

AS/400e



TCP/IP Configuration and Reference

Version 4

AS/400e



TCP/IP Configuration and Reference

Version 4

Note

Before using this information and the product it supports, be sure to read the information in "Notices" on page 595.

Fourth Edition (May 1999)

This edition replaces SC41-5420-02. This edition applies only to reduced instruction set computer (RISC) systems.

© **Copyright International Business Machines Corporation 1997, 1999. All rights reserved.**

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

Contents

About TCP/IP Configuration and Reference (SC41-5420)	xv
TCP/IP Topics in the Information Center	xv
Who should read this book	xv
AS/400 Operations Navigator	xvi
Installing Operations Navigator.	xvi
Prerequisite and related information	xvii
How to send your comments	xvii
Chapter 1. TCP/IP on AS/400	1
Linking Networks Together	1
Internetwork Communications	2
Internet Addresses	2
Accessing the Internet	3
IP Security	4
Classes of Networks	4
IP Subnets	5
Subnetworks and Subnet Masks	6
Broadcast Addresses	8
Domain Name System (DNS)	9
Domain and Host Name	9
Naming Conventions for Domain Names and Host Names	10
Routing	12
Introduction to TCP/IP Protocols on AS/400	12
Application Protocols	13
Application Protocol Standards.	13
OS/400 Network File System Support	15
Application Program Interfaces (APIs)	15
Transport Protocol	16
Transmission Control Protocol (TCP)	16
User Datagram Protocol (UDP)	17
TCP and UDP Ports	17
Point-to-Point TCP/IP	17
Internetwork Protocol	18
Internet Protocol	18
Internet Control Message Protocol	19
Internet Group Management Protocol	19
Address Resolution Protocol	19
AnyNet/400	19
Chapter 2. Configuring TCP/IP	21
What you need to know before you can configure TCP/IP.	21
Planning for TCP/IP Installation and Configuration	22
Gathering Information About your Network	22
Installing the TCP/IP Application Programs	23
TCP/IP Addressing	24
Using the TCP/IP Administration Menu	26
Using the Configure TCP/IP Menu	27
Configuring TCP/IP using the Command Line Interface.	29
Step 1—Configuring a Line Description	30
Step 2—Configuring a TCP/IP Interface	30
Step 3—Configuring TCP/IP Routes.	32
Step 4—Configuring TCP/IP attributes	36
Step 5—Configuring TCP/IP Remote System Information (X.25)	36

Step 6—Configuring TCP/IP Host Table Entries	38
Step 7—Configuring the Local Domain and Host Name	42
Step 8—Starting TCP/IP and TCP/IP Servers	43
Step 9—Verifying the TCP/IP Connection	46
Verifying Additional TCP/IP Connections	47
Step 10—Saving Your TCP/IP Configuration	50
TCP/IP Planning Checklists	51
Sample Network Drawing.	52
Chapter 3. TCP/IP: Operation, Management, and Advanced Topics	55
Network Status	55
Work with TCP/IP Network Status Menu	55
Work with TCP/IP Interface Status	56
Display TCP/IP Route Information	59
Work with TCP/IP Connection Status	60
Working with Configuration Status	63
Displaying TCP/IP Network Status Information	64
TCP/IP Host Tables	72
Managing TCP/IP Host Tables	73
Host File Formats	73
Tips for Merging Host Tables	74
Merging TCP/IP Host Tables	75
Managing the Host Table from a Central Site	75
Domain Name System (DNS) Server	76
IP Routing and Internet Control Message Protocol (ICMP) Redirecting	76
Dead Gateway Processing	78
Negative Advice from TCP or the Data Link Layer	78
How IP Responds to Negative Advice	78
Multihoming Function	79
Example: A Single Host on a Network over a Communications Line	79
Example: Multiple Hosts on the Same Network over the Same Communications Line	80
Example: Multiple Hosts on the Same Network over Multiple Communications Lines	80
Example: Multiple Hosts on Different Networks over the Same Communications Line	81
Example: Multiple Hosts on Different Networks over Multiple Communications Lines	81
Example: The Multihoming function	82
Type of Service (TOS)	83
Multiple Routes	84
TCP/IP Port Restriction	85
Configuring TCP/IP Port Restrictions	86
Related Tables and the Host Table	88
Using X.25 PVC instead of SVC	91
IP Multicasting.	91
Multicast Application Programming Information	91
Multicast Restrictions	92
Chapter 4. Configuring Point-to-Point TCP/IP (PPP and SLIP)	93
Networks and Point-to-Point Connections	93
PPP versus SLIP.	94
Requirements for AS/400 SLIP.	95
Point-to-Point Request for Comments (RFC)	95
Line Pools	95
Configuring Point-to-Point Network Connections	96

Configuring PPP Connection Profiles	96
Accessing Point-to-Point functions through Operations Navigator	96
Checking for existing PPP Connection Profiles	96
PPP Configuration Scenarios	97
Example: Configuring Windows 95/98 to an AS/400 using a PPP Connection	97
Example: Connecting to the Internet using an ISP	97
Example: Connecting two AS/400s using dial-on-demand	98
Example: AS/400 Office-to-Office Scenarios	100
Example: Remote LAN Access with Transparent Subnetting	104
Example: Remote LAN Access with Dynamic Routing (RIP)	107
Monitoring Activity	109
Point-to-Point Jobs	110
Connection Alternatives	111
Analog Phone Lines	112
Digital Data Service	112
DDS	112
Switched-56	113
ISDN	113
T1/E1	114
Fractional T1	115
Using an Asynchronous Modem or ISDN Terminal Adapter	115
PPP ISDN Support	115
Configuring SLIP Connection Profiles	115
Writing Connection Dialog Scripts	118
Connection Script Considerations for SLIP	118
Connection Script Considerations for PPP	124
NLS Considerations.	125
Using SLIP with an Asynchronous Line Description	126
Connection Dialog Scripts	127
Configuring AS/400 Point-to-Point for SLIP	127
Monitoring Point-to-Point Activity	134
PPP/SLIP over *PPP	156
Chapter 5. Telnet Client	159
5250 Full-Screen Mode Considerations	159
TN5250—Start TCP/IP Telnet Command	159
TN5250—Screen Size	159
3270 Full-Screen Mode Considerations	159
TN3270—Start TCP/IP Telnet Command	160
Using a Display Station during Telnet 3270 Full-Screen Mode	160
TN3270—Screen Size	161
TN3270—Cursor Select Key	162
TN3270—Messages	162
TN3270—Handling Null Characters	162
VTxxx Full-Screen Mode Considerations	163
Operational Differences	163
Keyboard Issues	164
Screen Issues	165
VTxxx—Screen Size	166
VTxxx—Character Attributes	166
VTxxx—Start TCP/IP Telnet Command.	166
Changing the VTxxx Keyboard Map	167
VTxxx—National Language Support.	179
VTxxx—Multinational Mode	180
VTxxx—National Mode	180
System Functions Available during a Telnet Client Session	182

Print	182
Chapter 6. Telnet Server	183
Setting Up the Telnet Server	183
Determining Which Emulation Is Negotiated	183
5250 Full-Screen Mode	183
Examples of 5250 Server to 5250 Full-Screen Telnet Client	183
3270 Full-Screen Mode	186
Setting up for 3270 Full-Screen Mode	187
Break Messages in 3270 Full-Screen Mode	195
Input-Inhibited Light	195
Defining Capabilities for 3270 Devices	195
VTxxx Full-Screen Mode	196
Setting up for VTxxx Full-Screen Mode	196
VTxxx Automatic Wrap.	207
System Request Processing for VTxxx Sessions	207
Error Conditions on 5250 Keyboard	207
Display Screens and VTxxx Support.	208
VT220 Control Characters	208
Some Practical Examples	209
ASCII Line Mode	210
Setting up for ASCII Line Mode	211
Telnet Printer Pass-Through Mode	216
Setting Up for Telnet Printer Pass-Through Mode	217
Telnet Printer Pass-Through Mode Server to Client Access Win95 Telnet Client	219
Ending a Telnet Server Session	220
Starting Cascaded Telnet or DSPT Sessions	220
Using System Request Options	220
Telnet Scenarios for Establishing Cascaded Sessions	220
System Request Processing—Scenarios	225
Using a Group Job—Scenario	227
Workstation Type Negotiations and Mappings	229
System API Enhancement	231
Dynamic Application Printing with TCP/IP	231
Exit Point Performance	233
Work Management	233
Chapter 7. File Transfer Protocol (FTP) Client	235
Functions Supported by FTP Client	235
Functions Not Supported by FTP Client	235
FTP Client and Server-Overview	235
Starting the FTP Client Session	237
Alternative Start Commands.	237
Connecting to Another Server without Ending the FTP Session.	240
Ending the FTP Client Session.	240
Transferring Files with File Transfer Protocol (FTP)	241
Naming Format Indicator for AS/400 Names.	241
File Naming for the Library File System (QSYS.LIB)	242
Names for Document Library Services (QDLS) Folders and Documents	242
Names for “root,” QOpenSys, QLANSrv and QFileSvr.400 File Systems	242
Localfile and Remotefile Parameters for FTP Client Subcommands	242
Default File Names for Client Transfer Subcommands	242
FTP Client Subcommands	244
FTP Examples.	244
FTP Considerations (for Both Client and Server)	258

FTP Client Considerations	266
FTP as Batch Job	269
Exit Points for FTP	277
Chapter 8. File Transfer Protocol (FTP) Server	279
FTP Server-What It Does and Does Not Support	279
Functions Supported by AS/400 FTP Server.	279
Functions Not Supported by FTP Server	280
Configuring FTP Servers	280
Starting FTP Servers	280
Available FTP Servers	281
Ending FTP Servers	281
Ending and Restarting FTP Server Jobs	282
FTP Server Subcommands	282
FTP Server Considerations	282
FTP Server Considerations for Non-AS/400 Clients	282
FTP Server NAMEFMT	283
Exit Points for FTP Server Security and Anonymous FTP	284
Chapter 9. Post Office Protocol (POP) Mail Server	285
How the POP Server Works.	285
The POP Server and Client Access-based Mail	287
How to Get the POP Server Up and Running	288
Setting Up Your System and Users	289
Adding POP Mail Users to the System Distribution Directory.	289
POP Mailboxes	292
Setting Up Standard POP Mail Clients	292
Setting Up Client Access-Based Mail Clients	293
Configuring the POP Server.	293
Configuring POP for Client Access-Based Mail Users	294
Removing POP Mail Users from the System.	295
Setting the Number of SNA Servers	295
Starting the POP Server	296
Ending the POP Server	296
Supported POP Verbs	296
How the POP Server Uses the Mail Server Framework.	297
Exchanging Mail with OfficeVision	297
Configuring Both POP and SMTP	297
Using *ANY Support with the POP Server	297
MIME Mail Sent To OfficeVision	298
Long Line Conversion	299
Data Area Values.	299
MIME Content Types	302
Supported Content Types of the POP Server	303
How the File Name is Derived	305
MIME Content Types	305
What Happens When You Send OfficeVision Mail to POP Clients	305
Setting Up MIME Headers to Differentiate between Recipients	307
Sending MIME (POP Server) Mail across a SNADS Network	308
How SNADS Tunneling Works	308
How to Configure System Distribution Directory Entries for SNADS Tunneling.	308
Address Types	310
AS/400 Address Book	311
The Address Book Cache	312
ASCII-EBCDIC Conversion and National Language Support	313

