            : 00                0-24 | Public
 Status         : Enabled           Enabled | Disabled
REMOTE ATTRIBUTES
 REMOTESystem   :
 REMOTEName     :
 Transid        :
 Executionset   : Fullapi           Fullapi | Dplsubset
```

The key parameters are:

- ***PROGRAM.*** The name of the Message Transfer Agent program.
- ***GROUP.*** The name of the group to which this definition belongs.
- ***LANGUAGE.*** The programming language of this program is Assembler.

24.3.5.3 OC/SAM Customization

IESMTA uses the ′GET SERVICE BY NAME′ function to obtain the port number and protocol type for communication with the SMTP server. You need to make sure that your ′GET SERVICE BY NAME′ table contains an entry for this port. The default table provided by OC/SAM contains this entry, as shown in the figure below.

```
         DC    CL20¢SMTP¢               Name of service
         DC    H¢25¢                    Port number service is using
         DC    CL4¢TCP¢                 Protocol service is using
         DC    CL20¢ ¢                  Alias name for service
```

Figure 254. SMTP Entry in Default ′GET SERVICE BY NAME′ Table

If the table does not contain an entry for SMTP, add an entry as described in 11.2.1, “GET SERVICE BY NAME Table Customization” on page 150.

Chapter 25. Implementation Summary

The successful implementation of OCS products with VSE/ESA depends heavily on your understanding of the:

- Definition requirements in VSE/ESA and the OCS products
- Relationships between the VSE/ESA and OCS products customization

The definitions required in VSE/ESA and the OCS II Gateway are covered in previous chapters. This chapter summarizes the **relationships** between the definitions in the different subsystems affected using a number of figures.

These figures are meant to help you to better understand the implementation of OCS products. Pointers in the figures illustrate the relationships of the definitions and where parameter values must match.

The following definition relationships are illustrated:

- VTAM and OCS II Gateway Definitions
- OC/FTP Server Definitions
- OC/FTP Client, OC/RSH Client, and OC/LPD Definitions
  - VTAM and CICS Definitions
  - VTAM, OC/FTP Client, OC/RSH Client and OCS II Gateway Definitions
- OC/TELNET FS Definitions

25.1 VTAM and OCS II Gateway Definitions

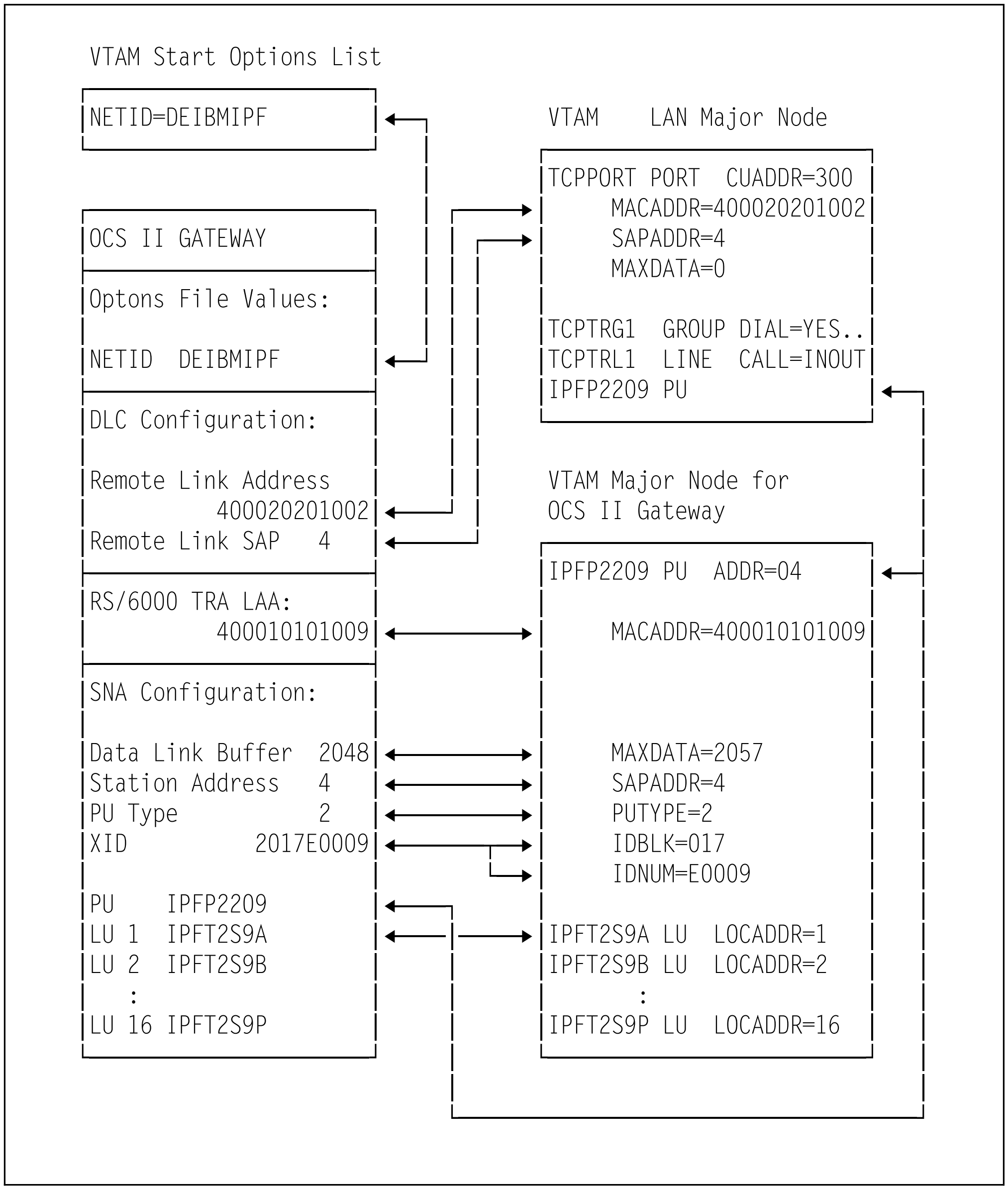


Figure 255. OCS II Gateway and VTAM Definitions

25.2 OC/FTP Server Definitions

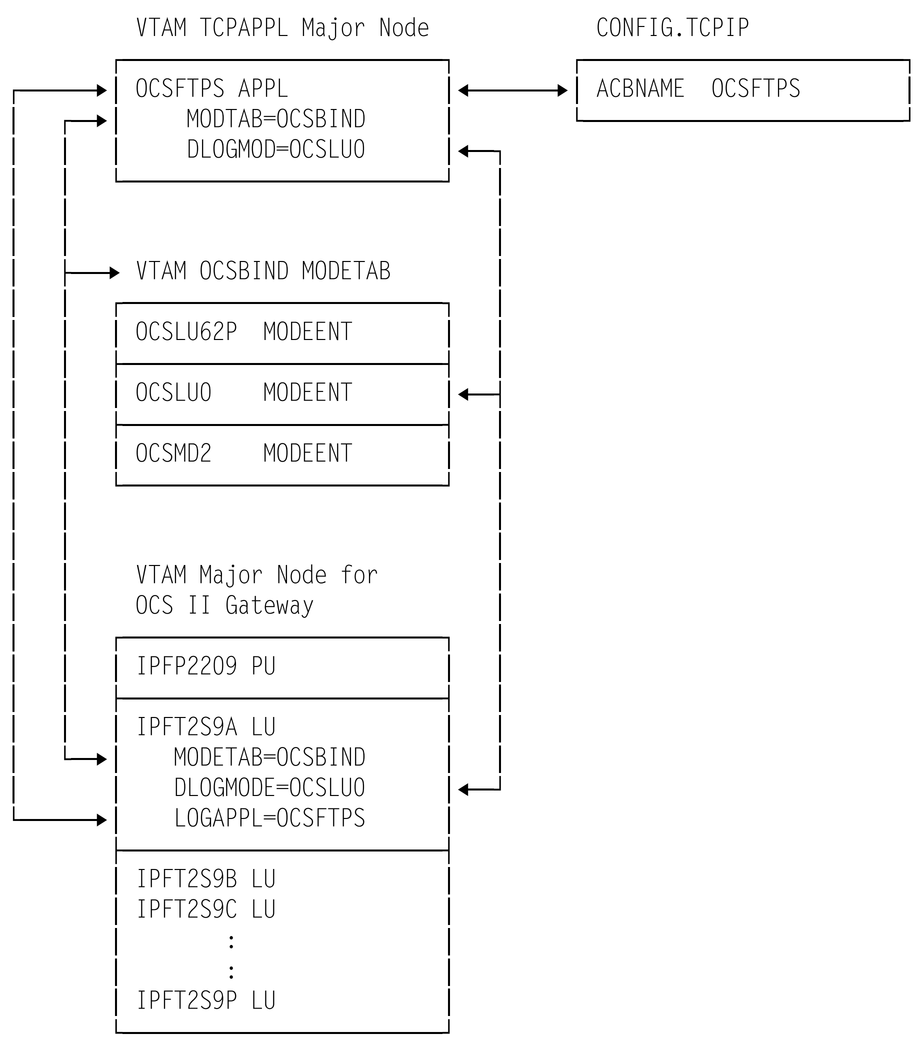


Figure 256. OC/FTP Server Definitions Summary

25.3 OC/FTP Client, OC/RSH Client and OC/LPD Definitions

25.3.1 VTAM and CICS Definitions

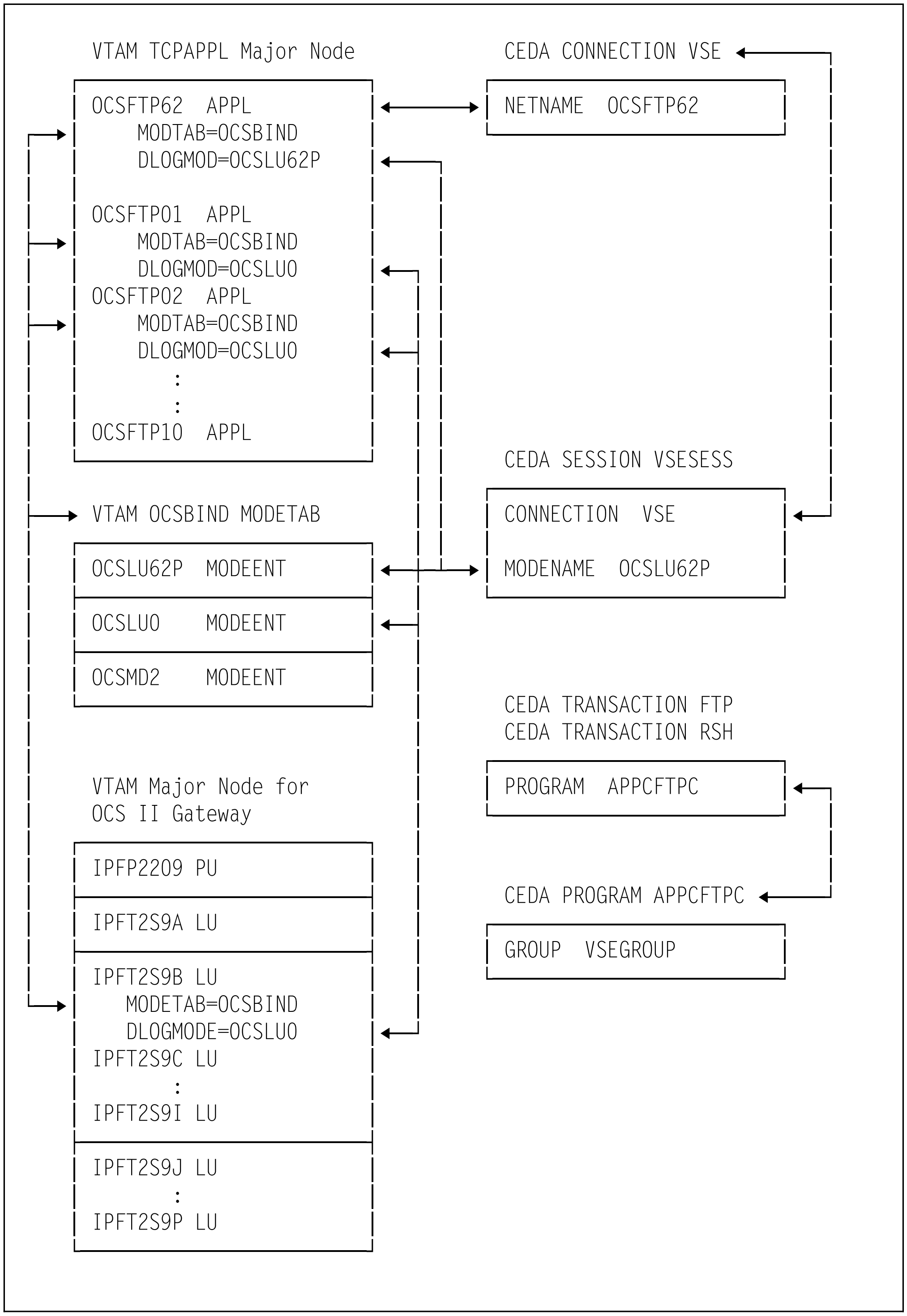


Figure 257. OC/FTP Client and OC/RSH Client Definitions Summary - VTAM and CICS

25.3.2 VTAM, OC/FTP Client and OCS II Gateway Definitions

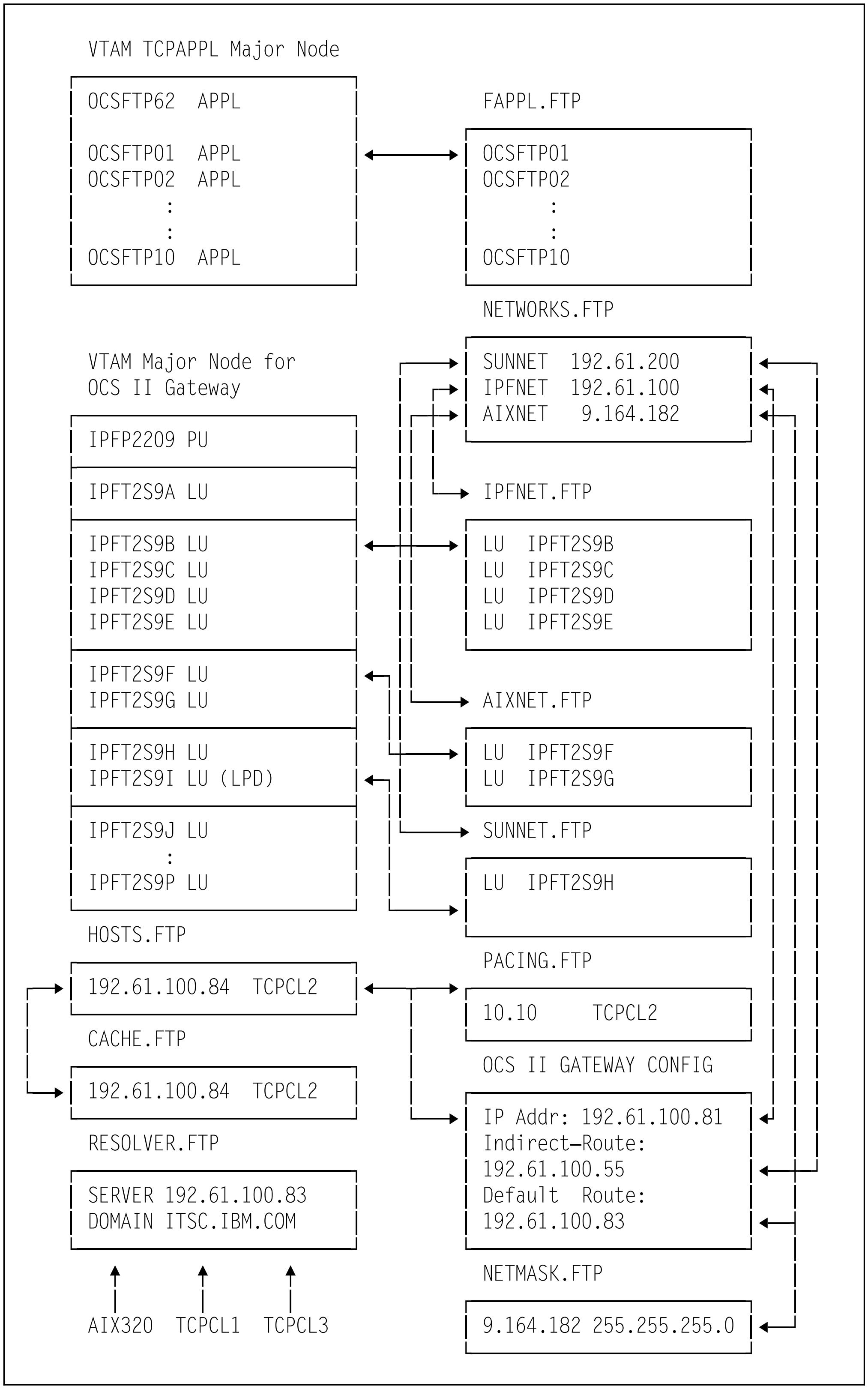


Figure 258. OC/FTP Client Definitions Summary - VTAM, FTP Client and OCS II Gateway

25.4 OC/TELNET FS Definitions

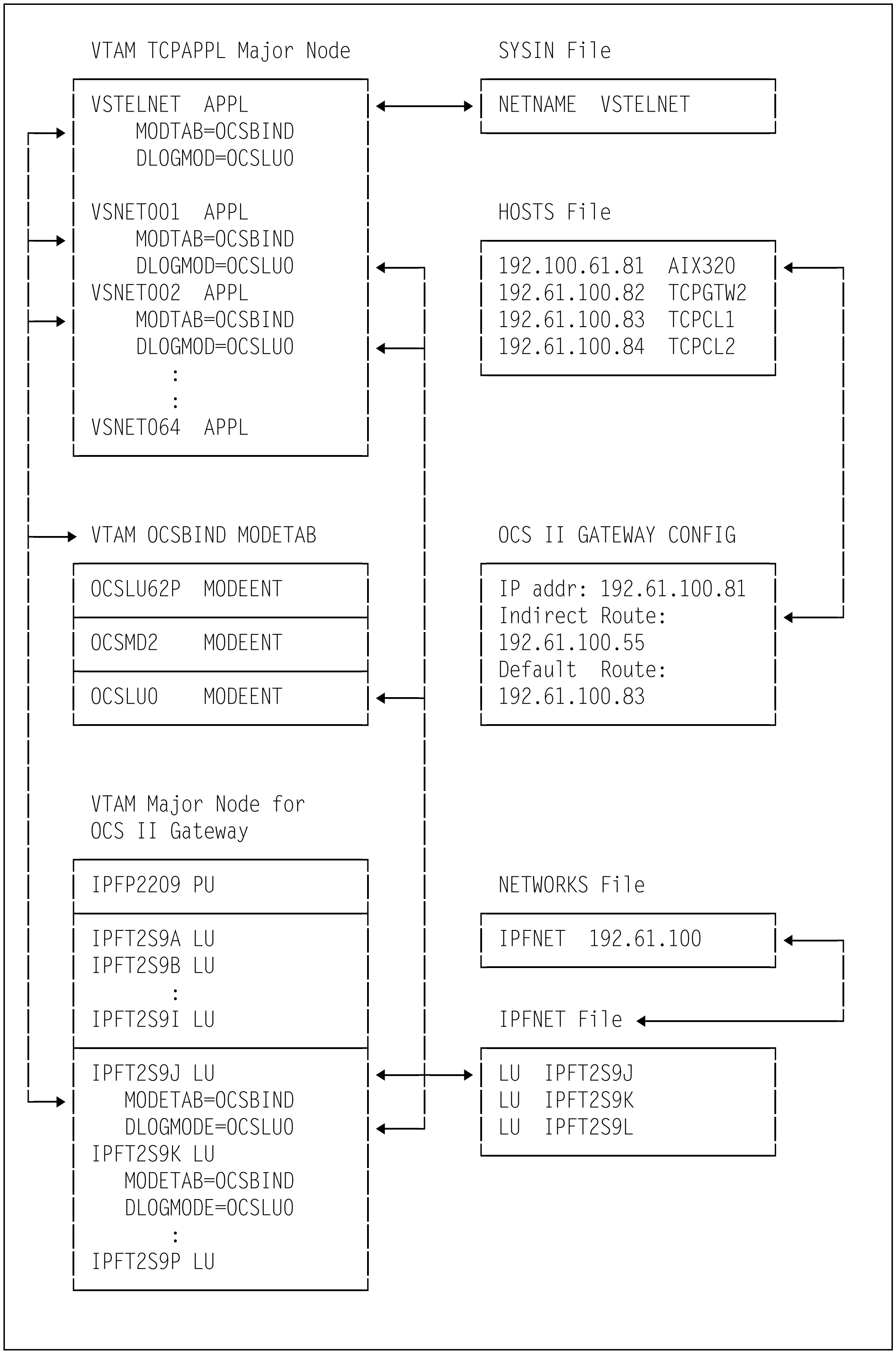


Figure 259. OC/TELNET FS Definitions Summary

25.5 Issues and Concerns

The following list summarizes the issues and concerns found during our implementation of the OCS II Gateway and its products as described in the previous chapters. The comments are based on the software levels and documentation available to us and described in 4.2, "Software" on page 32 and the list of OpenConnect Systems publications in the Bibliography.

- Security aspects of OC/FTP Server. Once an FTP client has been able to log in to OC/FTP Server, he can change access modes and working directories within VSE without having to pass any further access barriers (refer to 6.2.3, "FTP Server Customization" on page 67).
- A problem determination manual for all components of OCS TCP/IP for VSE is not available; this makes it difficult to perform any kind of more detailed analysis if problems arise.
- In general, the quality of the documentation, particularly with respect to topics related to VSE/ESA, can be improved by showing the relationships between the different systems and subsystems involved.

Part 3. Summary

The purpose of this part is to summarize the functions and limitations of the OpenConnect Systems products.

Note

The conclusions drawn in this part are based on the following levels of subject products:

- OCC Software V1.3
- OC/FTP Server V2.2.4.1
- OC/FTP Client V2.2.4.1
- OC/LPD V1.1.0
- OC/RSH Client V2.1.5
- OC/TELNET FS V4.11
- OC/SAM V2.1
- OCS II Gateway V3.7.5
- OC/Print Server for AIX V1.2

Future enhancements and/or modifications of the products may make this part or portions of it obsolete. Please check with the latest announcement letters and product manuals before making your purchase decision.

The OpenConnect Systems product provides gateway functions to allow VSE/ESA users to participate in TCP/IP networks.

This part provides a quick summary of the products′ functions and limitations. We also discuss a few user requirements to illustrate how the product addresses these situations. We hope that this explanation will help you to:

- Better understand the application aspects of the products
- Make a better selection of specific products based on your requirements

The following topics are covered:

- OpenConnect Systems Products Functions and Limitations Summary
- Selected User Requirements
- Conclusion

Chapter 26. OpenConnect Systems Products Functions and Limitations Summary

The following products are currently available within the IBM Cooperative Software Program (CSP) from OpenConnect Systems:

- OC/FTP Server
- OC/FTP Client
- OC/LPD
- OC/RSH Client
- OC/TELNET FS
- OC/SAM
- OCS II Gateway
- OCS II Telnet Server
- OCS Print Server for AIX

OCS II Gateway function is already explained in 3.1, “OCS II Gateway Functional Overview” on page 16. We will concentrate on the FTP, RSH and TELNET functions of the OpenConnect Systems products.

26.1 OpenConnect Systems Products Functions Summary

26.1.1 File Transfer Functions

OC/FTP Server allows TCP/IP FTP clients to initiate and perform FTP functions between VSE/ESA and TCP/IP nodes, accessing the following VSE/ESA files:

- VSAM ESDS and KSDS
- VSE Libraries
- POWER RDR, LST, and PUN queues

OC/FTP Server also allows TCP/IP FTP clients to issue POWER commands to manipulate POWER spool files.

OC/FTP Client allows VSE/ESA users to initiate and perform file transfer operations between VSE/ESA and TCP/IP nodes, accessing the TCP/IP FTP server′s file systems and VSAM files and VSE Libraries.

26.1.2 Remote Command Execution

OC/RSH Client allows you to pass commands from the VSE/ESA system to a remote TCP/IP host for execution.

26.1.3 Socket API

The OC/SAM provides a TCP/IP socket application program interface on VSE/ESA systems which enables the development of client/server applications in heterogeneous environments. In addition to the OC/SAM function library, the BSD 4.3 function library is available for C and ASM/370 programming.

26.1.4 Line Printer Daemon Functions

OC/LPD allows the LPR clients in the TCP/IP network to initiate and send print data to VSE/ESA. If requested, a response mail message is sent back through the gateway to the user via a standard SMTP well-known port. Together with OCS Print Server for AIX, a bidirectional printer solution is available.

26.1.5 TELNET Functions

OC/TELNET FS allows VSE/ESA users to initiate and establish terminal emulation sessions with TCP/IP TELNET servers.

OCS II Telnet Server allows TCP/IP TELNET clients to initiate and establish 3278 terminal emulation sessions with VSE/ESA.

26.2 Additional OpenConnect Systems Products Summary

26.2.1 OpenConnection for Channel Usage

OCC provides a direct channel attachment to the gateway. The installation of a small Token-Ring network between the mainframe attached 3174 controller and the RS/6000 gateway is avoided.

26.2.2 OCS Print Server for AIX

The OCS PRINT Server is an AIX process to emulate multiple 3287 CICS terminal printers simultaneously. The print output is automatically converted from EBCDIC to ASCII and routed to any local or remote printer defined on the AIX system.

26.2.3 OC://WebConnect Overview

OC://WebConnect provides users access to real-time mainframe application data from their desktops using any Java-capable Web browser.

Please check with OpenConnect Systems for additional information.

26.3 OpenConnect Systems Products Limitations

The OpenConnect Systems products that we tested have the following limitations:

- No programming capability in command files

  There is no logic statement available for the OC/FTP Client and OC/TELNET FS command files. Without the programming capability, file transfer or TELNET logon that require conditional decisions must be handled manually by the users.

  Please check with OpenConnect Systems for additional information.

Part 4. Job Streams and Definition Examples

Appendix A. Job Streams and Definition Examples for OCS Products

A.1 Product Installation Job Streams

```
* $$ JOB JNM=INSTN,DISP=D,PRI=3,                                         C
* $$ NTFY=YES,                                                           C
* $$ LDEST=*,                                                            C
* $$ CLASS=0
// JOB SCAN   SCAN OPTIONAL PRODUCT TAPE
// LIBDEF PHASE,SEARCH=(PRD1.BASE,IJSYSRS.SYSLIB)
* *
* *     -  SCAN PROGRAM TAPE
* *
// ASSGN SYS006,180
// EXEC DTRIPRE,PARM=¢ADDR=180¢
/*
/&
* $$ EOJ
```

```
* $$ JOB JNM=OCSINST,CLASS=0,DISP=D,NTFY=YES
// JOB OCSINST INSTALL OCS PRODUCT
// ASSGN SYS006,180
// MTC REW,SYS006
// EXEC MSHP
   INSTALL PRODUCT FROMTAPE -
   PRODUCTION INTO=lib.sublib
/*
// MTC RUN,SYS006
/&
* $$ EOJ
```

A.2 VSE VTAM Start Options List

```
* $$ JOB JNM=CATSTR1,DISP=D,PRI=3,NTFY=YES,LDEST=*,CLASS=0
// JOB CATSTR1  CATALOG VTAM START OPTION LIST
// EXEC LIBR,PARM=¢MSHP¢
ACCESS SUBLIB=PRD2.CONFIG
CATALOG     ATCSTR01.B        REPLACE=YES
SSCPID=22,                                                              C
HOSTSA=22,                                                              C
SSCPNAME=IPFV2B,                                                        C
HOSTPU=IPFVM22,                                                         C
NETID=DEIBMIPF,                                                         C
MAXSUBA=255,                                                            C
CONFIG=01,                                                              C
IOINT=0,                                                                C
SGALIMIT=0,                                                             C
BSBUF=(28,,,1),                                                         C
CRPLBUF=(60,,,1),                                                       C
LFBUF=(200,288,,20),            FROM 70 TO 200                          C
LPBUF=(12,,,6),                                                         C
SFBUF=(20,,,20),                                                        C
```

```
SPBUF=(210,,,32),                                                    C
VFBUF=204800,               FROM 102400 TO 204800                    C
VPBUF=528384,               FROM 446464 TO 528384                    C
XDBUF=(6,,,1)
/+
/*
/&
* $$ EOJ
```

A.3 VSE TCPSW.B Switched Major Node

```
* $$ JOB JNM=TCPSW,DISP=D,PRI=3,NTFY=YES,LDEST=(,RSCS),CLASS=0
* $$ LST CLASS=A
// JOB TCPSW
// EXEC LIBR,PARM=¢MSHP¢
ACCESS SUBLIB=PRD2.CONFIG
CATALOG   TCPSW.B   REPLACE=YES
*
*        LAN RELATED SWITCHED MAJOR NODE FOR RS/6000 GATEWAY AND PS/2
*
TCPSW          VBUILD TYPE=SWNET
*
*  PU AND LU DEFINITION FOR RS/6000 OCS GATEWAY
*
IPFP2209 PU    ADDR=04,MACADDR=400010101009,  LAA FOR RS/6000        C
               LANSW=YES,         LAN capable                        C
               IDBLK=017,         ID BLOCK REQUIRED BY OCS           C
               IDNUM=E0009,       IDNUM SET BY ITSC CONVENTION       C
               DISCNT=NO,             VTAM DOES NOT HANG UP          C
               PACING=0,VPACING=0,    NO PACING AS DEFAULT           C
               PUTYPE=2,              PU TYPE REQUIRED BY OCS        C
               MAXDATA=2057,          SET TO 2057 FOR 2K PIU         C
               ISTATUS=ACTIVE,                                       C
               LANACK=(01.0,1),   LAN ACKNOWLEDGEMENT VALUES         C
               LANINACT=02.0,     timer for inactive link station    C
               LANCON=(05.0,1),   LAN TIMER AND RETRY COUNT          C
               LANSDWDW=(2,1),    send window & window step          C
               LANRESP=(02.0,2),  Timer for connected state          C
               SAPADDR=4          service access point address
*
*   LU DEFINITION FOR RS/6000 GATEWAY
*
*   IPFT2S9A FOR OC/FTP SERVER
*   IPFT2S9B,C,D,E,F,G,H,I FOR OC/FTP CLIENT
*   IPFT2S9J,K,L FOR OC/TELNET CLIENT OUTBOUND SESSIONS
*   IPFT2S9M FOR TERMINAL PRINTER
*   IPFT2S9N,O FOR OC/TELNET SERVER INBOUND SESSIONS
*
IPFT2S9A LU    LOCADDR=1,DLOGMOD=OCSLU0,LOGAPPL=OCSFTPS,             C
               ISTATUS=ACTIVE,MODETAB=OCSBIND,                       C
               MDLTAB=VTMMDL,MDLENT=VSELU2A,                          C
               SSCPFM=USSSCS,USSTAB=VTMUSSTR
*
IPFT2S9B LU    LOCADDR=2,DLOGMOD=OCSLU0,                             C
               ISTATUS=ACTIVE,MODETAB=OCSBIND,                       C
               MDLTAB=VTMMDL,MDLENT=VSELU2A,                          C
               SSCPFM=USSSCS,USSTAB=VTMUSSTR
```