EBCDIC-to-ASCII Conversion	314
ASCII-to-EBCDIC Conversion	315
Chapter 10. Workstation Gateway Server	319
Accessing Workstation Gateway Functions through Operations Navigator	319
Starting the Workstation Gateway Server	320
Automatically Starting the Workstation Gateway Server.	320
Ending the Workstation Gateway Server	320
Configuring the Workstation Gateway Server	321
Managing Virtual Devices for the Client	322
Changing the Workstation Gateway Configuration.	323
Number of Clients per Server (NBRCLT)	324
Inactivity Timeout (INACTTIMO)	325
Data Request Timeout (DTARQSTIMO)	325
Display Sign-on Panel (DSPSGN)	325
Access Logging (ACCLOG)	327
Top Banner URL (TOPBNRURL)	327
Bottom Banner URL (BOTBNRURL).	327
Help Panel URL (HLPPNLURL)	327
Coded Character Set Identifier (CCSID)	328
Server Mapping Tables (TBLWSGOUT) and (TBLWSGIN).	328
Workstation Gateway Exit Point for Accessing a User Profile Directly	329
Granting Access to the Web Browser Online Help Information	329
Customizing Web Browser Online Help Information	330
Managing the Access Log	330
The QATMTLOG File	330
Accessing the Workstation Gateway from a Web Browser.	332
Security	334
Workstation Gateway — Requirements	335
How the 5250 Display is Formatted for the Workstation Gateway	335
Configuration Examples	337
Online Help Information	338
Chapter 11. Line Printer Requester (LPR)	345
LPR Command	345
Client (LPR) and Server (LPD) Relationship	345
Configuration Requirements for LPR	346
Sending a Spooled File (LPR)	346
Step 1 — Locate the Spooled File that you Want to Send.	346
Step 2 — Start the Spooled File Transfer	347
Sending Spooled Files to an AS/400 at V2R3 or V3R0M5.	348
How the System Sends a Spooled File from an AS/400 System to Another AS/400 System	348
How the System Sends a Spooled File from an AS/400 System to a Non-AS/400 System.	349
Transformation of Spooled Files	350
Sending Spooled File — Tips	353
Sending Large Spooled Files	353
Printer Pass-Through	353
Setup	354
Starting Printer Pass-Through	355
Configuring for a RISC System/6000 System — Scenario	356
Setting Up for LPD on the RISC System/6000 System — Scenario	356
Configuring Device and Virtual Printer for AIX Printing	356
Verifying LPD Started on the RISC System/6000 System	358
Verifying Your Configuration on the RISC System/6000 System.	359

Print Services Facility/6000 Function	360
Configuring PSF/6000 Function	360
Verifying Your Configuration of PSF/6000	361
Chapter 12. Line Printer Daemon (LPD)	363
Configuring for Line Printer Daemon (LPD)	363
How the Destination System Receives a Spooled File	363
How an AS/400 System Receives a Spooled File from Another AS/400 System	364
How an AS/400 System Receives a Spooled File from a Non-AS/400 System	365
How Spooled Files are Named on the Destination AS/400	366
Starting an LPD Server Job	366
Ending an LPD Server Job	367
Attributes of the Received Spooled File	367
User Profile Library Lists	367
How the Ownership of Spooled Files Is Determined	370
How an LPD Server Selects an Output Queue for a File	370
How Authority for Putting Spooled Files on Output Queue is Determined	372
Using LPD to Print ASCII Files.	373
Using LPD to Print ASCII Files Converted to EBCDIC	373
Authority Required to Receive Spooled Files	375
Chapter 13. BOOTP Server	377
Accessing BOOTP Functions through Operations Navigator	377
Starting the BOOTP Server	378
Automatically Starting the BOOTP Server.	378
Ending the BOOTP Server	378
Configuring the BOOTP Server	378
Changing BOOTP Attributes.	379
Working with the BOOTP Table	379
Adding IBM Network Stations to an Existing BOOTP Environment.	379
Adding Network Stations with the Command Line Interface	380
Adding Network Stations with Operations Navigator	380
Chapter 14. TFTP Server	383
Accessing TFTP Functions through Operations Navigator	383
Starting the TFTP Server	383
Automatically Starting the TFTP Server	383
Ending the TFTP Server	384
Changing TFTP Attributes	384
Server and Client Ports	385
TFTP Extensions.	385
TFTP Transfer Size Option	385
TFTP Subnet Broadcast Option	385
Configuring TFTP for Clients other than IBM Network Station	389
Chapter 15. RouteD Server	391
Accessing RouteD Functions through Operations Navigator	391
Starting the RouteD Server	391
Automatically Starting the RouteD Server.	392
Ending the RouteD Server	392
Configuring the RouteD Server	392
Working with RouteD Configuration	393
RouteD Configuration Scenario	393
RIP_INTERFACE Statement	394

Supply Values	395
DIST_ROUTES_IN	395
Metric	395
Community	396
Additional Parameters	396
Changing Routed Attributes	397
Chapter 16. REXEC Server	399
Accessing REXEC Functions through Operations Navigator	399
Starting the REXEC Server from the Command Line Interface	399
Automatically Starting the REXEC Server	400
Ending the REXEC Server	400
Changing Attributes	400
REXEC Command Considerations	400
Selecting a Command Processor	401
REXEC Connection Usage	401
For AS/400 CL command processing	401
For Qshell and spawned path command processing	401
Spooled Output Considerations	402
Client Considerations	402
REXEC Server Jobs and Job Names	402
Creating REXEC Server Spooled Job Logs	403
Exit Points for Controlling REXEC Server	403
Chapter 17. DHCP Server	405
DHCP Overview	405
What is DHCP?	405
Planning for DHCP	406
Setting Up a DHCP Network	408
Specifying DHCP Options	412
Request for Comment and Internet Draft Documents	413
Accessing DHCP Functions through Operations Navigator	414
Starting and Ending the DHCP Server from the Command Line Interface	415
Starting the DHCP Server	415
Automatically Starting the DHCP Server	415
Ending the DHCP Server	416
Changing DHCP Attributes	416
Exit Points for a DHCP Server	417
Examples of DHCP Configurations	417
Configuring DHCP for a Local Area Network	417
Configuring DHCP for a Local Area Network with a Router	417
Using DHCP to Configure Clients Attached to a Twinax Workstation Controller	418
Migrating an Existing BOOTP Configuration	419
DHCP Relay Agent	420
Adding Network Stations	420
Chapter 18. AS/400 Domain Name System (DNS)	421
How DNS works	421
Additional DNS documentation	422
Chapter 19. Client SOCKS Support	423
Accessing SOCKS Functions through Operations Navigator	423
Chapter 20. TCP/IP Performance	425
*BASE Pool Size	425

TCP/IP Jobs	425
TCP/IP Protocol Support Provided by IOP	425
Merge Host Table Performance	427
Running TCP/IP Only: Performance Considerations	427
Chapter 21. TCP/IP Problem Analysis	429
General TCP/IP Problems	430
PING Command Considerations	437
Working with the Job Log and Message Queues	438
Determining Problems for SNMP	439
Determining Problems for Serial Line Internet Protocol (SLIP)	439
Problem: SLIP Connection Is Failing	439
Problem: SLIP Job 'Hung' with STRSSN Status	442
Problem: SLIP Connection Complete but Unable to PING	442
Materials Required for Reporting SLIP Problems	442
Determining Problems with TELNET.	443
Materials Needed when Reporting TELNET Problems	445
Determining Problems with FTP	450
Materials Required for Reporting FTP Problems	452
Tracing FTP Server	453
Tracing FTP Client	456
Getting a Copy of an FTP Server Job Log	456
Determining Problems for SMTP	457
Determining Problems for SMTP When Using OfficeVision	462
Determining Problems for SMTP Without Using OfficeVision	464
Tracing SMTP Distributions	469
Materials Required for Reporting SMTP Problems	475
Cleaning Up Unprocessed SMTP Distributions	475
Determining Problems with the POP Server	476
Problems with Mail Delivery	476
Problem Determination Flows	477
Determining Problems with the Workstation Gateway Server.	480
First Failure Data Capture (FFDC)	480
Determining Problems for DNS Server	482
Problem Determination Tools	482
Problem Determination Flows	483
Determining Problems for LPR.	486
LPR Command Considerations	486
Common Error Messages	486
Determining Problems for LPD.	487
Materials Required for Reporting LPD Problems	489
Determining Problems with REXEC	489
Materials Required for Reporting REXEC Problems	491
Getting a Copy of an REXEC Server Job Log	491
Tracing the REXEC Server	491
Tracing TCP/IP Protocol Layer Problems	492
APPC Over TCP/IP Debugging Capabilities	492
Tracing APPC over TCP/IP Problems	493
Collecting a Communications Trace	493
Planning to Set up a Trace	493
Starting a Communications Trace.	495
Stopping a Communications Trace	498
Formatting and Saving the Communications Trace	499
Verifying the Contents of the Communications Trace.	501
Using the Product Activity Log for TCP/IP Problem Analysis	503