```
IPFT2S9C LU    LOCADDR=3,DLOGMOD=OCSLU0,                               C
               ISTATUS=ACTIVE,MODETAB=OCSBIND,                         C
               MDLTAB=VTMMDL,MDLENT=VSELU2A,                           C
               SSCPFM=USSSCS,USSTAB=VTMUSSTR
IPFT2S9D LU    LOCADDR=4,DLOGMOD=OCSLU0,                               C
               ISTATUS=ACTIVE,MODETAB=OCSBIND,                         C
               MDLTAB=VTMMDL,MDLENT=VSELU2A,                           C
               SSCPFM=USSSCS,USSTAB=VTMUSSTR
IPFT2S9E LU    LOCADDR=5,DLOGMOD=OCSLU0,                               C
               ISTATUS=ACTIVE,MODETAB=OCSBIND,                         C
               MDLTAB=VTMMDL,MDLENT=VSELU2A,                           C
               SSCPFM=USSSCS,USSTAB=VTMUSSTR
IPFT2S9F LU    LOCADDR=6,DLOGMOD=OCSLU0,                               C
               ISTATUS=ACTIVE,MODETAB=OCSBIND,                         C
               MDLTAB=VTMMDL,MDLENT=VSELU2A,                           C
               SSCPFM=USSSCS,USSTAB=VTMUSSTR
IPFT2S9G LU    LOCADDR=7,DLOGMOD=OCSLU0,                               C
               ISTATUS=ACTIVE,MODETAB=OCSBIND,                         C
               MDLTAB=VTMMDL,MDLENT=VSELU2A,                           C
               SSCPFM=USSSCS,USSTAB=VTMUSSTR
IPFT2S9H LU    LOCADDR=8,DLOGMOD=OCSLU0,                               C
               ISTATUS=ACTIVE,MODETAB=OCSBIND,                         C
               MDLTAB=VTMMDL,MDLENT=VSELU2A,                           C
               SSCPFM=USSSCS,USSTAB=VTMUSSTR
IPFT2S9I LU    LOCADDR=9,DLOGMOD=OCSLU0,                               C
               ISTATUS=ACTIVE,MODETAB=OCSBIND,                         C
               MDLTAB=VTMMDL,MDLENT=VSELU2A,                           C
               SSCPFM=USSSCS,USSTAB=VTMUSSTR
*
IPFT2S9J LU    LOCADDR=10,DLOGMOD=OCSLU0,                              C
               ISTATUS=ACTIVE,MODETAB=OCSBIND,                         C
               MDLTAB=VTMMDL,MDLENT=VSELU2A,                           C
               PACING=1,VPACING=2,                                     C
               SSCPFM=USSSCS,USSTAB=VTMUSSTR
IPFT2S9K LU    LOCADDR=11,DLOGMOD=OCSLU0,                              C
               ISTATUS=ACTIVE,MODETAB=OCSBIND,                         C
               MDLTAB=VTMMDL,MDLENT=VSELU2A,                           C
               PACING=1,VPACING=2,                                     C
               SSCPFM=USSSCS,USSTAB=VTMUSSTR
IPFT2S9L LU    LOCADDR=12,DLOGMOD=OCSLU0,                              C
               ISTATUS=ACTIVE,MODETAB=OCSBIND,                         C
               MDLTAB=VTMMDL,MDLENT=VSELU2A,                           C
               PACING=1,VPACING=2,                                     C
               SSCPFM=USSSCS,USSTAB=VTMUSSTR
IPFT2S9M LU    LOCADDR=13,DLOGMOD=OCSDSC,                              C
               ISTATUS=ACTIVE,MODETAB=OCSBIND,                         C
               MDLTAB=VTMMDL,MDLENT=VSELU2A,                           C
               PACING=1,VPACING=2,                                     C
               SSCPFM=USSSCS,USSTAB=VTMUSSTR
IPFT2S9N LU    LOCADDR=14,DLOGMOD=OCSMD2,                              C
               ISTATUS=ACTIVE,MODETAB=OCSBIND,                         C
               MDLTAB=VTMMDL,MDLENT=VSELU2A,                           C
               PACING=1,VPACING=2,                                     C
               SSCPFM=USSSCS,USSTAB=VTMUSSTR
IPFT2S9O LU    LOCADDR=15,DLOGMOD=OCSMD2,                              C
               ISTATUS=ACTIVE,MODETAB=OCSBIND,                         C
               MDLTAB=VTMMDL,MDLENT=VSELU2A,                           C
               PACING=1,VPACING=2,                                     C
               SSCPFM=USSSCS,USSTAB=VTMUSSTR
```

```
IPFT2S9P LU    LOCADDR=16,DLOGMOD=OCSMD2,                           C
               ISTATUS=ACTIVE,MODETAB=OCSBIND,                      C
               MDLTAB=VTMMDL,MDLENT=VSELU2A,                        C
               PACING=1,VPACING=2,                                  C
               SSCPFM=USSSCS,USSTAB=VTMUSSTR
*
/+
/*
/&
* $$ EOJ
```

A.4 VTAM APPL Major Node for OCS Products

```
* $$ JOB JNM=TCPAPPL,DISP=D,PRI=3,NTFY=YES,LDEST=*,CLASS=0
// JOB TCPAPPL  CATALOG VTAM BOOK
// EXEC LIBR,PARM=¢MSHP¢
ACCESS SUBLIB=PRD2.CONFIG
CATALOG   TCPAPPL.B         REPLACE=YES
TCPAPPL  VBUILD TYPE=APPL
DBDCCICS APPL  AUTH=(PASS,ACQ)
CICSSA22 APPL  AUTH=(PASS,ACQ,VPACE),PARSESS=YES,ACBNAME=CICSSA22,    C
               EAS=4000,MODETAB=CICSIPMT,APPC=NO,                     C
               SONSCIP=YES,VPACING=5
POWER    APPL  AUTH=(ACQ)
PNET     APPL  AUTH=(PASS,ACQ),VPACING=3,MODETAB=VTMLOGTB,DLOGMOD=PNET
IESWAITT APPL  AUTH=(NOACQ)
*
*  OCS FTP SERVER APPL
OCSFTPS  APPL  MODETAB=OCSBIND,DLOGMOD=OCSLU0,AUTH=(NVPACE,ACQ),EAS=2
*
*  OCS FTP CLIENT APPL
OCSFTP62 APPL  MODETAB=OCSBIND,DLOGMOD=OCSLU62P,AUTH=(ACQ),APPC=YES,  C
               PARSESS=YES,DSESLIM=10,DMINWNL=0,DMINWNR=10
OCSFTP01 APPL  MODETAB=OCSBIND,DLOGMOD=OCSLU0,AUTH=(NVPACE,ACQ),EAS=2
OCSFTP02 APPL  MODETAB=OCSBIND,DLOGMOD=OCSLU0,AUTH=(NVPACE,ACQ),EAS=2
OCSFTP03 APPL  MODETAB=OCSBIND,DLOGMOD=OCSLU0,AUTH=(NVPACE,ACQ),EAS=2
OCSFTP04 APPL  MODETAB=OCSBIND,DLOGMOD=OCSLU0,AUTH=(NVPACE,ACQ),EAS=2
OCSFTP05 APPL  MODETAB=OCSBIND,DLOGMOD=OCSLU0,AUTH=(NVPACE,ACQ),EAS=2
OCSFTP06 APPL  MODETAB=OCSBIND,DLOGMOD=OCSLU0,AUTH=(NVPACE,ACQ),EAS=2
OCSFTP07 APPL  MODETAB=OCSBIND,DLOGMOD=OCSLU0,AUTH=(NVPACE,ACQ),EAS=2
OCSFTP08 APPL  MODETAB=OCSBIND,DLOGMOD=OCSLU0,AUTH=(NVPACE,ACQ),EAS=2
OCSFTP09 APPL  MODETAB=OCSBIND,DLOGMOD=OCSLU0,AUTH=(NVPACE,ACQ),EAS=2
OCSFTP10 APPL  MODETAB=OCSBIND,DLOGMOD=OCSLU0,AUTH=(NVPACE,ACQ),EAS=2
*
*  OCS TELNET FS  APPL (64 SESSIONS)
VSTELNET APPL  AUTH=(ACQ,PASS),ACBNAME=VSTELNET
VSNET001 APPL  AUTH=(ACQ),EAS=2,ACBNAME=VSNET001
VSNET002 APPL  AUTH=(ACQ),EAS=2,ACBNAME=VSNET002
       :
       :
VSNET063 APPL  AUTH=(ACQ),EAS=2,ACBNAME=VSNET063
VSNET064 APPL  AUTH=(ACQ),EAS=2,ACBNAME=VSNET064
*
* ACB FOR SOCK$AM. THESE ACB¢S ARE USED BY BOTH TCP AND UDP.
UDPTEST1 APPL  ACBNAME=UDPTEST1,       APPLID FOR ACB                 C
               AUTH=(ACQ),             AUTHORIZED FOR ACQUIRING LUS   C
               EAS=2                   ESTIMATED CONCURRENT SESSIONS
```

```
UDPTEST2 APPL  ACBNAME=UDPTEST2,       APPLID FOR ACB                C
               AUTH=(ACQ),             AUTHORIZED FOR ACQUIRING LUS  C
               EAS=2                   ESTIMATED CONCURRENT SESSIONS
*
TCPTEST1 APPL  ACBNAME=TCPTEST1,       APPLID FOR ACB                C
               AUTH=(ACQ),             AUTHORIZED FOR ACQUIRING LUS  C
               EAS=2                   ESTIMATED CONCURRENT SESSIONS
TCPTEST2 APPL  ACBNAME=TCPTEST2,       APPLID FOR ACB                C
               AUTH=(ACQ),             AUTHORIZED FOR ACQUIRING LUS  C
               EAS=2                   ESTIMATED CONCURRENT SESSIONS
*
* THIS ACB IS USED BY THE SAMPLE LIBRARY MEMBER ¢VTAMOPER¢
VTAMOPR  APPL  ACBNAME=VTAMOPR,        APPLID FOR ACB                C
               AUTH=(SPO),             AUTHORIZED FOR ACQUIRING LUS  C
              EAS=2                   ESTIMATED CONCURRENT SESSIONS

/+
/*
/&
* $$ EOJ
```

A.5 VTAM USSTAB

```
* $$ JOB JNM=USSTAB,CLASS=0,DISP=D
// JOB USSTAB        CREATE USS TABLE FOR OCS AND XDOMAIN
* *****************************************************************
* *                                                               *
* *  JOBSTEP 1                                                    *
* *                                                               *
* *    IF THERE IS NO APPLICATION NAME CORRESPONDING TO A         *
* *    PARTICULAR VARIABLE, DELETE THE USSPARM STATEMENT          *
* *    CONTAINING THE VARIABLE; ALSO, DELETE THE USSCMD STATEMENT *
* *    PRECEEDING IT AND THE USSPARM STATEMENT FOLLOWING IT.      *
* *                                                               *
* *****************************************************************
// EXEC LIBR,PARM=¢MSHP¢
ACCESS SUBLIB=PRD2.CONFIG
CATALOG  VTMUSSCD.A   REPLACE=YES
*
A        USSCMD  CMD=A,REP=LOGON,FORMAT=BAL
         USSPARM PARM=P1,REP=APPLID,DEFAULT=DBDCCICS
         USSPARM PARM=P2,REP=DATA
*
B        USSCMD  CMD=B,REP=LOGON,FORMAT=BAL
         USSPARM PARM=P1,REP=APPLID,DEFAULT=CICSSA22
         USSPARM PARM=P2,REP=DATA
*
C        USSCMD  CMD=C,REP=LOGON,FORMAT=BAL
         USSPARM PARM=P1,REP=APPLID,DEFAULT=IPFA2GL3
         USSPARM PARM=P2,REP=DATA
*
D        USSCMD  CMD=D,REP=LOGON,FORMAT=BAL
         USSPARM PARM=P1,REP=APPLID,DEFAULT=VSTELNET
         USSPARM PARM=P2,REP=DATA
*
E        USSCMD  CMD=E,REP=LOGON,FORMAT=BAL
         USSPARM PARM=P1,REP=APPLID,DEFAULT=IPFA2VSC
```

```
         USSPARM PARM=P2,REP=DATA
*
F        USSCMD  CMD=F,REP=LOGON,FORMAT=BAL
         USSPARM PARM=P1,REP=APPLID,DEFAULT=CICSSA22
         USSPARM PARM=P2,REP=DATA
*
/+
/*
* *****************************************************************
* *                                                               *
* *  JOBSTEP 2                                                    *
* *                                                               *
* *    IF THERE IS NO APPLICATION NAME CORRESPONDING TO A         *
* *    PARTICULAR VARIABLE, REPLACE IT WITH ________ TO INDICATE  *
* *    THAT THERE IS NO NAME.                                     *
* *                                                               *
* *****************************************************************
// EXEC LIBR,PARM=¢MSHP¢
ACCESS SUBLIB=PRD2.CONFIG
CATALOG  VTMUSSTZ.A   REPLACE=YES
*
* THE FOLLOWING MENU WILL BE DISPLAYED ON SNA TERMINALS ONLY
*
*
         DC    X¢15¢                NEW LINE (ROW 5)
         DC    CL9¢ ¢
         DC    CL2¢A ¢
         DC    CL8¢DBDCCICS¢
*
         DC    X¢15¢                NEW LINE (ROW 6)
         DC    CL9¢ ¢
         DC    CL2¢B ¢
         DC    CL8¢CICSSA22¢
*
         DC    X¢15¢                NEW LINE (ROW 7)
         DC    CL9¢ ¢
         DC    CL2¢C ¢
         DC    CL8¢VMESASA2¢
*
         DC    X¢15¢                NEW LINE (ROW 8)
         DC    CL9¢ ¢
         DC    CL2¢D ¢
         DC    CL8¢VSTELENT¢
*
         DC    X¢15¢                NEW LINE (ROW 9)
         DC    CL9¢ ¢
         DC    CL2¢E ¢
         DC    CL8¢VM/ESA12¢
*
         DC    X¢15¢                NEW LINE (ROW 10)
         DC    CL9¢ ¢
         DC    CL2¢F ¢
         DC    CL8¢________¢
*
/+
/*
* *****************************************************************
* *                                                               *
* *  JOBSTEP 3                                                    *
```