Appendix A. Configuring a Physical Line for TCP/IP Communication	505
Configuration Steps	506
Creating the Line Description	506
Setting the Maximum Transmission Unit	507
Determining the Maximum Size of Datagrams	507
Appendix B. TCP/IP Security	509
TCP/IP Command Security	509
Object Security for Network Configuration	513
IBM-Written Programs Security	513
Customer-Written Programs Security	514
User-Supplied Mapping Tables	514
Appendix C. Mapping Tables Associated with TCP/IP Function	517
National Language Support Mapping	517
Summary of Mapping Tables	518
Creating ASCII and EBCDIC Mapping Tables	518
Creating a Source Member for Incoming Data	519
Creating a Source Member for Outgoing Data	519
Creating a Mapping Table	520
Specifying User-Defined ASCII and EBCDIC Mapping Tables	520
Creating 3270 Mapping Tables	521
Creating a Source Member for Incoming Data	521
Creating a Source Member for Outgoing Data	522
Creating a Mapping Table	522
Using Mapping Tables for 3270 Full-Screen Mode	523
Reading a Mapping Table	523
Changing a Mapping Table	523
Sample Mappings	523
EBCDIC and ASCII Character Sets	524
USA Standard 7-Bit ASCII Character Set	526
EBCDIC-to-ASCII Mapping Table	527
ASCII-to-EBCDIC Mapping Table	529
ASCII Line Drawing Character Set	530
Appendix D. TELNET 3270 Keyboard Mappings	531
AS/400 CL Programs for the CHGKBDMAP Command	531
Appendix E. TCP/IP Application Exit Points and Programs	535
TCP/IP Exit Points and Exit Programs	535
OS/400 Registration Facility	535
TCP/IP Application Exit Points	536
Creating Exit Programs	537
Adding Your Exit Program to the Registration Facility	537
Removing Exit Programs	540
TELNET Exit Points	541
Telnet Device Initialization Exit Program	541
TELNET Device Termination Exit Program	546
Required Parameter Group	547
Exit Point Interfaces for TCP/IP Application Exit Points	547
TCP/IP Application Request Validation Exit Point Interface	547
TCP/IP Application Server Logon Exit Point Interface	551
Remote Execution Server Command Processing Selection Exit Point	551
File Transfer Protocol (FTP) Exit Points	553
Considerations and Recommendations for FTP Exit Programs	554
FTP Exit Program—Scenario	554

Sample FTP Server Logon Exit Program (C Language)	555
Anonymous FTP	568
Sample Scenario for Anonymous FTP	569
Workstation Gateway Server (WSG) Exit Point	569
Workstation Gateway Server Sign-on Exit Point Interface (QAPP0100)	
Required Parameters	569
Descriptions of Required Parameters for the WSG Exit Point Interface	
(QAPP0100)	570
Using a WSG exit program to bypass the AS/400 Sign-on Display	571
Sample WSG Server Logon Exit Program	572
Notices	595
Programming Interface Information	596
Trademarks	596
Bibliography	599
Client Access Books	599
Communications Manuals	599
Integrated Netfinity Server Manuals	600
Internet Connection Server Manuals	600
Programming Manuals	600
Security	601
System Manuals	601
Systems Network Architecture (SNA) Display Stations	601
Request For Comments (RFC).	602
Other Information	602
Index	605
Readers' Comments — We'd Like to Hear from You.	635

About TCP/IP Configuration and Reference (SC41-5420)

This book contains information about configuring and using Transmission Control Protocol/Internet Protocol (TCP/IP) and writing programs to the TCP/IP application interface. Some topics have moved to the Information Center. See "TCP/IP Topics in the Information Center".

TCP/IP Packaging:

When you purchase OS/400, the ordering system automatically places an order for the TCP/IP Utilities Licensed Program (LP). The TCP/IP Utilities LP is shipped with OS/400 at *no additional* charge. However, you must install the TCP/IP Utilities LP separately by using normal installation support.

TCP/IP Topics in the Information Center

These related topics are described in the *AS/400e Information Center* instead of in this book:

Under the TCP/IP heading:

- Getting started with TCP/IP
- Connecting two systems with Point-to-Point Protocol (PPP)
- Logging on to a remote computer (Telnet)
- Managing host names with Domain Name System (DNS)
- Sending and receiving e-mail
- Transferring files with File Transfer Protocol (FTP)

Under the Internet and Secure Networks heading:

- Client SOCKets
- IP packet security
- Virtual Private Network (VPN)

You can access the Information Center from the AS/400e Information Center CD-ROM (English version: *SK3T-2027*) or from one of these Web sites:

<http://www.as400.ibm.com/infocenter>

<http://publib.boulder.ibm.com/pubs/html/as400/infocenter.htm>

To find out more about the Information Center, see "Prerequisite and related information" on page xvii.

Who should read this book

This book is intended for the following audience:

- Users who configure TCP/IP and its associated applications
- Users who use TCP/IP functions or commands
- Programmers who write to the sockets applications programming interface, or API. For details on the AS/400 sockets API see *System API Reference*, SC41-5801-03. For additional information about sockets, including sample programs, see the *Sockets Programming*, SC41-5422-03 book. For information

about firewall concepts and an IBM firewall product for the AS/400 system, see the *Getting Started with IBM Firewall for AS/400*, SC41-5424-02 book or go to <http://www.as400.ibm.com/firewall>.

You need to be familiar with, or have previous experience in, the following areas:

- TCP/IP. If this is your first experience with TCP/IP, consider reading some of the material listed in the Bibliography.
- AS/400 menus and commands.
- Operations Navigator. Some of the applications require the use of this function.
- Writing applications on the AS/400 system. If you plan to write programs to the TCP/IP application program interface, you must know how to write applications on the AS/400 system.

AS/400 Operations Navigator

AS/400 Operations Navigator is a powerful graphical interface for Windows clients. With AS/400 Operations Navigator, you can manage and administer your AS/400 systems from your Windows desktop.

You can use Operations Navigator to manage communications, printing, database, security, and other system operations. Operations Navigator includes Management Central for managing multiple AS/400 systems centrally.

Figure 1 shows an example of the Operations Navigator display:

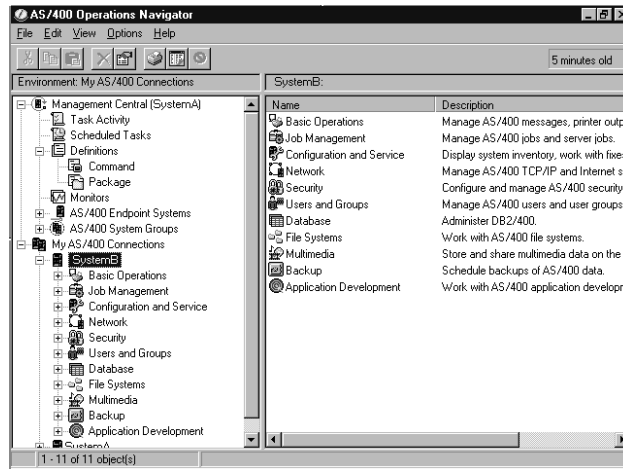


Figure 1. AS/400 Operations Navigator Display

This new interface has been designed to make you more productive and is the only user interface to new, advanced features of OS/400. Therefore, IBM recommends that you use AS/400 Operations Navigator, which has online help to guide you. While this interface is being developed, you may still need to use a traditional emulator such as PC5250 to do some of your tasks.

Installing Operations Navigator

To use AS/400 Operations Navigator, you must have Client Access installed on your Windows PC. For help in connecting your Windows PC to your AS/400 system, consult *Client Access Express for Windows - Setup*, SC41-5507-00.

AS/400 Operations Navigator is a separately installable component of Client Access that contains many subcomponents. If you are installing for the first time and you use the **Typical** installation option, the following options are installed by default:

- Operations Navigator base support
- Basic operations (messages, printer output, and printers)

To select the subcomponents that you want to install, select the **Custom** installation option. (After Operations Navigator has been installed, you can add subcomponents by using Client Access Selective Setup.)

1. Display the list of currently installed subcomponents in the **Component Selection** window of **Custom** installation or Selective Setup.
2. Select AS/400 Operations Navigator.
3. Select any additional subcomponents that you want to install and continue with **Custom** installation or Selective Setup.

After you install Client Access, double-click the **AS400 Operations Navigator** icon on your desktop to access Operations Navigator and create an AS/400 connection.

Prerequisite and related information

Use the AS/400 Information Center as your starting point for looking up AS/400 technical information. You can access the Information Center from the AS/400e Information Center CD-ROM (English version: *SK3T-2027*) or from one of these Web sites:

<http://www.as400.ibm.com/infocenter>
<http://publib.boulder.ibm.com/pubs/html/as400/infocenter.htm>

The AS/400 Information Center contains important topics such as logical partitioning, clustering, Java, TCP/IP, Web serving, and secured networks. It also contains Internet links to Web sites such as the AS/400 Online Library and the AS/400 Technical Studio. Included in the Information Center is a link that describes at a high level the differences in information between the Information Center and the Online Library.

For a list of related publications, see the “Bibliography” on page 599.

How to send your comments

Your feedback is important in helping to provide the most accurate and high-quality information. If you have any comments about this book or any other AS/400 documentation, fill out the readers’ comment form at the back of this book.

- If you prefer to send comments by mail, use the readers’ comment form with the address that is printed on the back. If you are mailing a readers’ comment form from a country other than the United States, you can give the form to the local IBM branch office or IBM representative for postage-paid mailing.
- If you prefer to send comments by FAX, use either of the following numbers:
 - United States and Canada: 1-800-937-3430
 - Other countries: 1-507-253-5192
- If you prefer to send comments electronically, use one of these e-mail addresses:
 - Comments on books:

RCHCLERK@us.ibm.com

IBMMAIL, to IBMMAIL(USIB56RZ)

– Comments on the AS/400 Information Center:

RCHINFOC@us.ibm.com

Be sure to include the following:

- The name of the book.
- The publication number of the book.
- The page number or topic to which your comment applies.

Chapter 1. TCP/IP on AS/400

Transmission Control Protocol/Internet Protocol (TCP/IP) refers to a specific set of protocols that allows computers to share resources and exchange information in a network. Because TCP and IP are two of the best-known protocols in this set, the term TCP/IP has become the standard name for the whole family.

TCP/IP support on AS/400 contains many of the commonly used protocols in the TCP/IP family. Some of the protocols provide low-level and high-level data delivery functions that are needed by several applications.

Most applications require Internet Protocol (IP), Transmission Control Protocol (TCP), and User Datagram Protocol (UDP), which are low-level functions. High-level functions or applications provide services such as file transfer, mail, and remote logon. File Transfer Protocol (FTP), Simple Mail Transfer Protocol (SMTP), and TELNET are examples of high-level protocols.

TCP/IP is used to interconnect networks on a global basis across universities, research institutions, businesses and industries, and military installations. The term **Internet** applies to this entire set of interconnected networks. Because these networks are interconnected, information is sent from one to another as security restrictions permit. The Internet is governed by a central authority, which is responsible for assigning network addresses to new users and subnetworks. Many smaller private networks around the world use TCP/IP protocols, without connecting to the Internet. TCP/IP provides a good solution for these smaller networks.