```
* *                                                              *
* *    IF THERE IS NO APPLICATION NAME CORRESPONDING TO A        *
* *    PARTICULAR VARIABLE, REPLACE IT WITH ________ TO INDICATE  *
* *    THAT THERE IS NO NAME.                                    *
* *                                                              *
* ****************************************************************
// EXEC LIBR,PARM=¢MSHP¢
ACCESS SUBLIB=PRD2.CONFIG
CATALOG VTMUSSTX.A REPLACE=YES
*
* THE FOLLOWING MENU WILL BE DISPLAYED ON NON-SNA TERMINALS ONLY
*
*
         DC    X¢11¢                    SET BUFFER ADDRESS ORDER
         DC    X¢C5C9¢                  ROW 5 COLUMN 10
         DC    X¢1D¢                    START FIELD ORDER
         DC    X¢F8¢                    PROTECT SKIP INTENSIFIED ATTRIBUTE
         DC    CL2¢A ¢
         DC    X¢1D¢                    START FIELD
         DC    X¢F0¢                    PROTECT SKIP NORMAL
         DC    CL8¢DBDCCICS¢
*
         DC    X¢11¢                    SET BUFFER ADDRESS ORDER
         DC    X¢C6D9¢                  ROW 6 COLUMN 10
         DC    X¢1D¢                    START FIELD ORDER
         DC    X¢F8¢                    PROTECT SKIP INTENSIFIED ATTRIBUTE
         DC    CL2¢B ¢
         DC    X¢1D¢                    START FIELD
         DC    X¢F0¢                    PROTECT SKIP NORMAL
         DC    CL8¢CICSSA22¢
*
         DC    X¢11¢                    SET BUFFER ADDRESS ORDER
         DC    X¢C7E9¢                  ROW 7 COLUMN 10
         DC    X¢1D¢                    START FIELD ORDER
         DC    X¢F8¢                    PROTECT SKIP INTENSIFIED ATTRIBUTE
         DC    CL2¢C ¢
         DC    X¢1D¢                    START FIELD
         DC    X¢F0¢                    PROTECT SKIP NORMAL
         DC    CL8¢VMESASA2¢
*
         DC    X¢11¢                    SET BUFFER ADDRESS ORDER
         DC    X¢C8F9¢                  ROW 8 COLUMN 10
         DC    X¢1D¢                    START FIELD ORDER
         DC    X¢F8¢                    PROTECT SKIP INTENSIFIED ATTRIBUTE
         DC    CL2¢D ¢
         DC    X¢1D¢                    START FIELD
         DC    X¢F0¢                    PROTECT SKIP NORMAL
         DC    CL8¢VSTELNET¢
*
         DC    X¢11¢                    SET BUFFER ADDRESS ORDER
         DC    X¢4AC9¢                  ROW 9 COLUMN 10
         DC    X¢1D¢                    START FIELD ORDER
         DC    X¢F8¢                    PROTECT SKIP INTENSIFIED ATTRIBUTE
         DC    CL2¢E ¢
         DC    X¢1D¢                    START FIELD
         DC    X¢F0¢                    PROTECT SKIP NORMAL
         DC    CL8¢VM/ESA12¢
*
         DC    X¢11¢                    SET BUFFER ADDRESS ORDER
```

```
         DC    X¢4BD9¢               ROW 10 COLUMN 10
         DC    X¢1D¢                 START FIELD ORDER
         DC    X¢F8¢                 PROTECT SKIP INTENSIFIED ATTRIBUTE
         DC    CL2¢F ¢
         DC    X¢1D¢                 START FIELD
         DC    X¢F0¢                 PROTECT SKIP NORMAL
         DC    CL8¢________¢
*
/+
/*
// LIBDEF *,SEARCH=(PRD1.BASE,PRD2.CONFIG),TEMP
// LIBDEF PHASE,CATALOG=PRD2.CONFIG
// OPTION CATAL
   PHASE VTMUSSTR,*
// EXEC ASSEMBLY
         PRINT   NOGEN
VTMUSSTR USSTAB  TABLE=STDTRANS,FORMAT=DYNAMIC
*
         COPY    VTMUSSCD
*
TEST     USSCMD  CMD=TEST,REP=IBMTEST,FORMAT=BAL
         USSPARM PARM=P1,DEFAULT=10
         USSPARM PARM=P2,DEFAULT=OK
*
MESSAGES USSMSG  MSG=0,TEXT=¢COMMAND ACCEPTED¢
         USSMSG  MSG=1,BUFFER=M1
         USSMSG  MSG=2,BUFFER=M1
         USSMSG  MSG=3,TEXT=¢ERROR IN VTMUSSTR. PRESS ENTER¢
         USSMSG  MSG=4,TEXT=¢APPLICATION NOT ACTIVATED. PRESS ENTER¢
         USSMSG  MSG=5,BUFFER=M1
         USSMSG  MSG=6,TEXT=¢LOGON ALREADY PENDING¢
         USSMSG  MSG=7,TEXT=¢%(1) UNABLE TO ESTABLISH SESSION - %(2) F*
               AILED WITH SENSE %(3)¢
         USSMSG  MSG=8,TEXT=¢INSUFFICIENT STORAGE¢
         USSMSG  MSG=9,TEXT=¢MAGNETIC CARD DATA ERROR¢
         USSMSG  MSG=10,BUFFER=M1
         USSMSG  MSG=12,TEXT=¢REQUIRED PARAMETER OMITTED¢
         USSMSG  MSG=13,TEXT=¢IBMECHO%¢
*
STDTRANS DC      128AL1(*-STDTRANS)
         DC      X¢80C1C2C3C4C5C6C7C8C98A8B8C8D8E8F¢
         DC      X¢90D1D2D3D4D5D6D7D8D99A9B9C9D9E9F¢
         DC      X¢A0A1E2E3E4E5E6E7E8E9AAABACADAEAF¢
         DC      X¢B0B1B2B3B4B5B6B7B8B9BABBBCBDBEBF¢
         DC      X¢C0C1C2C3C4C5C6C7C8C9CACBCCCDCECF¢
         DC      X¢D0D1D2D3D4D5D6D7D8D9DADBDCDDDEDF¢
         DC      X¢E0E1E2E3E4E5E6E7E8E9EAEBECEDEEEF¢
         DC      X¢F0F1F2F3F4F5F6F7F8F9FAFBFCFDFEFF¢
END      USSEND
*
M1       DC    AL2(M1E-M1S)
M1S      DC    X¢15¢                 NEW LINE (ROW 1)
*
* VTMUSSTR       VTAM APPLICATION SELECTION MENU
*
         DC    CL1¢ ¢
         DC    CL8¢VSEESA22¢
         DC    CL12¢ ¢
         DC    C¢VTAM APPLICATION SELECTION MENU¢
```

```
*
* ENTER THE CHARACTER OF YOUR SELECTION AND PRESS THE ENTER KEY:
* (MIXED-CASE)
*
         DC    X¢15¢                NEW LINE (ROW 2)
         DC    X¢15¢                NEW LINE (ROW 3)
         DC    CL3¢ ¢
         DC    X¢C595A3859940A3888540838881998183A3859940968640¢
         DC    X¢A896A49940A285938583A38996954081958440979985A2A240¢
         DC    X¢A3888540C5D5E3C5D9409285A87A¢
*
         DC    X¢15¢                NEW LINE (ROW 4)
*
         COPY  VTMUSSTZ
*
         DC    9X¢15¢               SKIP 9 LINES (ROW 19)
         DC    C¢ ==> ¢
M1E      EQU   *
         END
/*
// EXEC LNKEDT
// OPTION CATAL
   PHASE VTMUSSTB,*
// EXEC ASSEMBLY
         PRINT   NOGEN
VTMUSSTB USSTAB  TABLE=STDTRANS,FORMAT=DYNAMIC
*
         COPY    VTMUSSCD
*
TEST     USSCMD  CMD=TEST,REP=IBMTEST,FORMAT=BAL
         USSPARM PARM=P1,DEFAULT=10
         USSPARM PARM=P2,DEFAULT=OK
*
MESSAGES USSMSG  MSG=0,TEXT=¢COMMAND ACCEPTED¢
         USSMSG  MSG=1,BUFFER=M1
         USSMSG  MSG=2,BUFFER=M1
         USSMSG  MSG=3,TEXT=¢ERROR IN VTMUSSTB. PRESS ENTER¢
         USSMSG  MSG=4,TEXT=¢APPLICATION NOT ACTIVATED. PRESS ENTER¢
         USSMSG  MSG=5,BUFFER=M1
         USSMSG  MSG=6,TEXT=¢LOGON ALREADY PENDING¢
         USSMSG  MSG=7,TEXT=¢%(1) UNABLE TO ESTABLISH SESSION - %(2) F*
               AILED WITH SENSE %(3)¢
         USSMSG  MSG=8,TEXT=¢INSUFFICIENT STORAGE¢
         USSMSG  MSG=9,TEXT=¢MAGNETIC CARD DATA ERROR¢
         USSMSG  MSG=10,BUFFER=M1
         USSMSG  MSG=12,TEXT=¢REQUIRED PARAMETER OMITTED¢
         USSMSG  MSG=13,TEXT=¢IBMECHO%¢
*
STDTRANS DC      128AL1(*-STDTRANS)
         DC      X¢80C1C2C3C4C5C6C7C8C98A8B8C8D8E8F¢
         DC      X¢90D1D2D3D4D5D6D7D8D99A9B9C9D9E9F¢
         DC      X¢A0A1E2E3E4E5E6E7E8E9AAABACADAEAF¢
         DC      X¢B0B1B2B3B4B5B6B7B8B9BABBBCBDBEBF¢
         DC      X¢C0C1C2C3C4C5C6C7C8C9CACBCCCDCECF¢
         DC      X¢D0D1D2D3D4D5D6D7D8D9DADBDCDDDEDF¢
         DC      X¢E0E1E2E3E4E5E6E7E8E9EAEBECEDEEEF¢
         DC      X¢F0F1F2F3F4F5F6F7F8F9FAFBFCFDFEFF¢
END      USSEND
*
```

```
M1       DC    AL2(M1E-M1S)
M1S      DC    X¢F5¢                    ERASE WRITE COMMAND
         DC    X¢C7¢                    WCC ALARM
*
* VTMUSSTB       VTAM APPLICATION SELECTION MENU
*
         DC    X¢11¢                    SET BUFFER ADDRESS ORDER
         DC    X¢40C1¢                  ROW 1 COLUMN 2
         DC    X¢1D¢                    START FIELD
         DC    X¢F0¢                    PROTECT SKIP NORMAL
         DC    CL8¢VSEESA22¢
         DC    X¢11¢                    SET BUFFER ADDRESS ORDER
         DC    X¢40D7¢                  ROW 1 COLUMN 24
         DC    X¢1D¢                    START FIELD ORDER
         DC    X¢F8¢                    PROTECT SKIP INTENSIFIED ATTRIBUTE
         DC    C¢VTAM APPLICATION SELECTION MENU¢
*
* ENTER THE CHARACTER OF YOUR SELECTION AND PRESS THE ENTER KEY:
* (MIXED-CASE)
*
         DC    X¢11¢                    SET BUFFER ADDRESS ORDER
         DC    X¢C2E3¢                  ROW 3 COLUMN 4
         DC    X¢1D¢                    START FIELD
         DC    X¢F0¢                    PROTECT SKIP NORMAL
         DC    X¢C595A3859940A38885408388819981A3859940968640¢
         DC    X¢A896A49940A285938583A389969540819584409799985A2A240¢
         DC    X¢A3888540C5D5E3C5D9409285A87A¢
*
         COPY  VTMUSSTX
*
         DC    X¢11¢                    SET BUFFER ADDRESS ORDER
         DC    X¢D661¢                  ROW 19 COLUMN 2
         DC    X¢1D¢                    START FIELD
         DC    X¢F8¢                    PROTECT SKIP INTENSIFIED ATTRIBUTE
         DC    C¢==> ¢
         DC    X¢1D¢                    START FIELD ORDER
         DC    X¢40¢                    UNPROTECTED NORMAL ATTRIBUTE
         DC    X¢13¢                    INSERT CURSOR ORDER
         DC    X¢3C¢                    REPEAT TO ADDRESS ORDER
         DC    X¢D7F0¢                  ROW 20 COLUMN 1
         DC    C¢ ¢
         DC    X¢1D¢                    START FIELD ORDER
         DC    X¢F0¢                    PROTECT SKIP NORMAL ATTRIBUTE
M1E      EQU   *
         END
/*
// EXEC LNKEDT
/&
* $$ EOJ
```

A.6 OCSBIND MODETAB for OCS Products

```
* $$ JOB JNM=OCSBIND,CLASS=0,DISP=D,NTFY=YES
* $$ LST CLASS=Q,DISP=H
// JOB OCSBIND  ASSEMBLE
// LIBDEF *,CATALOG=PRD2.CONFIG
// LIBDEF SOURCE,SEARCH=PRD1.BASE
// OPTION CATAL,LIST
// EXEC ASMA90,SIZE=(ASMA90,50K)
         PUNCH   ¢ CATALOG  OCSBIND.OBJ    REPLACE=YES¢
         PUNCH   ¢ PHASE    OCSBIND,*¢
         PRINT   NOGEN
         EJECT
OCSBIND  MODETAB
         EJECT
  TITLE ¢FTP-CLIENT¢
*********************************************************************
*                                                                   *
*           FTP CLIENT FOR OCS TCP/IP                               *
*                                                                   *
*********************************************************************
OCSLU62P MODEENT LOGMODE=OCSLU62P,                                     C
               TYPE=X¢00¢,                                             C
               FMPROF=X¢13¢,                                           C
               TSPROF=X¢07¢,                                           C
               PRIPROT=X¢B0¢,                                          C
               SECPROT=X¢B0¢,                                          C
               COMPROT=X¢50A1¢,                                        C
               RUSIZES=X¢8585¢,                                        C
               SSNDPAC=X¢07¢,                                          C
               SRCVPAC=X¢07¢,                                          C
               PSNDPAC=X¢07¢,                                          C
               PSERVIC=X¢060200000000000000002300¢
  TITLE ¢FTP-SERVER¢
*********************************************************************
*                                                                   *
*           FTP SERVER FOR OCS TCP/IP                               *
*           TELNET FS  FOR OCS TCP/IP                               *
*                                                                   *
*********************************************************************
OCSLU0   MODEENT LOGMODE=OCSLU0,                                       C
               TYPE=X¢00¢,                                             C
               FMPROF=X¢06¢,                                           C
               TSPROF=X¢03¢,                                           C
               PRIPROT=X¢00¢,                                          C
               SECPROT=X¢00¢,                                          C
               COMPROT=X¢0000¢,                                        C
               RUSIZES=X¢F8F8¢,                                        C
               PSERVIC=X¢000000000000000000000000¢
  TITLE ¢TELNET¢
*********************************************************************
*                                                                   *
*           TELNET CLIENT FOR OCS TCP/IP                            *
*                                                                   *
*********************************************************************
OCSMD2   MODEENT LOGMODE=OCSMD2,                                       C
```

```
               FMPROF=X¢03¢,                                          C
               TSPROF=X¢03¢,                                          C
               PRIPROT=X¢B1¢,                                         C
               SECPROT=X¢90¢,                                         C
               COMPROT=X¢3080¢,                                       C
               RUSIZES=X¢87C7¢,                                       C
               PSERVIC=X¢028000000000000000000200¢
  TITLE ¢Printer Client¢
*********************************************************************
*                                                                   *
*            PRINTER CLIENT FOR OCS TCP/IP                          *
*                                                                   *
*********************************************************************
OCSSCS   MODEENT LOGMODE=OCSSCS,                                      C
               FMPROF=X¢03¢,                                          C
               TSPROF=X¢03¢,                                          C
               PRIPROT=X¢B1¢,                                         C
               SECPROT=X¢90¢,                                         C
               COMPROT=X¢3080¢,                                       C
               RUSIZES=X¢8587¢,                                       C
               SRCVPAC=X¢02¢,                                         C
               PSNDPAC=X¢04¢,                                         C
               PSERVIC=X¢01000000E100000000000000¢
OCSDSC   MODEENT LOGMODE=OCSDSC,                                      C
               FMPROF=X¢03¢,                                          C
               TSPROF=X¢03¢,                                          C
               PRIPROT=X¢B1¢,                                         C
               SECPROT=X¢90¢,                                         C
               COMPROT=X¢3080¢,                                       C
               RUSIZES=X¢8587¢,                                       C
               SRCVPAC=X¢02¢,                                         C
               PSNDPAC=X¢04¢,                                         C
               PSERVIC=X¢030000000000000000000200¢
  TITLE ¢SAM-APPLICATION¢
*********************************************************************
*                                                                   *
*            SAM PROGRAMMING                                        *
*                                                                   *
*********************************************************************
MTKLU62X MODEENT LOGMODE=MTKLU62X,                                    C
               TYPE=X¢00¢,                                            C
               FMPROF=X¢13¢,                                          C
               TSPROF=X¢07¢,                                          C
               PRIPROT=X¢B0¢,                                         C
               SECPROT=X¢B0¢,                                         C
               COMPROT=X¢50D1¢,                                       C
               RUSIZES=X¢8989¢,                                       C
               SSNDPAC=X¢07¢,                                         C
               SRCVPAC=X¢07¢,                                         C
               PSNDPAC=X¢07¢,                                         C
               PSERVIC=X¢060200000000000000002C00¢
         MODEEND
         END                        , END OF IESINCLM
/*
// EXEC LNKEDT
/*
/&
* $$ EOJ
```

Appendix B. Special Notices

This document describes how VSE/ESA hosts can be integrated into Transmission Control Protocol/Internet Protocol (TCP/IP) networks. It is intended primarily for customer personnel and IBM technical professionals with networking responsibilities in heterogeneous environments.

The purpose of this publication is to help with the installation and customization of OCS TCP/IP for VSE.

See the PUBLICATIONS section of the IBM Programming Announcement for the products listed below, for more information about what publications are considered to be product documentation.

- ACF/VTAM Version 4.2.0
- OpenConnection for Channel (OCC)
- OpenConnect/File Transfer Program Client (OC/FTPC)
- OpenConnect/File Transfer Program Server (OC/FTPS)
- OpenConnect/Line Printer Daemon (OC/LPD)
- OpenConnect/Remote Shell (OC/RSH)
- OpenConnect/TELNET Client Full Screen (OC/TELNET Client FS)
- OpenConnect/Socket Access Method (OC/SAM)
- OpenConnect Server II Gateway for RISC/6000 (OCS II Gateway)
- OpenConnect Systems Print Server for UNIX (OCS Print Server)
- VM/ESA Release 2.1
- VSE/ESA Version 2.1.2

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.

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.

The following document contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples contain the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

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:

ACF/VTAM	Advanced Peer-to-Peer Networking
AIX	APPN
AS/400	CICS
CICS/VSE	CUA
Current	DB2
ESCON	IBM
ILE	MVS/ESA
MVS/XA	OS/2
Presentation Manager	PS/2
PSL	RETAIN
RISC System/6000	RMF
RS/6000	S/370
SAA	SKI
SP1	SXM
System/370	VM/ESA
VSE/ESA	VTAM
XT	

The following terms are trademarks of other companies:

APOLLO	Hewlett-Packard Company
C-bus	Corollary, Inc.
DEC VT100	Digital Equipment Corporation
DEC VT200	Digital Equipment Corporation
DOS	Microsoft Corporation
Ethernet	Xerox Corporation
HP	Hewlett-Packard Company
Microsoft	Microsoft Corporation
Network File System	Sun Microsystems, Inc.
NFS	Sun Microsystems, Inc.
OpenConnect	Registered trademark of OpenConnect Systems, Inc.
OpenConnect Product Names	OpenConnect Systems, Inc.
PC Direct	Ziff Communications Company (used by IBM Corporation under license)
SunOS	Sun Microsystems, Inc.
SX	Intel Corporation
UNIX	X/Open Company Ltd. (registered trademark in the United States and other countries)
Windows, Windows 95 logo	Microsoft Corporation

386 Intel Corporation

Appendix C. Related Publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this redbook.

C.1 International Technical Support Organization Publications

For information on ordering these ITSO publications see "How To Get ITSO Redbooks" on page 359.

- *TCP/IP Solutions for VSE/ESA - Implementation Guide*, GG24-4195
- *TCP/IP Tutorial and Technical Overview*, GG24-3376
- *Cross Domain Networking in VM/ESA 2.0 and VSE/ESA 1.2 Environments Implementation Guide*, GG24-4174
- *TCP/IP V.2.0 for OS/2 Installation and Interoperability*, GG24-3531

A complete list of International Technical Support Organization publications, known as redbooks, with a brief description of each, may be found in:

International Technical Support Organization Bibliography of Redbooks, GG24-3070.

C.2 Other Publications

These publications are also relevant as further information sources.

IBM Publications

- *TCP/IP for OS/2 V2.0 Installation and Administration*, SC31-6075
- *Internetworking with TCP/IP, Volume I*, SC31-6144
- *Internetworking with TCP/IP, Volume II*, SC31-6145
- *VSE/ESA 2.1.2 Messages and Codes*, SC33-6607
- *VSE/ESA 2.1.2 System Control Statements*, SC33-6613
- *VSE/VSAM User's Guide and Application Programming*, SC33-6632
- *VTAM V4R2 Messages and Codes*, SC31-6493
- *VTAM V4R2 Operation*, SC31-6495
- *VTAM V4R2 Resource Definition Reference*, SC31-6498
- *AIX System User's Guide: Communication and Networks*, GC23-2523

OpenConnect Systems Publications

- *OpenConnection for Channel, Installation, Administration, and Operations Guide*, 350-0466-101
- *OpenConnect/FTP Client and FTP Server, VSE Installation and User Guide*, 350-0382-101
- *OpenConnect/Line Printer Daemon for MVS and VSE Installation, Operation, and Administration User's Guide*, 350-0443-101
- *OpenConnect Systems Print Server for UNIX - User's Guide*, 350-0439-101

- *OpenConnect/Remote File Shell Client Installation and User Guide, VSE/ESA*, 350-0201-101
- *OpenConnect Systems IBM S/370 Software Gen Guide*, 350-0007-101
- *OC/Socket Access Method Installation and Programming Guide, VSE/ESA*, 35101
- *OpenConnect Server II for RISC/6000 Installation and Operation Guide*, 350-0285-101.15
- *OpenConnect/TELNET Server Manager, Administration and User's Guide*, 350-0193-101
- *OpenConnect/TELNET Client Full Screen Client Full Screen Installation Guide*, 350-0146-101
- *OpenConnect/TELNET Client Full Screen Client Full Screen User and Administration Guide*, 350-0145-101

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 URL 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 lists of redbooks:

  ```
  TOOLS SENDTO WTSCPOK TOOLS REDBOOKS GET REDBOOKS CATALOG
  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:

  ```
  TOOLS SENDTO WTSCPOK TOOLS ZDISK GET ITSOREGI 1996
  ```

  For a list of product area specialists in the ITSO:

  ```
  TOOLS SENDTO WTSCPOK TOOLS ZDISK GET ORGCARD PACKAGE
  ```

- **Redbooks Home Page on the World Wide Web**

  http://w3.itso.ibm.com/redbooks

- **IBM Direct Publications Catalog on the World Wide Web**

  http://www.elink.ibmlink.ibm.com/pbl/pbl

  IBM employees may obtain LIST3820s of redbooks from this page.

- **ITSO4USA 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.ibmlink.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** (Do not send credit card information over the Internet) — send orders to:

	IBMMAIL	**Internet**
In United States:	usib6fpl at ibmmail	usib6fpl@ibmmail.com
In Canada:	caibmbkz at ibmmail	lmannix@vnet.ibm.com
Outside North America:	bookshop at 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	IBM Publications	IBM Direct Services
Publications Customer Support	144-4th Avenue, S.W.	Sortemosevej 21
P.O. Box 29570	Calgary, Alberta T2P 3N5	DK-3450 Allerød
Raleigh, NC 27626-0570	Canada	Denmark
USA		

- **Fax** — send orders to:

United States (toll free)	1-800-445-9269
Canada (toll free)	1-800-267-4455
Outside North America	(+45) 48 14 2207 (long distance charge)

- **1-800-IBM-4FAX (United States)** or **(+1) 415 855 43 29 (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

- **On the World Wide Web**

Redbooks Home Page	http://www.redbooks.ibm.com
IBM Direct Publications Catalog	http://www.elink.ibmlink.ibm.com/pbl/pbl

- **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.ibmlink.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

- **Please put me on the mailing list for updated versions of the IBM Redbook Catalog.**

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.

DO NOT SEND CREDIT CARD INFORMATION OVER THE INTERNET.

Index

Numerics

3174 41

A

ACB parameter 113
ACBNAME parameter 68, 136
ACCESSMETHOD parameter 89
ACQ parameter 66, 86, 111
adapter name parameter 176
additional configuration members 115
ADDR parameter 165
AIX Control Character 252
alternate PORT number 171
alternate port number for aix 186
APPC parameter 86
APPCFTPC program 91
APPL major node 66, 83, 111, 124, 135
APPL statement 86
application layer 3
apply site protection 115
ASCII Control Character 252
AUTOCONNECT ALL parameter 90
automatic startup OCS II 189

B

batch interface for RSH 125
bibliography 357
BUFND parameter 94
BUFSP parameter 94

C

CACHE member 99
call type parameter 176
CATALOG parameter 65
CD command 224
CDRM major node 46, 49
CDRSC major node 49
CEDA 90, 127, 202, 203
CICS
  CEDA 90, 127, 202, 203
  connection VSE 88
  customization for FTP 88
  customization for RSH 126
  session definition 89
CISIZE parameter 94
client installation for FTP 81
client installation for RSH 123
CLSDSP parameter 94
config file 197
CONFIG member 67
CONNECTION parameter 89, 90
connection VSE 88
Connection-oriented Protocol 308
Connectionless Protocol 306
Console log of online OC/FTP Client 267
Control Character during file transfer 252
CTCA major node 45, 48
CUADDR parameter 46, 53

D

D= operand 114
datalink buffer size 163
datalink encoding 163
DDNAME= operand 114
DDSN parameter 94
DEFAULT command 82, 263
defaults file 196
device definitions 44
DFLTSYS file 141
DFLTUSR file 141
DIR command 224
display HAT Table entries 299
display OC/FTP Server user and status 220
display OCSNA process id 191
DLIB parameter 94
DLOGMOD parameter 66, 86, 111, 136
DMBRN parameter 94
DMBRT parameter 94
DMINWNL parameter 86
DMINWNR parameter 86
DNS 101
Domain Name Server 101
DSESLIM parameter 86
DSLIB parameter 94
DYNALL member 82, 92
dynamic partition 83, 111, 124

E

EAS= parameter 66, 86, 111, 136
EBCDIC Control Character 252
end of record character 252
environment variable 189
eps_start command 205
eps_status command 205
eps_stop command 205
Error
  get/mget, put/mput 236
  PASSWD member 71
  RESOVER member 101
  SITE member 69
ETRACE parameter 68
example on OC/FTP Client
  Moving an AIX file into a KSDS 268

example on OC/FTP Client *(continued)*
Obtain AIX file into VSAMSAM file 271
Obtain multiple files from AIX 262
Remote Directory Operations 259
Send KSDS file to AIX 248
Send VSAMSAM file to AIX 253
Transfer Multiple files to AIX 266
example on OC/FTP Server
Accessing Power 231
establish connection 226
get/put files to/from AIX 236
receive output from AIX 234
send listing to VSE 233
submit job to VSE 231
switching/displaying libraries 228
switching/displaying VSAM Data Sets 230
Using site 226
executable phase for FTP 102
executable phase for FTP server 73
executable phase for RSH 129
executable phases 115

F
F= operand 114
FAPPL member 94
FTPSXIT1 parameter 69
function of routers 36

G
GATEWAY parameter 113
gateway software 16
get a file from PC/DOS 245
GET queue file command 225
Get/put files for OC/FTP Server 236
GROUP parameter 89, 91, 127

H
H= operand 114
hardware requirements 9
Host Access Table (HAT) 185
host selection 293
HOSTS member 99, 138

I
IDBLK parameter 56
IDNUM parameter 56, 165
inbound pacing 100
installation for SAM 145
installation sublibrary for FTP 81
installation sublibrary for LPD 109
installation sublibrary for RSH 123
installation sublibrary for SAM 145
installation verification 63
instance 156
internet address 161
Internet layer 3
IP address 95, 184
IP address of test environment 35
ISC 83, 111, 124

K
keepalive option 160
keyboard mapping 292
KEYS parameter 65

L
LAN configuration 160
LAN major node 53
LANGUAGE parameter 92
LIBR access 71
library space 61
license manager key 196
license protection job for FTP 76, 104
license protection job for RSH 130
limitations 337
link station type parameter 176
LOCADDR parameter 67, 87, 112, 136, 167
LOGAPPL parameter 67
logmode table 66, 83, 111, 134, 201
LPD installation 109
LPDCONF member 112
LRECL parameter 94
LU configuration 167
LU definition 66, 87, 111, 136, 201
LU name parameter 67, 87, 112, 136, 167
LU type parameter 167

M
MACADDR parameter 53, 56, 161, 176
MAIL parameter 113
MAIL= operand 114
MAXDATA parameter 53, 56, 163
MAXIMUM parameter 90
midrange connection parameter 165
MODE parameter 94
MODENAME parameter 90
MODETAB parameter 66, 86, 111, 136
MPUT command 263

N
NETID parameter 174
NETMASK member 101
NETNAME parameter 89
network identifier 174
network interface layer 4
network tuning parameters 176
NETWORKS LU List member 95, 137
NETWORKS member 95, 137

number of sessions 173
NVPACE parameter 66, 86, 111
NZRI parameter 163

O

OC/FTP
  Accessing POWER 230
  APPCFTPC program 91
  batch operation 245
  CACHE member 99
  CD command 224
  CICS 88
  client batch job 105
  client commands 220, 239, 262
  client customization 82
  client installation 81
  client overview 20
  client startup 102
  client startup job 104
  CONFIG member 67
  DIR command 224
  displaying libraries 228
  displaying VSAM data sets 230
  DYNALL member 92
  establishing connection 226
  executable phase 73, 102
  FAPPL member 94
  functions 335
  GET queue file command 225
  HOSTS member 99
  installation sublibrary 81
  installation verification 63
  librarian member 224
  library space 61
  license protection job 76, 104
  NETMASK member 101
  NETWORKS LU List member 95
  NETWORKS member 95
  online operation 262
  operation 219
  operation samples 226
  PACING member 100
  PASSWD member 71
  POWER command 225
  problems 236
  PUT queue file command 225
  Receiving output from VSE 233
  RESOLVER member 101
  security 73, 331
  sending listing to VSE 232
  server customization 64
  server installation 61
  server overview 18
  server startup 73, 219
  server startup job 76
  SITE command 221, 227
  SITE member 69
  Site parameter list 222
OC/FTP *(continued)*
  submit job to VSE 231
  transaction definition 90
  translation table OC/FTP Server 76
  VSAM 64, 82
  VSAM catalog space 62
  VTAM customization 66, 83
OC/RSH
  batch operation 284
  CICS 126
  client batch job 131
  client commands 282
  client customization 124
  client installation 123
  client overview 24
  client startup 129
  client startup job 130
  executable phase 129
  installation sublibrary 123
  license protection job 130
  online operation 282
  transaction definition 127
  VTAM customization 124
OC/SAM
  installation 145
  installation sublibrary 145
  overview 28
  sample programs 310
  total overview 30
OC/Socket Access Method 8
OCC 56
OCC Customization 212
OCC Operation Samples 217
occonfig program 157
OCS II
  customization 156
  gateway overview 153
  gateway software 16
  installation 154
  LAN configuration 160
  link station parameters 176
  listing for configuration 178
  misc system parameters 169
  network tuning parameters 176
  occonfig program 157
  ocsna command 189
  octermcap file 295, 302
  operation 189
  PU/LU configuration 164
  RISC/6000 DLC 174
  shutdown 191
  SNA DLC 163
  sna_status command 190
  Telnet server 171
  TELNET Server 3270 17
  telnet server commands 297
  Token-Ring adapter 50
  VM customization 44