Linking Networks Together

Networks are linked together by sharing a common node system that routes packets from one network to another. This common system is often referred to as an *IP router*, or simply as a router. A router is a computer that directly attaches to two or more IP networks and, as its name implies, routes packets from one network to the other.

The networks connected by a router may use the same or different physical network protocols. For example, one network could be an Ethernet LAN, and the other might be a token ring. If necessary, the router transfers packets from one network protocol to another. In this way, the system passes the packets from one router to another. The system continues transfer attempts until it delivers the packet to the final destination system directly across one physical network.

The terms *gateway* and *router* are often used interchangeably, particularly in this publication. However, be aware that gateway often implies a more specialized system. A gateway typically performs extra functions beyond the mere routing of packets. For example, a gateway might provide a firewall.

You can use a bridge to connect networks. A bridge connects two or more networks at the physical network level by forwarding data from one network to another. A bridge differs from IP routers because a bridge uses physical addresses instead of internet addresses. A bridge does not have an assigned internet address while an IP router does. Therefore, a bridge is transparent in the TCP/IP network. A set of bridged networks (or segments) appears and acts as a single physical network.

A network to which a node system is physically connected is a **local network** for that system. A network that a system reaches only after passing through one or more IP routers is called a **remote network**.

Internetwork Communications

An internetwork or **internet** is a collection of packet-switched physical networks that are connected by routers to form a single, large virtual network. Simply, it is a network of networks. **Packets** are units of data that are sent across packet-switched networks. All nodes in the internet communicate as if they are on the same physical network, regardless of their specific hardware or specific software architecture. This cooperation among otherwise incompatible networks and systems is known as *interoperability*.

Figure 2 shows how networks can be connected in an internet.

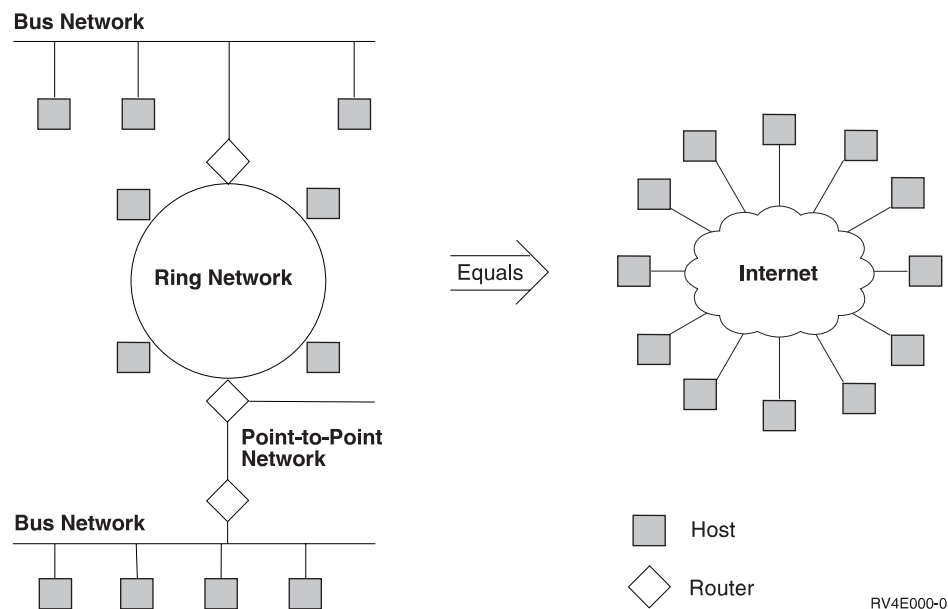


Figure 2. An Internetwork

The network connection of each node on an internetwork is assigned a unique address. This internet address differs from a physical hardware address in that the hardware address is often preset by the manufacturer, whereas you can assign or reassign an internet address by standard conventions. Also, internet addresses are in a standard form, while different hardware types use different address lengths and formats.

Internet Addresses

Each node on a network is known as a *host* and has a unique address called an *internet address*. This address is a 32-bit integer. An address is expressed in the form *nnn.nnn.nnn.nnn*, where each field is the decimal representation of one byte, or 8 bits, of the address. For example, the address whose hexadecimal representation is X'82638001' is expressed as 130.99.128.1.

Within your own networks, you can assign your own addresses. However, if you want to connect to the Internet, then your network addresses and domain names that need to be visible on the Internet must be officially assigned by a central authority. The authority at the time of this writing is Network Solutions, Inc. The address is:

Network Solutions, Inc.
ATTN: InterNIC Registration Services
505 Huntmar Park Drive
Herndon, VA 20170
USA
1-703-742-4777
FAX: 1-703-742-9552
E-mail: hostmaster@internic.net
URL: <http://rs.internic.net/>

Information about how to register a new domain name through the InterNIC is found at the URL address which is listed above. This address contains registration tools and forms for online registering, and an overview on domain name registration.

The Internet Assigned Numbers Authority, or IANA, has allocated a block of Internet network addresses for use only in private networks, or intranets. The addresses are dispensed through the InterNIC registration process. IANA guarantees that the addresses within this range are not used as valid host internet addresses on the Internet. The reserved addresses within this block are as follows:

10.0.0.0	-	10.255.255.255
172.16.0.0	-	172.31.255.255
192.168.0.0	-	192.168.255.255

Accessing the Internet

To obtain Internet access you must purchase it through an Internet Service Provider (ISP). If you need assistance, the InterNIC provides you with contact information for ISPs in the United States and other countries. See "Internet Addresses" on page 2 for address information to contact the InterNIC Registration Services.

Once you have selected an ISP to work with, the ISP obtains an internet address for you. If you choose not to connect to the Internet through an ISP but plan to do so directly, you still need to apply to the InterNIC for a domain address and an IP network ID. However, in order to apply to the InterNIC without the assistance of an ISP, you must be either a service provider or a large global corporation.

IBM also offers ISP services as part of its Internet Connection family of service offerings. To contact IBM within the United States, call 1-800-IBM-4YOU (1-800-426-4698), or you may call your local IBM office.

In addition, the following IBM Redbooks offer more information about choosing an Internet Service Provider:

- *Cool Title About the AS/400 and the Internet, SC24-4815*
- *Accessing the Internet, SG24-2597*
- *Using the Information Super Highway, GG24-2499*

IP Security

After choosing an Internet Service Provider (ISP) and setting up your Internet connection, you will also need to create and implement a security policy. Such a policy can be used to incorporate the rules governing computer resources and communications resources within your organization. The inherent security features of AS/400, when properly configured, provide you with the ability to minimize many risks. However, when you connect to the Internet, you should consider additional security measures to further ensure the safety of your AS/400 system and your network.

The first step in developing a security policy is that you understand the risks that are imposed by each service you intend to use or provide. Once you have identified these risks and created a security policy in response to them, you will be prepared to take the necessary steps to enforce them. To name a few, these steps may include employee training and the purchase of additional hardware or software.

As you create a security policy and outline security objectives for your organization, the following resources may be helpful:

- The book, *Tips and Tools for Securing Your AS/400*, SC41-5300-03
- The *AS/400e Information Center* offers a list of current topics about using the Internet. Look there for information about IP packet filtering and network address translation (NAT). It is located at the following URL address:

<http://publib.boulder.ibm.com/html/as400/infocenter.html>

Classes of Networks

Each internet address is comprised of a pair of numbers that correspond to its **network address**, or network ID and **host address**, or host ID. The network ID represents the network within the internet, and the host ID specifies an individual host or router within the network.

internet address = <network ID><host ID>

The value of the first byte of the Internet address specifies how the Internet address should be separated into its network and host part, as shown in Table 1. The 4-byte address is divided between network ID and host ID in five different ways or classes. The five classes of Internet addresses are: A, B, C, D, and E. Also shown is the maximum number of hosts per network for each class.

Table 1. *Classes of Networks*

Network Class	Range of First Byte	Network ID	Host ID	Maximum Number of Hosts per Network Class
Class A	0 to 127 ¹	First byte	Last 3 bytes	16 777 214
Class B	128 to 191	First 2 bytes	Last 2 bytes	65 534
Class C	192 to 223	First 3 bytes	Last byte	254
Class D	224 to 239			Multicast
Class E ²	240 to 255			Reserved for future use

Table 1. Classes of Networks (continued)

Network Class	Range of First Byte	Network ID	Host ID	Maximum Number of Hosts per Network Class
Notes:				
1. Although 127 is a class A network ID, it is reserved for loopback addresses and cannot be assigned.				
2. Not supported by AS/400 except for the limited broadcast address of 255.255.255.255.				

If the first byte of an Internet address is in the range 0 to 127, it is a **Class A** network. These are very large networks. The host IDs can range from 0.0.1 to 255.255.254, which allows for a maximum of 16,777,214 hosts. An example of a class A Internet address is 9.5.1.2. The network ID is 9 and the host ID is 5.1.2.

If the first byte of an Internet address is in the range 128 to 191, it is a **Class B** network. These are medium-size networks. The host IDs can range from 0.1 to 255.254, which allows for a maximum of 65,534 hosts. An example of a class B Internet address is 150.244.1.241. The network ID is 150.244 and the host ID is 1.241.

If the first byte of an Internet address is in the range 192 to 223, it is a **Class C** network. These are relatively small networks. The host IDs can range from 1 to 254, which allows for a maximum of 254 hosts. An example of a class C Internet address is 221.6.1.244. The network ID is 221.6.1 and the host ID is 244.

If the first byte of an Internet address is in the range 224 to 239, it is a **Class D** network. These addresses identify multicast networks. The first 4 bits in a class D Internet address identify the Internet address as a multicast address. The rest of the bits (28) identify a specific multicast group.

If the first byte of an Internet address is in the range 240 to 255, it is a **Class E** network. These addresses are undefined and experimental at this time. AS/400 does not support class E Internet addresses except for the limited broadcast address of 255.255.255.255.

Understanding network classes and how the Internet address is divided into different classes is important for the discussion of subnetworks in the following topic.

IP Subnets

An IP subnet is the division or split of a TCP/IP network. When you divide an IP network into multiple subnets, you avoid having to request additional IP network addresses and alleviate the waste of existing addresses. "Subnetting" is frequently performed on class A and B internetworks because they are larger and contain more available host address space.

Also worth noting is the fact that subnets can be assigned locally and the process of subnetting can remain transparent to remote networks.