OCS II *(continued)*
VSE customization 48
VTAM customization 45
OCS Print Server Installation and Customization 193
CICS Customization 202
config file 197
Customization 195
Customization in VSE/ESA 200
Customization of the Print Server 196
defaults file 196
Definition to OCS II Gateway 200
eps_start command 205
eps_status command 205
eps_stop command 205
installation 193
operation 205
overview 193
printing 207
starting up 205
stopping 205
stopping the license manager 206
ocsna command 189
OCSNA environment variable 189
octermcap file 295, 302
offline interface for FTP 84
online interface for FTP 84
online interface for RSH 124
OPEN command 84, 99
Open Connection for Channel (OCC)
hardware 11
Main Circuit Board 12
overview 8
SCSI Bus/Tag Transition Board 12
OpenConnection for Channel (OCC) 209
Configure OCC with OCC Configuration file 213
Configuring OCC Hardware 210
Customization 212
Customizing OCS II Gateway 212
Installing OCC Hardware 210
OCC Administration Software Installation 209
OCC customization 212
OCC installation 209
OCC operation 217
RISC/6000 Host Software Installation 209
SNA configuration 215
OpenConnection for Channel (OCC) Operation 217
OPNDSP parameter 94
outbound pacing 100

P

PACEIN parameter 68
PACEOUT parameter 68
PACING member 100
PACING parameter 56, 136
PALLOC parameter 94
Parameter Library 67
PARSESS= parameter 86
PASSWD member 71
path table 45, 48
PC Windows applications 38
port number for aix 186
PORT parameter 68
POWER 18, 69
POWER command 225
printing 207
PRODBLDL.PROC 110
PRODBUILD.PROC 73, 82, 124
PROFILE EXEC 44
PROGRAM parameter 91, 127
PROTOCOL parameter 89, 90
PS/2 workstations 41
PTF found 236
PU configuration 165
PU name parameter 54, 56, 165
PU type parameter 165
PUT queue file command 225
PUTYPE parameter 56

Q

QUEUE parameter 113
QUEUENAME operand 113
QUEUETYPE operand 113
QUOTE POWER command 225
QUOTE SITE command 69, 221, 227

R

Receive output from VSE 233
RECEIVESIZE parameter 90
RECORDFORMAT parameter 65
RECORDS parameter 65
RECORDSIZE parameter 65
remote IP address 99
remote link address parameter 176
remote SAP address parameter 176
Replace Duplicate Keys (RDK) 268
RESOLVER member 101
RETAIN parameter 94
RISC/6000 DLC 174
Route algorithm
default routing 36
direct routing 36
indirect routing 36
Route table of test environment 37
Route type
dynamic route 36
explicit route 36
implicit route 36
RTB (Remove Trailing Blanks) 256

S

SALLOC parameter 94
SAPADDR parameter 53, 56, 176

SCSI OCC Configuration 211
Security 331
  FTPSXIT1 69
  FTPSXIT1 parameter 69
  PASSWD member 73
Send listing to VSE 232
SENDSIZE parameter 90
session definition 89
SESSION parameter 90
SET ACBNAME command 139
SET KEYBOARD command 141
SET LUMAX command 140
SET POOL command 140
SET USERDATA command 140
SET USERMAX command 140
Shell script
  receiving output from VSE 233
  sending listing to VSE 232
  submit job to VSE 231
SITE command 221
SITE member 69
Site parameter list 222
SMTP (Simple Mail Transfer Protocol) 313
SNA DLC 163
SNA host configuration 215
SNA local major node 46
sna_status command 190
software requirements 10
SSCPNAME parameter 46
Starting the OCC Administration Daemon 217
startup for FTP server 73
startup job for FTP client 104
startup job for FTP server 76
startup job for LPD 117
startup job for RSH client 130
station address parameter 165
stopping 205
stopping the license manager 206
Submit job to VSE 231
Subnet Masks 101
switched major node 54
SYSCMDS file 142
SYSIN file 139
SYSOUT= operand 114
SYSTEM CONFIG file 44
system parameters 169

T

TCP/IP
  FTP client commands 220
  functional layers 3
  hardware 9
  LAN configuration 160
  LPD 8
  node considerations 184
  OC/FTP Client 6
  OC/FTP Server 7
  OC/Line Printer Daemon (LPD) 8
TCP/IP *(continued)*
  OC/RSH Client 7
  OC/SAM 8
  OC/Telnet 6
  overview 5
  program numbers 15
  software 10
  test environment 31
Telnet
  APPL statement 135
  client customization 137
  customization 134
  DFLTSYS file 141
  DFLTUSR file 141
  establishing a session 135
  functions 336
  host selection 293
  HOSTS member 138
  installation 133
  keyboard mapping 292
  NETWORKS LU List member 137
  NETWORKS member 137
  OC/TELNET Server Manager 17
  operation 287
  overview 26
  server configuration 171
  startup 287
  startup customization 143
  sublibrary 133
  SYSCMDS file 142
  SYSIN file 139
  terminal keyboard profiles 142
  terminal setup options 289
  translation table 291
  VTAM customization 134
Telnet Server consideration 153
term types negotiation 171
termcap file size 171
terminal definition 203
terminal keyboard profiles 142
terminal setup options 289
test environment 31
time interval 169
Token-Ring adapter 50
  configuration 177
  customization 52
  LAN major node 53
  switched major node 54
  VSE IPL procedure 52
  VSE user directory 52
  VTAM configuration 53
trace of packets 220
transaction definition 90, 127, 202
TRANSACTION parameter 91, 127
translation table 291
Translation table customize
  OC/FTP Client 105
  OC/FTP Server 76

transport layer 3
tuning parameters 176
TWASIZE parameter 91, 127

U

usage 336
User exits
  Access Control 79
  I/O Exit 79
USSTAB 137

V

VCTCA 41
VM/ESA
  customization 44
  SYSTEM CONFIG file 44
  VSE user directory 44
  VTAM customization 45
  VTAM user directory 44
VPACING parameter 56, 136
VSAM access 71
VSAM catalog space 62
VSAM definition for FTP 82
VSAMCAT parameter 94
VSE/ESA
  IPL procedure 52
  OCC 58
  POWER command 225
  PROFILE EXEC 44
  TCP overview 6
  Telnet overview 26
  Token-Ring adapter 52
  user directory 44, 52
  VTAM customization 48
VTAM
  APPL major node 66, 83, 111, 124, 135
  CDRM major node 46, 49
  CDRSC major node 49
  CONFIG member 67
  connection VSE 88
  CTCA major node 45, 48
  customization for FTP 83
  customization for LPD 110
  customization for OC/Print Server 200
  customization for RSH 124
  FAPPL member 94
  LAN major node 53
  local SNA major node 46
  logmode table 66, 83, 111, 134, 201
  LU definition 66, 87, 136, 201
  OCC 58
  path table 45, 48
  PROFILE EXEC 44
  session definition 89
  switched major node 54
  Telnet customization 134
  Telnet overview 26
VTAM *(continued)*
  Token-Ring adapter 53
  user directory 44
  USSTAB 137
  VSE customization 48

Printed in U.S.A.

SG24-4270-01

Artwork Definitions			
id	**File**	**Page**	**References**
ITLOGO	4270SU	i	i
ITLOGOS	4270SU	i	

Table Definitions			
id	**File**	**Page**	**References**
COL1	4270VARS	i	36, 36, 70, 70, 197, 199
COL1H	4270VARS	i	
COL31	4270VARS	i	
COL32	4270VARS	i	35
COL34	4270VARS	i	70, 196
COL3A0	4270VARS	i	36
COL3A1	4270VARS	i	36
ROUTAB1	4270VARS	i	37, 37, 37, 37
COL3A	4270VARS	i	
COL5	4270VARS	i	
TT1	4270VARS	i	222
TT2	4270VARS	i	222
T21	4270VARS	i	223
TT3	4270VARS	i	223
TT4	4270VARS	i	224
TT5	4270VARS	i	225
TT6	4270VARS	i	225
TT7	4270VARS	i	240
TT8	4270VARS	i	241
TT9	4270VARS	i	242

Figures				
id	**File**	**Page**	**References**	
OSITCP	4270CH01	4	1	4
OCSOV	4270CH02	6	2	5, 153, 153
OCCENV	4270CH02	11	3	10
FGATE	4270CH03	16	4	16
FTNS	4270CH03	18	5	17
FFTPS	4270CH03	19	6	19
FFTPC	4270CH03	21	7	21
LPD	4270CH03			

		23	8	23
FRSHC	4270CH03	25	9	25, 25, 281
FTNFS	4270CH03	27	10	27, 287
FSAM	4270CH03	29	11	
CH3OVER	4270CH03	30	12	
FONWD	4270CH04	34	13	31, 31, 32, 33, 35, 36, 96, 138, 160, 184, 245
LOGCONF	4270CH04	37	14	37
PCWIN	4270CH04	38	15	38
PCWIN1	4270CH04	38	16	38
PCWIN2	4270CH04	39	17	39
PCWIN3	4270CH04	40	18	39
FIVMVSE	4270CH05	43	19	42
FVMRDEV	4270CH05	44	20	
FMPATH	4270CH05	45	21	
FMCTCA	4270CH05	45	22	45
FMCDRM	4270CH05	46	23	46
FM3174	4270CH05	47	24	46, 66, 87, 111, 136, 165, 201, 311
FSPATH	4270CH05	48	25	48
FSCTCA	4270CH05	49	26	48
FSCDRM	4270CH05	49	27	49
FSCDRS	4270CH05	50	28	49
FIVSE	4270CH05	51	29	50
FSTR	4270CH05	53	30	53, 176, 176
FSSW	4270CH05	55	31	54, 161, 163, 165
OCCVSE	4270CH05	57	32	56
OCPU	4270CH05	59	33	58
FLIB	4270CH06	62	34	61
FSCANT	4270CH06	63	35	63, 82, 110, 123, 133
FRETR	4270CH06	64	36	63, 82, 110, 123, 134
FMVSAM	4270CH06	65	37	64, 82
FML0	4270CH06	66	38	

				66
FSAPPL	4270CH06	66	39	
FSLU	4270CH06	67	40	66
PARM	4270CH06	67	41	
FSCONF	4270CH06	68	42	67
FSSITE	4270CH06	70	43	69
FSPASS	4270CH06	72	44	71, 226
FTPSBIL	4270CH06	74	45	73
FTPSSP	4270CH06	76	46	76
FTPS	4270CH06	76	47	76, 219
FTPS1	4270CH06	78	48	77
FML62	4270CH07	83	49	83
FCVTAM	4270CH07	85	50	83, 88, 239
FCAPPL	4270CH07	86	51	85, 89
FCLU	4270CH07	87	52	87, 111
FCDYNAL	4270CH07	93	53	93, 239
FCFAPP	4270CH07	95	54	94
FIPADDR	4270CH07	96	55	96
FCNW	4270CH07	97	56	96
FCNWL	4270CH07	97	57	97
FCNWL1	4270CH07	98	58	98
FCNWL2	4270CH07	98	59	98
FCHOST	4270CH07	99	60	99
FCCACHE	4270CH07	100	61	99
FCPAC	4270CH07	100	62	100
SUBMASK	4270CH07	101	63	101
RESOLV	4270CH07	102	64	101
FTPCBIL	4270CH07	103	65	102, 105, 105
FTPCSP	4270CH07	104	66	104
FTPC	4270CH07	104	67	104, 262, 282
FTPCBAT	4270CH07			

		105	68	105
XLATE2	4270CH07	106	69	105
XLATE21	4270CH07	107	70	105
LPAPPL	4270CH08	111	71	
LDLU	4270CH08	112	72	111
LPDCONF	4270CH08	113	73	112, 114
LPDBIL	4270CH08	116	74	115, 118, 118, 119, 120
LPDSTJO	4270CH08	117	75	117, 273
XLATE3	4270CH08	119	76	118
BANNER	4270CH08	120	77	120
LPDSMIT	4270CH08	121	78	121
RCVTAM	4270CH09	126	79	124, 281
RSHCHE	4270CH09	129	80	128
RSHCRH	4270CH09	129	81	128
RSHCBIL	4270CH09	129	82	129
RSHCSP	4270CH09	130	83	130
STRTRSH	4270CH09	130	84	130
RSHCBAT	4270CH09	131	85	131
FMLTN	4270CH10	134	86	134
FTAPPL	4270CH10	135	87	135
FTLU	4270CH10	136	88	136
TNNW	4270CH10	138	89	138
TNNWL	4270CH10	138	90	138
TNHOST	4270CH10	139	91	138, 295
TNSYSIN	4270CH10	139	92	139
TNUSER	4270CH10	140	93	140
TNTSYS	4270CH10	141	94	141
TNTUSR	4270CH10	141	95	141
TNCMDS	4270CH10	142	96	142
TNJOB	4270CH10	143	97	

SAMSCAN	4270CH11	146	98	143, 289
SAMINP1	4270CH11	146	99	145
SAMINP2	4270CH11	147	100	
SAMINP3	4270CH11	147	101	
SAMINP4	4270CH11	148	102	
SAMINS	4270CH11	148	103	
SAMVER1	4270CH11	149	104	146, 148
SAMVER2	4270CH11	149	105	
SAMVER3	4270CH11	150	106	
SAMTAB	4270CH11	151	107	
SAMVER4	4270CH11	152	108	
TII14	4270CH12	154	109	151
OCSGIN1	4270CH12	155	110	154
INST	4270CH12	157	111	154
OCSGCUS	4270CH12	158	112	156, 183
OCSGCU2	4270CH12	158	113	157
OCSGF11	4270CH12	159	114	157, 212
OCSGC11	4270CH12	160	115	
OCSGC12	4270CH12	161	116	
OCSGC14	4270CH12	162	117	160
OCSGC13	4270CH12	162	118	161
OCSGC21	4270CH12	163	119	160
OCSGC31	4270CH12	164	120	163
OCSGC32	4270CH12	165	121	
OCSGC33	4270CH12	166	122	
OCSGT33	4270CH12	167	123	166
OCSGC41	4270CH12	168	124	166, 200
OCSGC42	4270CH12	169	125	168
OCSGT55	4270CH12	170	126	
OCSGC45	4270CH12	171	127	
OCSGT90	4270CH12	172	128	295, 295, 296
OCSGT91	4270CH12	173	129	
OCSGT93	4270CH12	173	130	
OCSGC51	4270CH12	174	131	
OCSGC52	4270CH12	175	132	