Subnetworks and Subnet Masks

A class A, B, or C network can be further divided into multiple smaller networks that are called subnetworks. These smaller networks are identified or addressed by a subnet ID. With an Internet address of any class we can subdivide the host ID bits to provide identification of the subnet ID. The subnetwork is identified by combining the network ID and subnet ID. This combining of the network ID and subnet ID is defined as a subnet mask.

A subnet mask further specifies for the host the exact number of bits that are to be used for the subnet ID and the number to be used for the host ID. Overall, subnet masks make it easier to divide a single network without wasting internet addresses and as a result, increase the performance of the network.

For example, consider a class B internet address that has a network ID portion equal to 144.22. This class B address has 16 bits allocated for the network ID and 16 bits allocated for the host ID as shown in Figure 3:

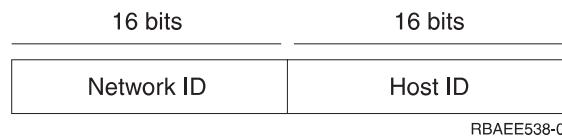


Figure 3. Standard Class B

If the high-order byte of the host ID portion of the class B internet address is used for a subnet ID, then the host ID can be divided into an 8-bit subnet ID and an 8-bit host ID as shown in Figure 4:

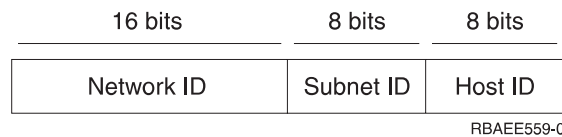


Figure 4. Subnetted Class B

The class B 144.22 network, using this identification of the subnet ID, can be divided into 254 subnetworks ranging from 144.22.1 through 144.22.254.

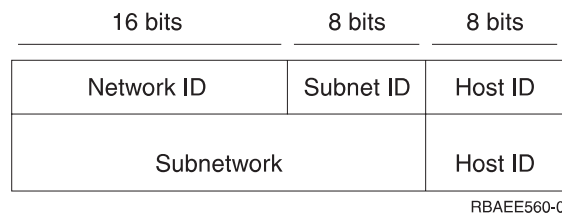


Figure 5. Subnetted Class B with Subnetworks

The subnet mask that identifies the subnetwork in this example would be 255.255.255.0.

Therefore, an internet address can be shown as consisting of the following:

internet address = <network id><subnet id><host id>

Note: The subnet ID does not have to be identified by contiguous bits in the host ID portion of the internet address. The subnet ID can be identified by using noncontiguous bits in the host ID portion of the internet address. It is strongly recommended to use contiguous bits to identify the subnet ID.

Assume, for example, two networks where subnetting is used: subnetwork 9.4.70 and subnetwork 9.4.73.192. The first byte of the internet address is 9, which makes it a class A network. 9.4.70 is an example of a network where two complete bytes of the host ID have been used for the subnet ID. 9.4.73.192 is an example where two complete bytes and part of a third byte of the host ID have been used for the subnet ID. For 9.4.70, the range of host IDs available is 9.4.70.1 through 9.4.70.254. For 9.4.73.192, the range of host IDs available is 9.4.73.193 through 9.4.73.254.

A subnet mask is used to distinguish between a subnet ID and a host ID. Bits in the subnet mask are set to 1 if the network treats the corresponding bit in the internet address as part of the network ID and subnet ID and to 0 if it treats the bit as part of the host ID. For example, a subnet mask of:

```
Subnet mask      = 11111111 11111111 11111111 00000000
```

specifies that the first three bytes identify the subnetwork and the fourth byte identifies the host. Remember that the network portion of the internet address is based on the class of internet address (A, B, C, D or E). The subnet mask must at a minimum mask off the network portion of the address. In this case, because we are discussing a class A address, the first byte must be all 1s. A subnet mask would normally be specified in the same dotted decimal notation as an internet address. Because the second and third bytes are being used to identify the subnet ID, those two bytes should also be all 1s. Thus, in this example, the subnet mask is 255.255.255.0. If an internet address of 9.4.70.254 is compared against the subnet mask:

```
Subnet mask      = 11111111 11111111 11111111 00000000
Internet address = 00001001 00000100 01000110 11111110
(logical and)    = -----
Subnetwork       = 00001001 00000100 01000110 00000000
```

the subnetwork is 9.4.70 and the host ID is 254.

In the second example, the subnet mask for subnetwork 9.4.70 is 255.255.255.0. The subnet mask for subnetwork 9.4.73.192 is 255.255.255.192. This is an example of a subnetwork where part of a host ID byte has been used for the subnet ID. In this case, the upper 2 bits of the fourth byte are used as part of the subnet ID. In bit form, the subnet mask is:

```
Subnet mask      = 11111111 11111111 11111111 11000000
```

This means that the first 3 bytes and the first 2 bits of the fourth byte identify the subnetwork, and the last 6 bits of the fourth byte identify the host. If an internet address of 9.4.73.212 is compared against the above subnet mask:

```
Subnet mask      = 11111111 11111111 11111111 11000000
Internet address = 00001001 00000100 01001001 11010100
(logical and)    = -----
Subnetwork       = 00001001 00000100 01001001 11000000
```

The subnetwork is 9.4.73.192 and the host ID is 20.

It is also possible to have a subnet mask where the network ID part is not contiguous. For example:

Subnet mask = 11111111 11111111 01100000 01000000

This method is not normally used and is not recommended.

For AS/400 business computing systems in the above network, the full route table could be:

```
Work with TCP/IP Routes                                System:  SYSNAM01
Type options, press Enter.
  1=Add  2=Change  4=Remove  5=Display

Opt  Route      Subnet      Next      Preferred
    Destination  Mask        Hop        Interface
-----
-    *DFTRROUTE  *NONE      9.4.73.193 *NONE
-    9.4.70.0    255.255.255.0 9.4.73.194 *NONE

Bottom
F3=Exit  F5=Refresh  F6=Print list  F10=Work with IP over SNA routes
F11=Display type of service  F12=Cancel  F17=Top  F18=Bottom
```

Figure 6. AS/400 TCP/IP Subnet and Subnet Mask

The default is to route all traffic to the router associated with that subnetwork. No routing configuration is required within the same subnetwork.

All hosts in the same subnetwork must use the same subnet mask.

Broadcast Addresses

Broadcast addresses are used to address multiple recipients on a network as opposed to addressing a single recipient, which is done through unicast addresses. Another type of IP address used for the purpose of sending a message to multiple nodes on a network is the multicast address. In the paragraphs that follow, broadcast addresses are discussed. For more information about IP multicasting, see "IP Multicasting" on page 91.

While there are several types of broadcast addresses, they can be separated into two categories: the limited broadcast address and the directed broadcast address.

On each physical network, a **limited broadcast address** is an internet address that consists of all 1 bits (255.255.255.255.). A **directed broadcast address** is an internet address with a valid network ID and a host ID of all 1 bits. There are different types of directed broadcast addresses, including network-directed, subnet-directed, and all-subnets-directed.

Limited broadcasts only apply to the physical network. Packets addressed to the limited broadcast address are not forwarded beyond the physical network of origin.

Domain Name System (DNS)

An Internet address is used to establish a connection between your system and another system in a network. Because Internet addresses can sometimes be difficult to remember, another naming convention is used. This naming convention is called the Domain Name System (DNS). DNS names provide an easier way to identify systems in a network.

When DNS is configured to run as part of TCP/IP, it is referred to as a DNS server. For more information about AS/400 DNS server configuration and functions, see “Chapter 18. AS/400 Domain Name System (DNS)” on page 421.

Domain and Host Name

A **domain name** identifies where your system is located within a hierarchy of groups of systems. Domain names consist of labels that are separated by periods (for example, ABC.DEF.XYZ). Within a domain represented by a domain name, there can be many systems, and each must have a unique host name. A system's host name can be used by remote servers to associate an Internet address with the system. Since host names are only unique within a domain, a host name is usually combined with a domain name in the form **host.domain**, which is called a fully-qualified host name.

Note: SMTP often refers to the host or host.domain combination as a domain. For clarity, it is referred to in this publication as a **SMTP domain**.

When a network is small, the host names can be a sequence of characters without any further structure. The advantage of this kind of name is that it is convenient and easy to remember. The disadvantage is that it is not suitable for large sets of machines for the following reasons:

- When the number of sites increases, the potential for conflicting names increases.
- The authority for adding new names must rest at a single site, and the administrative work load increases as the number of sites increase.
- The name-to-address bindings change frequently, and the cost of maintaining correct copies at each site is high.

The answer to these problems is the decentralization of the naming process by delegating authority for parts of the network. The name area of the network is partitioned at the top level. The top level delegates authority (where it has some authority) for the partitions. The top level need not be bothered by changes within the different partitions.

The syntax of hierarchically assigned names reflects the hierarchical delegation of authority used to assign them. For example, the names of the form *local.site* mean that *site* is the top level of the hierarchy and this name has been authorized by the central authority; *local* is a host name controlled by the *site*. The authority can be further subdivided at each level. In this example, the *site* can be divided into two groups:

- Production

- Marketing

Syntactically, adding a new level introduces another subdivision to the name. In this example, the machine can belong to the marketing group with the name: *local.market.site*. *Local* is the host name, and *market.site* is the domain name.

Naming Conventions for Domain Names and Host Names

Normally the hierarchical machine names are assigned according to the structure of organizations that obtain authority for parts of the namespace, not according to the structure of the physical network interconnections.

A domain name or a host name can be a text string having 2 to 255 characters. Domain names consist of one or more labels separated by periods. Each label can contain up to 63 characters. The following characters are allowed in domain names:

- Alphabetical characters A through Z
- Digits 0 through 9
- Underline (_)

Note: The Underline character (_) is not fully supported by all implementations outside of AS/400.

- Minus sign (-)
- Period (.). Periods are only allowed when they separate labels of domain style name (refer to RFC 1034, *Domain Names - Concepts and Facilities*).

Other naming conventions for domain names and host names include the following:

- Uppercase characters and lowercase characters are allowed, but no significance is attached to the case.
- The first character of each part of the name separated by periods must be an alphabetic character (uppercase A-Z or lowercase a-z).
- The last character of each part of the name separated by periods must be an alphabetic character (uppercase A-Z or lowercase a-z) or a numeric character (0-9).
- Blanks cannot be embedded.
- Only the special characters period (.), minus sign (-), and underline (_) are allowed.
- Parts of the name separated by periods (.) cannot exceed 63 characters.
- Each part of the name must be at least 1 character.
- Fully-qualified host names including all periods cannot be more than 255 characters.
- Advanced program-to-program communications (APPC), over TCP/IP (part of AnyNet/400) uses the host name to map location names to Internet addresses. The host name must be in the form:

```
location.netid.SNA.IBM.COM
```

Where *location* is the remote location the program is opening to, and *netid* is the network identifier for this connection. *SNA.IBM.COM* is the qualifier that designates this as the APPC over TCP/IP domain.