OCSGT52	4270CH12	175	133	176
OCSGC54	4270CH12	177	134	176
OCSGC53	4270CH12	178	135	176, 176
OCSGC61	4270CH12	179	136	56
OCSGO61	4270CH12	181	137	
OCSGC71	4270CH12	183	138	183
OCSGC81	4270CH12	184	139	
OCSC811	4270CH12	185	140	185
OCSC812	4270CH12	186	141	185
OCSC813	4270CH12	187	142	187
SMP25	4270CH13	189	143	189, 189
SMP251	4270CH13	190	144	189
SMP24	4270CH13	191	145	190
SMP26	4270CH13	191	146	191
EPSFIG1	4270CH14	196	147	196
EPSFIG2	4270CH14	198	148	198
EPSFIG3	4270CH14	200	149	200, 205
LOGPS	4270CH14	201	150	201, 201
LUPS	4270CH14	201	151	201
EPSFG10	4270CH15	205	152	205, 206
EPSFG11	4270CH15	206	153	206
EPSFG12	4270CH15	207	154	206
EPSFG13	4270CH15	207	155	207, 207
EPSFG14	4270CH15	207	156	207
OCC14	4270CH16	210	157	210
CONFLU	4270CH16	213	158	
OCC15	4270CH16	214	159	214
OCC16	4270CH16	215	160	214
OCC17	4270CH16	216	161	216
OCCST	4270CH17	217	162	217

SCAST	4270CH17	218	163	217
OCC26	4270CH17	218	164	218
SMP10	4270CH18	219	165	219
SMP101	4270CH18	220	166	220
SMP4	4270CH18	226	167	226
SMP2	4270CH18	227	168	227
SMP22	4270CH18	228	169	
SMP3	4270CH18	229	170	228
SMP1	4270CH18	230	171	230
SMP6	4270CH18	231	172	231
SMP62	4270CH18	232	173	231
SMP63	4270CH18	232	174	231
SMP64	4270CH18	233	175	233
SMP65	4270CH18	233	176	233, 233
SMP66	4270CH18	235	177	234
SMP23	4270CH18	236	178	236
TCPFB1	4270CH19	246	179	245
TCPFB2	4270CH19	246	180	245
TCPSK01	4270CH19	249	181	248
TCPSK02	4270CH19	250	182	248
TCPSK04	4270CH19	251	183	248
TCPSK05	4270CH19	252	184	248
TCPSK11	4270CH19	254	185	253
TCPSK12	4270CH19	254	186	253
TCPSK13	4270CH19	255	187	253
TCPSK14	4270CH19	256	188	253
TCPSK15	4270CH19	257	189	253
TCPFB3	4270CH19	259	190	259
TCPFB4	4270CH19	260	191	259
SMP11	4270CH19			

		262	192	239, 262
SMP12	4270CH19	264	193	262
SMP13	4270CH19	265	194	262
SMP16	4270CH19	267	195	267
SMP161	4270CH19	267	196	267
SMP162	4270CH19	268	197	266, 267
SMP18	4270CH19	269	198	268
SMP181	4270CH19	270	199	268
SMP19	4270CH19	271	200	271
SMP191	4270CH19	272	201	271
LPDS01	4270CH20	273	202	273
LPDS02	4270CH20	273	203	273
LPDS03	4270CH20	274	204	274
LPDS04	4270CH20	275	205	275
LPDS44	4270CH20	275	206	275
LPDS05	4270CH20	277	207	278
LPDS55	4270CH20	279	208	
LPDS06	4270CH20	279	209	
RSHST	4270CH21	282	210	282
SMP7	4270CH21	283	211	282
SMP8	4270CH21	283	212	282
TCPRB1	4270CH21	284	213	284
TCPRB2	4270CH21	284	214	284
SMP20	4270CH22	287	215	287, 287
SMP15	4270CH22	288	216	288, 288
TERMSET	4270CH22	289	217	288, 292, 294
SMP21	4270CH22	291	218	288
SMP17	4270CH22	292	219	288, 288
HOSTSEL	4270CH22	293	220	288, 288
AIXWEL	4270CH22	294	221	294

SMP29	4270CH22	294	222	
TII0	4270CH23	295	223	
TII1	4270CH23	296	224	296
TI2	4270CH23	297	225	297
TI3	4270CH23	299	226	297
TI4	4270CH23	299	227	297
HATTAB	4270CH23	299	228	298
TI6	4270CH23	300	229	298
TI7	4270CH23	300	230	298
TI8	4270CH23	300	231	298
TI9	4270CH23	301	232	298
TIA	4270CH23	301	233	298
TIB	4270CH23	302	234	298
TID	4270CH23	303	235	298
TIC	4270CH23	303	236	302
SAMUDP	4270CH24	307	237	306
SAMTCP	4270CH24	309	238	308
SAMAPPL	4270CH24	310	239	310
SAMLU	4270CH24	311	240	
SAMUDSR	4270CH24	312	241	
SAMUDCR	4270CH24	313	242	
SAMTCSR	4270CH24	314	243	
SAMTCCR	4270CH24	314	244	
SAMTCSM	4270CH24	315	245	
SAMML1	4270CH24	316	246	
SAMML2	4270CH24	317	247	316, 318, 319
SAMML3	4270CH24	318	248	
SAMM4	4270CH24	318	249	
SAMML5	4270CH24	320	250	
SAMML6	4270CH24	321	251	321
SAMML7	4270CH24	321	252	321
FSGATE	4270CH25	326	255	
FSFTPS	4270CH25	327	256	
FSFTPC1	4270CH25	328	257	

FSFTPC2	4270CH25	329	258
FSTN	4270CH25	330	259

Headings

id	File	Page	References
P1	4270CH01	1	Part 1, Introduction
CH1INT	4270CH01	3	Chapter 1, Introduction to TCP/IP xvii
IVSETCP	4270CH02	5	Chapter 2, VSE/ESA and TCP/IP xvii
CH1OCS	4270CH02	5	2.1, OCS TCP/IP for VSE
CH2TELN	4270CH02	6	2.1.1, OC/Telnet
CH2FTPC	4270CH02	6	2.1.2, OC/FTP Client
CH2FTPS	4270CH02	7	2.1.3, OC/FTP Server
CH2RSH	4270CH02	7	2.1.4, OC/RSH Client
CH2SAM	4270CH02	8	2.1.5, OC/SAM
CH2LPD	4270CH02	8	2.1.6, OC/Line Printer Daemon (LPD)
CH2GATE	4270CH02	8	2.1.7, OCS II Gateway
CH2PS	4270CH02	8	2.1.8, OCS Print Server for AIX
CH2OCC	4270CH02	8	2.2, OpenConnection for Channel (OCC) 209, 217
CH1HSW	4270CH02	9	2.3, Hardware and Software Requirements for OCS TCP/IP
CH1OCSH	4270CH02	9	2.3.1, Hardware Requirements
CH1OCSS	4270CH02	10	2.3.2, Software Requirements
CH2HW	4270CH02	11	2.3.3, OCC Hardware
CH2MCB	4270CH02	12	2.3.3.1, Main Circuit Board
CH2BTT	4270CH02	12	2.3.3.2, SCSI Bus/Tag Transition Board
P2	4270CH03	13	Part 2, OCS TCP/IP for VSE Installation and Implementation
OCOV	4270CH03	15	Chapter 3, OCS Software Functional Overview xvii, 153
OCGOV	4270CH03	16	3.1, OCS II Gateway Functional Overview 153, 335
TNS3270	4270CH03	17	TELNET Server 3270
FTPSOV	4270CH03	18	3.2, OC/FTP Server Functional Overview
FTPCOV	4270CH03	20	3.3, OC/FTP Client Functional Overview
LPDOV	4270CH03	22	3.4, OC/Line Printer Daemon Functional Overview
RSHOV	4270CH03	24	3.5, OC/RSH Client Functional Overview
TNFSOV	4270CH03	26	3.6, OC/TELNET FS Functional Overview 287, 287
SAMCOV	4270CH03	28	3.7, OC/SAM Functional Overview
TCPTOV	4270CH03	30	3.8, OCS Software Functional Overview
OLAB	4270CH04	31	Chapter 4, Sample Environment xvii
OHW	4270CH04	31	4.1, Hardware
OSW	4270CH04	32	4.2, Software 331
ONWC	4270CH04		

		33	4.3, Network Configuration	
OGATT	4270CH04	35	4.3.1, OCS II Gateway Attachment Configurations	
IPADDR	4270CH04	35	4.3.2, IP Address and Name Assignments	
FUNCT1	4270CH04	36	4.4, Router Functions	
CLIENTS	4270CH04	38	4.5, PC/DOS Windows Client in our Test Environment	
OCIGATE	4270CH05	41	Chapter 5, Defining the OCS II Gateway to the Host	xvii, 61, 81, 109, 123, 133, 145, 199
OCIVMVS	4270CH05	41	5.1, VM/VSE with OCS II Gateway Attached via the IBM 3174	35
LOCSNAM	4270CH05	46	5.1.2.4, VTAM Local SNA Major Node	311, 311
OCIVSE	4270CH05	50	5.2, VSE/ESA with OCS II Gateway Attached via the Token-Ring Adapter	35
SVTAM	4270CH05	53	5.2.3, Customization in VSE VTAM	46
OCCVS01	4270CH05	56	5.3, VSE/ESA with OSCII Gateway Attached via the OCC	35, 216
OCCVTAM	4270CH05	58	5.3.3, Customization in VSE VTAM	
OCIFTPS	4270CH06	61	Chapter 6, OC/FTP Server Installation and Customization	xvii
IFTPS	4270CH06	61	6.1, OC/FTP Server Installation	81, 81, 81, 109, 123, 133, 133, 145
CFTPS	4270CH06	64	6.2, OC/FTP Server Customization	54
VSAMC	4270CH06	64	6.2.1, VSAM Customization	82
VTAMFTS	4270CH06	66	6.2.2, VTAM Customization	311
FTPSFC	4270CH06	67	6.2.3, FTP Server Customization	331
HCONFIG	4270CH06	67	6.2.3.1, CONFIG Member	73, 79, 100
HSITE	4270CH06	69	6.2.3.2, SITE Member	61, 62, 65
CUSTXLT	4270CH06	77	6.2.5, Customizing the Translation Tables	105, 118
EXITPGM	4270CH06	79	6.2.6, User Exits	221
OCIFTPC	4270CH07	81	Chapter 7, OC/FTP Client Installation and Customization	xvii, 123, 124, 124, 127, 128
IFTPC	4270CH07	81	7.1, OC/FTP Client Installation	
CFTPC	4270CH07	82	7.2, OC/FTP Client Customization	22, 54
VSAMCC	4270CH07	82	7.2.1, VSAM Customization	
FTPCVTM	4270CH07	83	7.2.2, VTAM Customization	111, 114
CICSC	4270CH07	88	7.2.3, CICS Customization	86
FTPCFC	4270CH07	92	7.2.4, FTP Client Customization	36, 283
FTPCDYN	4270CH07	92	7.2.4.1, DYNALL Member	81, 81, 82
FTPCAPP	4270CH07	94	7.2.4.2, FAPPL Member	86
FTPCNW	4270CH07	95	7.2.4.3, NETWORKS Member	

				138
FTPCL	4270CH07	97	7.2.4.4, NETWORKS LU List Member for IPFNET NETWORKS Member	87, 87
FTPCL1	4270CH07	97	7.2.4.5, NETWORKS LU List Member for SUNNET NETWORKS Member	87
FTPCL3	4270CH07	98	7.2.4.6, NETWORKS LU List Member for AIXNET NETWORKS Member	
FTPCHOS	4270CH07	99	7.2.4.7, HOSTS Member	
FTPCCAC	4270CH07	99	7.2.4.8, CACHE Member	
FTPCPAC	4270CH07	100	7.2.4.9, PACING Member	
MASK	4270CH07	101	7.2.4.10, NETMASK Member	95
RESOLV	4270CH07	101	7.2.4.11, RESOLVER Member	99, 226
STRTJOB	4270CH07	104	7.2.5.3, Create the FTP Client Startup Online Job	101
OCILPD	4270CH08	109	Chapter 8, OC/Line Printer Daemon Installation and Customization	xvii
ILPD	4270CH08	109	8.1, OC/Line Printer Daemon Installation	
CLPD	4270CH08	110	8.2, OC/Line Printer Daemon Customization	
LPDFC	4270CH08	112	8.2.2, LPD Customization	121
LPDCONF	4270CH08	112	8.2.2.1, LPDCONF Member	111, 274
LPDADDL	4270CH08	115	8.2.2.2, Additional Configuration Members	
LPSTJO	4270CH08	117	8.2.3.2, Create the LPD Startup Job	115
CBANNER	4270CH08	119	8.2.5, Customizing the OC/Line Printer Daemon Print Banner	114
OCIRSHC	4270CH09	123	Chapter 9, OC/RSH Client Installation and Customization	xviii
IRSHC	4270CH09	123	9.1, OC/RSH Client Installation	
CRSHC	4270CH09	124	9.2, OC/RSH Client Customization	
RSHCICS	4270CH09	126	9.2.2, CICS Customization	
RSHCRC	4270CH09	128	9.2.3, RSH Client Customization	
OCITN	4270CH10	133	Chapter 10, OC/TELNET FS Installation and Customization	xviii, 287
ITN	4270CH10	133	10.1, OC/TELNET FS Installation	
CTN	4270CH10	134	10.2, OC/TELNET FS Customization	26, 54
TNCC	4270CH10	137	10.2.2, TELNET Client Customization	288
TNHOST	4270CH10	138	10.2.2.3, HOSTS File	293
SYSIN	4270CH10	139	10.2.2.4, SYSIN File	136
OCISAM	4270CH11	145	Chapter 11, OC/SAM Installation and Customization	xviii
ISAM	4270CH11	145	11.1, OC/SAM Installation	
SAMVER	4270CH11	149	11.1.3, Installation Verification	
ISAMGST	4270CH11	150	11.2.1, GET SERVICE BY NAME Table Customization	324

OCIGTW	4270CH12	153	Chapter 12, OCS II Gateway Installation and Customization	xviii, 17
OCIGTWO	4270CH12	153	12.1, OCS II Gateway Overview	
OCSGPI	4270CH12	153	12.2, Preinstallation Tasks	
OCSGI	4270CH12	154	12.3, Installation	
OCSGC	4270CH12	156	12.4, Customization	212
OCSGC1	4270CH12	160	12.4.1, Local Area Network (TCP/IP) Configuration	
OCSGC2	4270CH12	163	12.4.2, SNA Level Data Link Configuration	200
OCSGC3	4270CH12	164	12.4.3, Physical/Logical Unit Configuration	
OCSGC4	4270CH12	168	12.4.4, System Wide Configuration Options	295
OCSGC5	4270CH12	174	12.4.5, OCS II RISC/6000 Data Link Configuration	
OCSGC6	4270CH12	178	12.4.6, Generate a Configuration Listing	56
OCSGC7	4270CH12	182	12.4.7, Load Configuration File Set	
OCSGC8	4270CH12	183	12.4.8, Save Configuration File Set	297
OCISPC	4270CH12	184	12.5, Special Considerations	
OCISPCN	4270CH12	184	12.5.1, TCP/IP Node Considerations	
OCSPCH	4270CH12	185	12.5.2, Host Access Table (HAT)	
PORTNO	4270CH12	186	12.6, Set up Alternative Ports for AIX TELNET and FTP Servers	68, 171, 184, 226, 295
TCPTOGW	4270CH13	189	Chapter 13, OCS II Gateway Operation	xviii, 295
TCPGWS	4270CH13	189	13.1, Operation Samples (start,stop,status....)	156
OCIPRTS	4270CH14	193	Chapter 14, OCS Print Server Installation and Customization	xviii
OCIPRTO	4270CH14	193	14.1, OCS Print Server Overview	
OCSPRTI	4270CH14	193	14.2, OCS Print Server Installation	
OCSPRTC	4270CH14	195	14.3, OCS Print Server Customization	
OPCUST	4270CH14	196	14.3.1, Customization of the OCS Print Server	195
CUSTDEF	4270CH14	196	14.3.1.2, Customizing the Defaults File	199
CUSTCFG	4270CH14	197	14.3.1.3, Customizing the Config File	
OPGATE	4270CH14	200	14.3.2, Customization of the OCS II Gateway for OCS Print Server	195, 201
OPVSE	4270CH14	200	14.3.3, Customization in VSE/ESA	195
OPVTAM	4270CH14	200	14.3.4, VTAM Customization	200
PSVTAM	4270CH14	201	14.3.4.1, Logmode Table OCSBIND	
PSLU	4270CH14	201	14.3.4.2, VTAM LU Definition	
PSCICS	4270CH14	202	14.3.5, CICS Customization	
TCPTOPS	4270CH15	205	Chapter 15, OCS Print Server Operation	xviii
TCPPSS	4270CH15	205	15.1, Operation Samples (start,stop,status....)	
TCPPSV	4270CH15			

		207	15.2, Printing Files from CICS Report Controller	
OCCINST	4270CH16	209	Chapter 16, OpenConnection for Channel Installation and Customization	xviii
OCCSUM	4270CH16	209	16.1, Installation Summary	
OCCGT	4270CH16	209	16.1.1, RISC/6000 Host Software Installation	
OCCSW	4270CH16	209	16.1.1.1, OCC Administration Software Installation	
OCCHW	4270CH16	210	16.1.2, Installing OCC Hardware	
OCCCO	4270CH16	210	16.1.3, Configuring OCC Hardware	58
OCCCI	4270CH16	212	16.1.4, OCC Customization	
OCCGC	4270CH16	212	16.1.4.1, Customize OCS II Gateway	
OCCSN	4270CH16	215	16.1.5, Configure the SNA Host for OCS II Gateway	
OCCST	4270CH17	217	Chapter 17, OpenConnection for Channel Operation	xviii
OCCOP	4270CH17	217	17.1, OCC Operation Samples	
TCPTOFS	4270CH18	219	Chapter 18, OC/FTP Server Operation and Examples	xviii, 18, 63, 65, 76
TCPFSC	4270CH18	219	18.1, Operation	
TCPFSCH	4270CH18	219	18.1.1, VSE Host (FTP Server)	
TCPFSCC	4270CH18	220	18.1.2, TCP/IP FTP Clients	69
TCPFSS	4270CH18	226	18.2, Operation Samples	221
TCPFSLS	4270CH18	226	18.2.1, Establish Connection to the OC/FTP Server	
TCPFSSS	4270CH18	227	18.2.2, Using SITE	
TCPFSSL	4270CH18	228	18.2.3, Switching/Displaying Libraries (Librarian)	
TCPFSSV	4270CH18	230	18.2.4, Switching/Displaying VSAM Data Sets	
TCPFSPO	4270CH18	231	18.2.5, Accessing POWER	
TCPFSSJ	4270CH18	231	18.2.6, Submitting a Job to VSE/ESA	
TCPFSSP	4270CH18	233	18.2.7, Sending Print Lists to VSE/ESA	
TCPFSSR	4270CH18	234	18.2.8, Receiving Print Lists from VSE/ESA	
FTPSPRB	4270CH18	236	18.2.9, Transferring Files to/from VSE/ESA	
TCPTOFC	4270CH19	239	Chapter 19, OC/FTP Client Operation and Examples	xviii, 20, 82, 82, 92, 104
FCCS	4270CH19	239	19.1, FTP Client Command Summary	
TCPFCS	4270CH19	245	19.2, Batch Operation Samples	
TCPFCSB	4270CH19	245	19.2.1, Get a File from PC/DOS to VSE Library	101
TCPSKSD	4270CH19	248	19.2.2, Send a VSE/VSAM KSDS File to AIX	
TCPSVSM	4270CH19	253	19.2.3, Sending a VSE/VSAM SAM File to AIX	
TCPFCB1	4270CH19	259	19.2.4, Remote Directory Operations	
TCPFCC	4270CH19	262	19.3, Online Operation Samples	
TCPFCSO	4270CH19	262	19.3.1, Obtaining Multiple Files from AIX	
TCPFCO1	4270CH19	266	19.3.2, Transferring Multiple Files to AIX	
TCPFRK1	4270CH19	268	19.3.3, Moving an AIX Data File into a VSAM KSDS	
TCPFRK2	4270CH19	271	19.3.4, Transferring an AIX Data File into a VSAM SAM File	
TCPTOLP	4270CH20	273	Chapter 20, OC/Line Printer Daemon Operation and Examples	

				xviii, 110, 118
TCPTORS	4270CH21	281	Chapter 21, OC/RSH Client Operation and Examples	xviii, 24, 124, 130
TCPRCC	4270CH21	282	21.1, OC/RSH Client Online Operation Examples	
TCPRCS	4270CH21	284	21.2, Batch Operation Samples	
TCPRCSB	4270CH21	284	21.2.1, RSH Batch Example	
TCPTOTN	4270CH22	287	Chapter 22, OC/TELNET FS Operation and Examples	xviii, 134
TCPTNSS	4270CH22	287	22.1, Operation Samples	
TCPTNSP	4270CH22	288	22.2, Working with Telnet Panels	
TCPTOTS	4270CH23	295	Chapter 23, OCS II Telnet Server Operation and Examples	xix
TCPTSG	4270CH23	295	23.1, General Considerations	
TCPTSS	4270CH23	295	23.2, Operation Examples	
TCPTSSW	4270CH23	296	23.2.1, Telnet Server Welcome Panel	
TCPTSSC	4270CH23	297	23.2.2, Telnet Server Commands Panel	
TCPTSCL	4270CH23	299	23.2.3, Display OCS II Telnet Server Clients	
TCPTSCO	4270CH23	299	23.2.4, Display OCS II Telnet Server Configuration	
TCPTSHT	4270CH23	299	23.2.5, Display OCS II Telnet Server HAT Table	
TCPTSKE	4270CH23	300	23.2.6, Display OCS II Telnet Server Keycodes	
TCPTSAD	4270CH23	300	23.2.7, Display OCS II Telnet Server ′trace′ Facility	
TCPTSSE	4270CH23	300	23.2.8, Display OCS II Telnet Server Services	
TCPTSSH	4270CH23	301	23.2.9, Display OCS II Telnet Server Terminal Types	
TCPTSST	4270CH23	301	23.2.10, Display OCS II Telnet Server Status	
TCPTSTE	4270CH23	302	23.2.11, Display Terminal Characteristics	
TCPTSH	4270CH23	302	23.2.12, OCS II Telnet Server Help Information	
TCPTOSA	4270CH24	305	Chapter 24, OC/SAM Programming	xix
SAMTUT	4270CH24	305	24.1, TCP/IP Application Programming Tutorial	
SAMOVER	4270CH24	305	24.1.1, Network Programming Overview	
SAMPRG1	4270CH24	305	24.1.2, Socket Programming	
SAMAD	4270CH24	306	24.1.3, Socket Addresses	
TCPSAM	4270CH24	310	24.2, Running the Sample Programs	322
SAMMTAC	4270CH24	321	24.3.5, Message Transfer Agent Program Implementation and Setup	
OCISUM	4270CH25	325	Chapter 25, Implementation Summary	xix
P4	4270CH26	333	Part 3, Summary	
OCSSUM	4270CH26	335	Chapter 26, OpenConnect Systems Products Functions and Limitations Summary	xix
OCSADD	4270CH26	336	26.2, Additional OpenConnect Systems Products Summary	
P5	4270AX01	339	Part 4, Job Streams and Definition Examples	
APX1	4270AX01	341	Appendix A, Job Streams and Definition Examples for OCS Products	
INST	4270AX01	341	A.1, Product Installation Job Streams	
VTSTR	4270AX01	341	A.2, VSE VTAM Start Options List	46, 174
ASSW	4270AX01			

		342	A.3, VSE TCPSW.B Switched Major Node 54, 66, 87, 111, 136, 167, 176, 201
TCPAPPL	4270AX01		
		344	A.4, VTAM APPL Major Node for OCS Products 66, 85, 135
USSTAB	4270AX01		
		345	A.5, VTAM USSTAB 137
OCSBIND	4270AX01		
		351	A.6, OCSBIND MODETAB for OCS Products 66, 83, 111, 134, 201
NOTICES	SG244270 SCRIPT		
		353	Appendix B, Special Notices ii
BIBL	4270BIBL		
		357	Appendix C, Related Publications
ORDER	REDB$ORD		
		359	How To Get ITSO Redbooks 357

Index Entries

id	File	Page	References
TCPIND	4270CH01		
		3	(1) TCP/IP 3, 5, 6, 6, 7, 7, 8, 8, 8, 9, 10, 15, 31, 160, 184, 220
FTPIND	4270CH01		
		3	(1) OC/FTP 18, 20, 61, 61, 62, 63, 64, 64, 66, 67, 69, 71, 73, 73, 73, 76, 76, 76, 81, 81, 82, 82, 83, 88, 90, 91, 92, 94, 95, 95, 99, 99, 100, 101, 101, 102, 102, 104, 104, 105, 219, 219, 220, 221, 222, 224, 224, 224, 225, 225, 225, 226, 226, 227, 228, 230, 230, 231, 232, 233, 236, 239, 245, 262, 262, 331, 335
RSHIND	4270CH01		
		3	(1) OC/RSH 24, 123, 123, 124, 124, 126, 127, 129, 129, 130, 130, 131, 282, 282, 284
SAMIND	4270CH01		
		3	(1) OC/SAM 28, 30, 145, 145, 310
OCSIND	4270CH01		
		3	(1) OCS II 16, 17, 44, 45, 48, 50, 153, 154, 156, 157, 160, 163, 164, 169, 171, 174, 176, 176, 178, 189, 189, 190, 191, 295, 297, 302
TELIND	4270CH01		
		3	(1) Telnet 17, 26, 133, 133, 134, 134, 135, 135, 137, 137, 137, 138, 139, 141, 141, 142, 142, 143, 171, 287, 287, 289, 291, 292, 293, 336
VTAMIND	4270CH01		
		3	(1) VTAM 26, 44, 44, 45, 45, 46, 46, 48, 48, 48, 49, 49, 53, 53, 54, 58, 66, 66, 66, 67, 83, 83, 83, 87, 88, 89, 94, 110, 111, 111, 124, 124, 134, 134, 135, 136, 137, 200, 201, 201
CICSIND	4270CH01		
		3	(1) CICS 88, 88, 89, 90, 126, 127, 202, 203
VMIND	4270CH01		
		3	(1) VM/ESA 44, 44, 44, 44, 45
VSEIND	4270CH01		
		3	(1) VSE/ESA 6, 26, 44, 44, 48, 52, 52, 52, 58, 225
OCC	4270CH02		
		8	(1) Open Connection for Channel (OCC) 8, 11, 12, 12
ROUTYPE	4270CH04		
		36	(1) Route type
ROUTALG	4270CH04		
		36	(1) Route algorithm
TOKIND	4270CH05		
		50	(1) Token-Ring adapter 52, 52, 52, 53, 53, 54, 177
OCCVS01	4270CH05		
		56	(1) OCC
FTPSSEC	4270CH06		
		69	(1) Security 73
UEXIT	4270CH06		
		69	(1) Security (2) FTPSXIT1

OCSERR	4270CH06	69	(1) Error 71, 101, 236
TRANCUT	4270CH06	76	(1) Translation table customize 105
USREXIT	4270CH06	78	(1) User exits
PRTIND	4270CH14	193	(1) OCS Print Server Installation and Customization 193, 193, 195, 196, 196, 197, 200, 200, 202, 205, 205, 205, 205, 205, 205, 206, 207
OCCIN	4270CH16	209	(1) OpenConnection for Channel (OCC) 209, 209, 209, 210, 210, 212, 212, 212, 213, 215, 217
OCCST	4270CH17	217	(1) OpenConnection for Channel (OCC) Operation
EXAMS	4270CH18	226	(1) example on OC/FTP Server 226, 228, 230, 231, 231, 233, 234, 236
SHELLPG	4270CH18	231	(1) Shell script 232, 233
EXAMC	4270CH19	245	(1) example on OC/FTP Client 248, 253, 259, 262, 266, 268, 271

Tables

id	File	Page	References
OCST1	4270CH02	9	1 9
OCST2	4270CH02	10	2 10
TIPADDR	4270CH04	36	3 35, 96, 99, 138, 138
ROUTAB	4270CH04	37	4 37
TSITE	4270CH06	70	5 69
LPQUEUE	4270CH08	115	6
TDEF	4270CH14	197	7 196
TCONF	4270CH14	199	8 198
FTPSHC	4270CH18	220	9 220
FTPSCC	4270CH18	222	10 221
FTPSCC2	4270CH18	225	11 224
FTPCC1	4270CH19	239	12 239
FTPCJC	4270CH19	262	13 262, 262, 282
RSHCJC	4270CH21	282	14 282

Processing Options

Runtime values:

Document fileid	SG244270 SCRIPT
Document type	USERDOC
Document style	REDBOOK
Profile	EDFPRF30
Service Level	0029
SCRIPT/VS Release	4.0.0
Date	96.09.03
Time	04:55:33
Device	3820A
Number of Passes	4
Index	YES
SYSVAR D	YES
SYSVAR G	INLINE
SYSVAR S	OFFSET
SYSVAR X	YES

Formatting values used:

Annotation	NO
Cross reference listing	YES
Cross reference head prefix only	NO
Dialog	LABEL
Duplex	YES
DVCF conditions file	(none)
DVCF value 1	(none)
DVCF value 2	(none)
DVCF value 3	(none)
DVCF value 4	(none)
DVCF value 5	(none)
DVCF value 6	(none)
DVCF value 7	(none)
DVCF value 8	(none)
DVCF value 9	(none)
Explode	NO
Figure list on new page	YES
Figure/table number separation	YES
Folio-by-chapter	NO
Head 0 body text	Part
Head 1 body text	Chapter
Head 1 appendix text	Appendix
Hyphenation	NO
Justification	NO
Language	ENGL
Layout	OFF
Leader dots	YES
Master index	(none)
Partial TOC (maximum level)	4
Partial TOC (new page after)	INLINE
Print example id′s	NO
Print cross reference page numbers	YES
Process value	(none)
Punctuation move characters	.,
Read cross-reference file	(none)
Running heading/footing rule	NONE
Show index entries	NO
Table of Contents (maximum level)	3
Table list on new page	YES
Title page (draft) alignment	RIGHT
Write cross-reference file	(none)

Imbed Trace

Page 0	4270SU
Page 0	4270VARS
Page 0	REDB$BOE
Page i	REDB$ED1
Page i	4270EDNO
Page i	REDB$ED2
Page xvii	4270ABST
Page xvii	4270ORG
Page xix	4270ACKS
Page xix	REDB$COM
Page xx	4270MAIN
Page xx	4270CH01
Page 4	4270CH02
Page 12	4270CH03
Page 30	4270CH04
Page 40	4270CH05
Page 59	4270CH06
Page 79	4270CH07
Page 107	4270CH08
Page 121	4270CH09
Page 131	4270CH10
Page 143	4270CH11
Page 152	4270CH12
Page 187	4270CH13
Page 191	4270CH14
Page 204	4270CH15
Page 207	4270CH16
Page 216	4270CH17
Page 218	4270CH18
Page 237	4270CH19
Page 272	4270CH20
Page 279	4270CH21
Page 285	4270CH22
Page 294	4270CH23
Page 303	4270CH24
Page 324	4270CH25
Page 331	4270CH26
Page 338	4270AX01
Page 353	4270SPEC
Page 353	REDB$SPE
Page 354	4270TMKS
Page 355	4270BIBL
Page 358	REDB$ORD