Location names support characters that cannot be present in host names. Therefore, the APPC application can open only to locations that fulfill the TCP/IP host name syntax. This limits location names used for APPC over TCP/IP to the

characters A-Z (uppercase and lowercase), 0-9, \$ (dollar), @ (at sign), and # (number sign). The location name must begin with an alphabetic character.

- Try to limit your domain name labels to 12 characters because shorter labels are easier to remember.
- It is a common practice to use hierarchical names that allow predictable extensions for change and growth. Domain names normally reflect the delegation of authority or hierarchy that is used to assign them.

For example, the name *SYS1.MFG.ABC.COM* can be broken down into the following:

- COM: All commercial networks.
- ABC.COM: All systems in the ABC company's commercial network.
- MFG.ABC.COM: All manufacturing systems in the ABC company's commercial network.
- SYS1.MFG.ABC.COM:

A host that is named *SYS1* in the manufacturing area of the ABC company's commercial network.

The COM designation is one of several domain names that are used when connecting to the Internet. Some of the other domain names are as follows:

ARTS Entities with activities related to culture and entertainment

COM Commercial organizations

Country code

Countries that may be other than US

EDU Educational institutions

FIRM Business or firm

GOV Government institutions

INFO Entities that provide information services

MIL Military groups

NET Major network support centers (Internet)

NOM An individual or personal nomenclature

ORG Organizations

REC Entities with activities related to recreation and entertainment

STORE

Businesses offering goods to purchase

WEB Entities with activities related to the world wide web

For example, one of the domain names used in this book (*SYSNAM01.SALES.ABC.COM*) indicates the following:

- The *Internet* authority has assigned company "ABC" into the commercial organizations group.
- Company ABC authority has assigned this address to the group named SALES.
- Company ABC authority has assigned SYSNAM01 as the name for this IBM AS/400 business computing system.

Routing

Routing is the process of mapping a path to send a packet to its destination Internet address. Routing can be direct or indirect. Direct routing is used when the source and destination nodes are on the same physical network. When direct routing is used, the source node sends the packet on the network with the destination hardware address in the link layer protocol header. The destination node hardware detects its own address in the packet header and accepts the packet.

Indirect routing is used when the source and destination nodes are not on the same physical network. The source node uses its routing tables to determine which router will forward packets to the destination node. The packet is put on the network with the router's hardware address in the network header, and the destination's Internet address in the IP header. Thus, the router hardware receives the packet from the network, and the router IP software determines the packet's destination from the IP header.

Introduction to TCP/IP Protocols on AS/400

Network **protocols** are sets of rules that control the communication and transfer of data between two or more devices in a communications system. The term TCP/IP refers to a family of nonproprietary network protocols, of which **TCP**, providing host-to-host transmission, and **IP**, providing data routing from source to destination, are two important parts.

The topics that follow discuss only those protocols that are available on the AS/400 business computing system.

TCP/IP consists of a layered structure of protocols that range from low-level, hardware-dependent programs, to high-level applications. Each TCP/IP layer provides services to the layer above it and uses the services provided by the layer below it.

The layers are defined as follows:

Application

Provides a way for a process to cooperate with another process on the same or a different host.

Transport

Provides communication from one application program to another. Such communication is often called end-to-end data transfer.

Internetwork

Makes the entire physical network seem like a single virtual network. This is achieved by shielding the higher levels from the underlying network architecture.

Network Interface or Data Link

Provides the interface to the actual network hardware. Examples are token-ring and Ethernet.

Hardware

This layer is not part of the TCP/IP family and is represented by dotted lines. This layer consists of the hardware-specific network protocols.

Application Protocols

Application protocols are the highest-level protocols within the application layer. The application layer overall consists of several independent protocols that put into effect commonly used applications.

These protocols are able to communicate with applications on the same host or on different hosts. They serve as the user-visible interface to the TCP/IP protocol suite and use UDP and TCP protocols as methods for data transmission. Some of the most frequently used application protocols include FTP, SMTP, and TELNET.

Application Protocol Standards

The TCP/IP protocols include standards for many common applications. A discussion of each follows.

File Transfer Protocol (FTP)

FTP is an AS/400 TCP/IP application that enables you to transfer files between local and remote hosts. It does this through the use of the FTP subcommands, GET and PUT. FTP transfers files by using either an ASCII or EBCDIC mode. ASCII mode transfers data sets that contain only text characters.

Remote Printing (Line Printer Requester and Line Printer Daemon)

AS/400 TCP/IP provides client support and server support for remote printing. The client, line printer requester (LPR), allows the user to send spooled files to a remote system that is running a remote printer daemon (LPD) server.

Bootstrap Protocol (BOOTP)

BOOTP is a TCP/IP protocol that allows for booting systems remotely to the network. BOOTP is used to support the IBM Network Station for AS/400.

Trivial File Transfer Protocol (TFTP)

Trivial File Transfer Protocol (TFTP) is a simple protocol and is used solely to provide basic file transfers to and from a remote server. TFTP is used to support the IBM Network Station for AS/400.

Route Daemon (Routed) Server

The Route Daemon provides support for the Routing Information Protocol (RIP) on AS/400. RIP is the most widely used routing protocol today. RIP is an Internet Gateway Protocol used to assist TCP/IP in the routing of IP data packets.

Remote Execution (REXEC) Server

The Remote Execution server allows a client user to submit system commands to a remote server system.

Simple Mail Transfer Protocol (SMTP)

AS/400 TCP/IP provides for the exchange of electronic mail between host servers running TCP/IP by using SMTP. The SMTP application is based on end-to-end delivery, which means that an SMTP client contacts the destination host's SMTP

server directly in an effort to deliver the mail. The SMTP client will actually retain and retry transmission until the mail item is successfully delivered to the intended destination or recipient.

Post Office Protocol (POP) Mail Server

The POP server is the AS/400 implementation of the Post Office Protocol (POP) Version 3 mail interface. This server is a store-and-forward mail system that provides electronic mailboxes on AS/400 from which clients can retrieve mail. POP allows users to exchange multimedia mail.

The POP application allows AS/400 systems to act as POP servers for any clients that support the POP mail interface, including clients running on Windows, OS/2, AIX and Macintosh.

TELNET Protocol (TELNET)

AS/400 TCP/IP TELNET provides client and server support that allows remote logon to hosts within an internet.

Simple Network Management Protocol (SNMP)

AS/400 TCP/IP SNMP provides a means for managing an internet environment. SNMP allows network management by elements, such as routers and hosts. Network elements act as servers and contain management agents that perform the management functions requested. Network management stations act as clients; they run the management applications that monitor and control the network. SNMP provides a means of communicating between these elements and stations to send and receive information about network resources.

AS/400 can be an agent in an SNMP network. That is, the AS/400 system gathers information about the network and performs the management functions that are requested by some remote SNMP manager. At this time, AS/400 cannot be an SNMP manager in a TCP/IP network.

Simple Network Management Protocol (SNMP) is used predominately in TCP/IP networks. However, OS/400 AnyNet support allows OS/400 SNMP support to be used in a SNA network.

For additional information about SNMP, see the *Simple Network Management Protocol (SNMP) Support*, SC41-5412-00 book.

5250/Hypertext Markup Language (HTML) Workstation Gateway

The 5250/Hypertext Markup Language (HTML) Workstation Gateway (WSG) server is an application that automatically transforms AS/400 5250 applications to Hypertext Markup Language (HTML). This allows users to run AS/400 applications from any PC that has a WEB browser, and allows for TCP/IP network connectivity to the AS/400 host.

Domain Name System Server (DNS)

The Domain Name System (DNS) server is used by applications to translate domain names of hosts to IP addresses. The Domain Name System is the network naming service of intranets and the Internet.

Dynamic Host Configuration Protocol (DHCP)

The Dynamic Host Configuration Protocol (DHCP) provides a framework for passing configuration information to hosts on a TCP/IP network.

DHCP can function either as a DHCP server or a BOOTP/DHCP Relay Agent.

If DHCP functions as a DHCP server, it processes DHCP packets on the local system. If DHCP functions as a BOOTP/DHCP Relay Agent, it then relays DHCP and BOOTP packets on the local system, but does not process them.

Point-to-Point Protocol (PPP)

PPP is a method of transmitting datagrams over serial point-to-point links for wide area network (WAN) connectivity.

OS/400 Network File System Support

OS/400 Network File System (NFS) Support is a replacement for the TCP/IP File Server Support/400 (FSS/400) licensed program offering. Users who are accustomed to working with FSS/400 will notice many similarities between FSS/400 and NFS. It is important to note, however, that FSS/400 and NFS are **not** compatible with each other. At any given time, only one of these applications may be executing.

NFS support allows the AS/400 system to function as a file server on the Internet. It does this by making remote objects stored in a file system appear to be local, as if they reside in the local host.

With NFS, all systems in a network can share a single set of files. This eliminates the need for duplicate file copies on every network system. Using NFS aids in the overall administration and management of users, systems, and data. For more information about the Network File System see the book, *OS/400 Network File System Support*, SC41-5714-01.

Application Program Interfaces (APIs)

Many times an enterprise has unique interoperability requirements for its private networks. This means that the enterprise must provide its own applications to fulfill these unique requirements. On AS/400, this is accomplished with several application programming interfaces (APIs).

For helpful reference information about all of the OS/400 APIs, see the *System API Reference*, SC41-5801-03.

Sockets Interface

A sockets interface (sockets) allows you to write your own applications to supplement those that are supplied by TCP/IP. Sockets allow unrelated processes to exchange data locally and over networks. Both connection-oriented and connectionless communications are provided for TCP/IP. With this support you can write applications to the TCP, UDP, and IP protocols directly. The TCP/IP applications that run on sockets are FTP, SMTP, SNMP, LPR, LPD, BOOTP, TFTP, Routed, and REXEC. The sockets interface operates over TCP/IP or AnyNet/400.

For additional information about sockets, see the *Sockets Programming*, SC41-5422-03 book.

Send MIME Mail API

The Send MIME Mail API allows applications to use SMTP and TCP/IP to send mail to the Internet.

Pascal API

The TCP/UDP programming interface was originally developed to provide system programmers with a programming interface to the TCP or UDP protocols via a set of procedure calls from an AS/400 Pascal program. This interface is no longer documented in this publication because the AS/400 Pascal compiler is no longer available. For information about the withdrawal of support for the Pascal compiler see, *New Release Planning for V3R7*, SA41-4100.

The run-time support for the Pascal API is still included in the program product IBM TCP/IP Connectivity Utilities for AS/400 (5769-TC1). Existing applications that use the Pascal API can continue to be used. However, any new AS/400 TCP/IP applications should use the sockets API.

Important Note:

Because of changes made to AS/400 TCP/IP in V3R1, pre-V3R1 TCP/IP applications that use the Pascal API must be recompiled to function correctly in V3R1 or later.

Multiprotocol Transport Networking Architecture

The Multiprotocol Transport Networking (MPTN) architecture implementation on AS/400 allows Common Programming Interface Communications (CPI Communications), intersystem communications function (ICF), and sockets to flow over either TCP/IP, Systems Network Architecture (SNA), or Internetwork Packet Exchange (IPX). On AS/400, the MPTN architecture is put into effect as AnyNet/400 support. For more information about AnyNet/400, see "AnyNet/400" on page 19.

Transport Protocol

This layer provides the end-to-end data transfer. This layer allows communication between application programs. Example protocols that provide transport services are Transmission Control Protocol (TCP) and User Datagram Protocol (UDP).

Transmission Control Protocol (TCP)

TCP is a widely used connection-oriented protocol that enables the transfer of data from a source to a destination. TCP provides a reliable delivery of a stream of bytes in sequence. TCP takes a stream of data, breaks it into segments (a TCP header and application data), sends each one individually (using IP), and then reassembles the segments back into the original stream. If any segments are lost or damaged during the transmission, TCP detects this fact and resends the missing segments.

Most of the user application protocols, such as TELNET, FTP and SMTP, use TCP.

User Datagram Protocol (UDP)

UDP allows your application programs to send datagrams to other programs on other systems with a minimum of protocol overhead. UDP does this through the use of ports. Unlike TCP, UDP is **datagram** oriented and does not guarantee the delivery of data in sequence. Datagrams may possibly be dropped or reordered as they travel from the source to the destination. UDP can be used instead of TCP when the application does not want to incur the overhead of TCP connecting and disconnecting. It then becomes the responsibility of the application to ensure reliable data transfer and sequencing of datagrams.

The following TCP/IP applications use UDP:

- SNMP
- TFTP
- DNS

AS/400 UDP also includes multicast support, beginning with Version 4 Release 2.

TCP and UDP Ports

A port is an integer value from 1 to 65535 that is used to identify a TCP/IP application. TCP and UDP protocols use ports to identify a unique origin or destination of communication with an application. There are two unique sets of ports. One set is for TCP processing, and the other is for UDP processing. They are completely independent sets of ports and have no relationship to one another.

Well-known Ports

Commonly used protocols and applications, such as FTP and SMTP, have assigned port numbers. These assigned port numbers are called **well-known ports**. TCP and UDP port numbers 1 to 1023 are reserved for the well-known ports and should not be used by user application programs. If the user specifies one of these ports, it can affect the operation of those applications. Refer to RFC 1700, entitled, *Assigned Numbers*, for a list of well-known ports.

To see the list of ports currently defined on AS/400, perform the following steps:

1. Use the Configure TCP/IP (CFGTCP) command to get to the Configure TCP/IP menu
2. Select option 21 (Configure related tables)
3. Select option 1 (Work with service table entries)

Point-to-Point TCP/IP

Dial-up TCP/IP, known as point-to-point TCP/IP, is used to dial into remote systems, or allow remote systems to dial into AS/400 over a telephone line using a modem. Null modem or non-switched connections are also supported. The Serial Line Internet Protocol (SLIP) and the Point-to-Point Protocol (PPP) are supported on AS/400.

Internetwork Protocol

This layer provides the virtual network image of the physical network. This layer shields the higher levels from the typical network architecture below it. Internet Protocol (IP) is the most important protocol at this level. IP is a connectionless protocol that does not provide reliability, flow control, or error recovery, and does not assume reliability from the lower layers. An example of a protocol at this level is the Internet Control Message Protocol (ICMP).

Internet Protocol

The Internet Protocol (IP) provides the basic transportation rules for communication between hosts on the different networks that make up an internet. A **host** is a node on the network that has a unique internet address and an associated system name. IP is responsible for routing packets from a host on one network through a series of routers to a host on the same or another network.

In IP-based networks, information is transmitted between nodes in the form of packets. A packet includes an IP header and data. A packet or network frame can contain a complete IP datagram or a fragment of an IP datagram. A **datagram** is the basic unit of information in the TCP/IP protocol, consisting of a source address, destination address, and data. At the internet level, all addressing is host-to-host, using fixed-length addresses to identify source and destination hosts. The protocol layers above only need to know each host's internet address to make a connection. See Figure 7 and Figure 8 for illustrations of the terms packet, datagram, and segment.

If there is a transmission on the network:

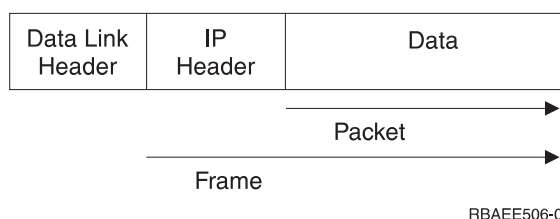


Figure 7. Packet and Frame Terminology

Before the IP is broken into pieces or after IP reassembly:

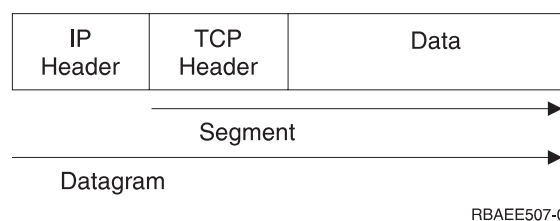


Figure 8. Segment and Datagram Terminology

After packets are received, they are passed to the IP layer. Datagrams are reassembled if necessary, and stripped of the header before being passed on to the next higher protocol layer. IP does not acknowledge receiving a datagram, nor is it responsible for retransmitting or providing flow and error control. Reliable delivery

must be ensured by a higher level protocol, such as TCP. Integral to every IP implementation is the Internet Control Message Protocol (ICMP), which is used for reporting errors, congestion reporting, and first-hop router redirection.

Internet Control Message Protocol

The Internet Control Message Protocol (ICMP) provides for error and control messages between **host systems** (peer computers in a network) and routers. Routers and host systems use ICMP to send reports of problems. ICMP also includes an echo request or reply message that is used to test whether a destination can be reached and is responding. This is commonly known as PING (Packet InterNet Groper).

Internet Group Management Protocol

The Internet Group Management Protocol (IGMP) is used by IP hosts to report their host group memberships to neighboring multicast routers. Multicast routers send Host Membership Query messages to discover which host groups have members on their attached networks. Hosts respond to a Query by generating Host Membership Reports, reporting each host group to which they belong. The multicast routers use this information to determine where multicast datagrams need to be forwarded.

Address Resolution Protocol

The Address Resolution Protocol (ARP) dynamically associates internet addresses to physical hardware addresses on a local network. ARP relies on the broadcast capabilities of the underlying media to provide this function.

AnyNet/400

AnyNet/400 is part of the AnyNet family of products. AnyNet products allow application programs written for one communications protocol to run over non-native protocols without changing (or recompiling) the application programs. The destination address determines if the request is sent over the native protocol or through the AnyNet code and on to a non-native protocol.

AnyNet/400 allows sockets, intersystem communications function (ICF), CPI Communications (CPI-C), and CICS/400 applications to run over APPC, TCP/IP, and Internetwork Packet eXchange (IPX). AnyNet/400 is based on the Multiprotocol Transport Network (MPTN) architecture, and is designed to allow any application to run over any networking protocol. You can use AnyNet/400 to:

- Access APPC using TCP/IP if your applications were developed for system network architecture (SNA) but you are using TCP/IP to connect the systems
- Access APPC using IPX if your applications were developed for SNA but you are using IPX to connect the systems
- Access sockets using SNA if your sockets applications were developed for TCP/IP but you are using SNA to connect the systems
- Access sockets using IPX if your sockets applications were developed for TCP/IP but you are using IPX to connect the systems.

AnyNet/400 is shipped with the AS/400 base operating system, OS/400.

Accessing APPC Using TCP/IP (SNA Over IP)

AnyNet/400 APPC over TCP/IP allows you to extend intersystem communications function (ICF), common programming interface for communications (CPI-C), and CICS/400 applications to TCP/IP users without adding a separate APPC network. You can also allow any OS/400, ICF, CICS/400 or CPI-C application (such as DRDA) to communicate across a TCP/IP network.

The *Communications Configuration*, SC41-5401-00 book, tells you how to configure AnyNet/400 APPC over TCP/IP.

Accessing APPC Using IPX (SNA Over IPX)

AnyNet/400 APPC over IPX allows you to extend ICF, CPI-C or CICS/400 applications to IPX users without adding a separate APPC network. You can also allow any OS/400, ICF, CICS/400 or CPI-C application (such as DRDA) to communicate across an IPX network.

Accessing Sockets Using IPX (IP Over IPX)

AnyNet/400 Sockets over IPX allows you to add Berkeley Software Distribution (BSD) Sockets applications to existing IPX networks, without adding a separate TCP/IP network. This allows OS/400 users to use most Sockets applications (such as FTP, SMTP and SNMP) across an IPX network.

The book *Internetwork Packet Exchange (IPX) Support*, SC41-5400-00, tells you how to configure and use the Novell protocol suite with OS/400.

Accessing Sockets Using SNA (IP over SNA)

AnyNet/400 Sockets over SNA allows you to add BSD Sockets applications to existing SNA networks without adding a separate TCP/IP network. This allows OS/400 users to use most sockets applications (such as FTP, SMTP and SNMP) across an SNA network.

The book *Sockets Programming*, SC41-5422-03, describes how to use both AnyNet and non-AnyNet sockets.

Chapter 2. Configuring TCP/IP

This chapter explains how to configure an AS/400 business computing system for Transmission Control Protocol/Internet Protocol (TCP/IP). If this is the first time that you have configured TCP/IP on an AS/400 system, you should read the entire chapter before performing any of the configuration tasks.

If you are unfamiliar with TCP/IP, consider reading Chapter 1. TCP/IP on AS/400. For a complete formal description of TCP/IP, you can read the Request for Comments (RFC). Or, refer to any of the TCP/IP references that are listed in "Request For Comments (RFC)" on page 602. Information about how to order an RFC is also listed within that topic.

What you need to know before you can configure TCP/IP

Before you start configuring TCP/IP, you must ensure that the *TCP/IP Connectivity Utilities for AS/400* licensed program (LP) is installed on your system. See "Installing the TCP/IP Application Programs" on page 23 for more information.

The AS/400 has many commands and menus available to help you configure TCP/IP on AS/400. Before you begin this task, take time to review the TCP/IP Administration (TCPADM) menu, Figure 9 on page 26, and the Configure TCP/IP (CGFTCP) menu, Figure 10 on page 28.

The initial displays and menus that are shown when you configure TCP/IP on your system may not contain any entries. The sample command line interface displays in this chapter may already contain data, which was entered for the purpose of example in previous configuration steps.

Performing configuration tasks on a single network or even a simple multiple network requires that you do some planning before configuring TCP/IP on any system in that network, including an AS/400. To help you get started with setting up TCP/IP, this chapter includes complete planning details and checklists.

Once you have designed a plan, follow the step-by-step process that is outlined for you in this chapter. Each step guides you through TCP/IP installation and configuration on your system, defines various terms, and describes how these terms relate to TCP/IP.

Using the Operations Navigator interface: You can configure TCP/IP through the Operations Navigator interface wizard. Information related to Operations Navigator is located in the online help. See the online help in Operations Navigator for information about the following TCP/IP functions:

- Configuring TCP/IP, including basic functions such as starting and stopping TCP/IP
- Creating a new Ethernet line
- Creating a new token-ring line
- Working with TCP/IP interfaces, including configuring a TCP/IP route
- Working with TCP/IP host tables, including configuring a TCP/IP host name and domain name
- Verifying a TCP/IP connection (PING)

Planning for TCP/IP Installation and Configuration

If you are in charge of configuring an AS/400 system for TCP/IP communications you will, in most cases, include your AS/400 system in an existing TCP/IP network. Before you are able to start configuring, you will need to collect all of the required information.

Start by contacting the person responsible for the TCP/IP network of your company, your TCP/IP network administrator. When talking to the administrator, you may want to use Table 3 on page 51 and Table 4 on page 51 as checklists to record the necessary information.

Gathering Information About your Network

After collecting the preliminary information about your network, plan the installation and configuration of TCP/IP by using the steps that are listed below:

1. **Draw a diagram of your network:** A diagram similar to that shown in Figure 32 on page 53 will help you decide how you want to attach your AS/400 system to the other systems in the network. Include other data that relates to your network, such as:
 - Line description information
 - Internet Protocol addresses and domain names
 - The number of route entries that are required

Refer to Table 3 on page 51.

2. **Identify the names of the systems in your network:** For example, do either of the following:
 - Build a local host table.
 - Identify a Domain Name System (DNS) server for maintaining host table entries.
3. **Install the appropriate hardware and software:** You must install the appropriate hardware adapters in your AS/400 system if you are going to connect to the following networks:
 - X.25 packet-switching
 - Frame relay
 - Token-ring
 - Ethernet
 - Fiber distributed data interface (FDDI)
 - Shielded twisted pair distributed data interface (SDDI)
 - Wireless local area network (LAN)
 - Synchronous or asynchronous communications line
 - Twinaxial data link support (TDLC)

You also need to make sure that the appropriate software is installed on all the systems. On the AS/400 system, the OS/400 licensed program and the TCP/IP Connectivity Utilities for AS/400 licensed program must be installed.

4. **Assign names and Internet addresses:** If you are attaching to an existing network, you need to know the Internet addresses and names used by the other systems. An existing network usually has an administrator who keeps such

information. If this person is someone other than you, have that individual provide you with the Internet address to use for your system.

Depending on the size of your network and its complexities, determine whether a host table or a DNS server is the preferred method for maintaining and updating host name and IP address associations. In this chapter, refer to “Step 6—Configuring TCP/IP Host Table Entries” on page 38. For information about configuring and using a DNS server, see page 421.

5. **Obtain X.25 network addresses:** If you plan to use TCP/IP on an X.25 private or public data network, you need to know whether you will be using a switched virtual circuit (SVC) or permanent virtual circuit (PVC).
 - To use an SVC, you need to know the network address of each remote system in the network with which you want to communicate.
 - To use a PVC, you need to know the related logical channel identifier. You can have a network address or a permanent virtual circuit, but not both, for a remote system information entry.

If a remote system is an AS/400 system, you can determine its network address by using the Display Line Description (DSPLIND) command on that remote system.
6. **Familiarize yourself with the TCP/IP Administration Menu:** The TCP/IP Administration menu (Figure 9 on page 26) provides easy access to common functions associated with administering TCP/IP.

To get to this menu, enter the GO TCPADM command from the AS/400 Main Menu.
7. **Familiarize yourself with the Configure TCP/IP Menu:** The Configure TCP/IP menu (Figure 10 on page 28) guides you through all the tasks for configuring your AS/400 system to communicate with other systems in a TCP/IP network. You can reach this menu in two ways:
 - Select option 1 on the TCPADM menu.
 - Enter the Configure TCP/IP (CFGTCP) command.

Once you have documented configuration information, you are ready to install the TCP/IP program on your AS/400 system. The information in the section that follows will help you do that. See “Installing the TCP/IP Application Programs”.

For information about TCP/IP addressing and connecting to the Internet, see “TCP/IP Addressing” on page 24. This topic discusses the methods for assigning addresses within your own network and offers an example.

Installing the TCP/IP Application Programs

Important

To determine whether the TCP/IP LP is already installed, enter GO LICPGM (Go Licensed Program) on the command line and then select Option 10 to display the installed licensed programs. If the TCP/IP Connectivity Utilities LP is not installed on your system, continue by following the instructions in this section to perform the installation.

Installing TCP/IP on your AS/400 allows you to connect an AS/400 to a network.

Perform the following steps to install TCP/IP on your AS/400 system:

1. Insert your installation media for TCP/IP into your AS/400. If your installation media is a CD-ROM, insert it into your optical device. If your installation media is a tape, insert it into your tape drive.
 2. Type **GO LICPGM** at the command prompt and press **Enter** to access the Work with Licensed Programs display.
 3. Select option 11 (Install licensed programs) on the Work with Licensed Programs display to see a list of licensed programs and optional parts of licensed programs.
 4. Type **1** in the option column next to *5769TC1 TCP/IP Connectivity Utilities for AS/400* licensed program. The Confirm Licensed Programs to Install display shows the licensed program you selected to install. Press **Enter** to confirm.
 5. Fill in the following choices on the Install Options display:
 - Installation Device
Type **OPT01**, if installing from a CD drive.
Type **TAP01**, if installing from a tape drive.
 - Objects to Install
The Objects to Install option allows you to install both programs and language objects, only programs, or only language objects.
 - Automatic IPL
The Automatic IPL option determines whether the system automatically starts when the installation process has completed successfully.
When TCP/IP successfully installs, either the Work with Licensed Programs menu or the Sign On display appears.
- Note:** For more detailed information about installing software, including objects to install and the automatic IPL option, refer to the *Software Installation*, SC41-5120-03 book.
6. Select option 50 (Display log for messages) to verify that you have installed the licensed program successfully. If an error occurs, you will see the message Work with licensed program function not complete on the bottom of the Work with Licensed Programs display.

To use TCP/IP, you must configure it after you have completed the installation. See “Configuring TCP/IP using the Command Line Interface” on page 29.

TCP/IP Addressing

TCP/IP addressing uses a unique Internet Protocol (IP) address for a specific node in a TCP/IP network. Each node on a network is known as a *host* and has a unique address.

You can assign your own addresses within your own network. To connect to the Internet, the InterNIC assigns your network addresses and domain names. For more information about how to contact the InterNIC registration services, see Table 1 on page 4.

TCP/IP addressing is comprised of two parts: The TCP/IP address itself and the subnet mask. The TCP/IP address is a 32-bit integer that contains both a network portion and a host portion. This address is usually expressed in the decimal form

xxx.xxx.xxx.xxx, where xxx is an integer between 0 and 255. The subnet mask is a bit mask that determines which bits in the TCP/IP address designate the network versus the host part of the address.

IP protocol works closely with the network and host portions of the address as well as the subnet mask.

TCP/IP addressing reflects your licensing and networking needs. The size of your network is the starting point for determining your TCP/IP addressing.

The following TCP/IP example is based on six existing networks you might have in your company:

- Network 1 needs 100 IP addresses.
- Network 2 needs 90 IP addresses.
- Network 3 needs 40 IP addresses.
- Network 4 needs 50 IP addresses.
- Network 5 needs 50 IP addresses.
- Network 6 needs 50 IP addresses.

For this example, if you are going to use class C addressing, you need to have two class C addresses. You could use Networks 1 and 2 for one class C address and Networks 3, 4, 5, and 6 for the other class C address. This arrangement allows for future expansion of the networks.

Continuing with this example, assume that the class C addresses are 192.1.1.x and 192.1.2.x. Here is how TCP/IP addressing could work in this example:

Network 1:
Network: 192.1.1.0
Subnet mask: 255.255.255.128
Usable range of IP addresses for hosts: 192.1.1.1 through
192.1.1.126.

Network 2:
Network: 192.1.1.128
Subnet mask: 255.255.255.128
Usable range of IP addresses for hosts: 192.1.1.129 through
192.1.1.254.

Network 3:
Network: 192.1.2.0
Subnet mask: 255.255.255.192
Usable range of IP addresses for hosts: 192.1.2.1 through
192.1.2.62.

Network 4:
Network: 192.1.2.64
Subnet mask: 255.255.255.192
Usable range of IP addresses for hosts: 192.1.2.65 through
192.1.2.126.

Network 5:
Network: 192.1.2.128
Subnet mask: 255.255.255.192
Usable range of IP addresses for hosts: 192.1.2.129 through
192.1.2.190.

Network 6:

Network: 192.1.2.192
Subnet mask: 255.255.255.192
Usable range of IP addresses for hosts: 192.1.2.193 through
192.1.2.254.

The domain name for the entire network could be xyz.com. Lower-level domain names could be group1.xyz.com for Networks 1 and 2 and group2.xyz.com for Networks 3, 4, 5, and 6. In this example, only the local routers know that the networks are actually different, physical networks.

Note: If you need further assistance with planning for TCP/IP addressing, refer to RFC 1219, *On the Assignment of Subnet Numbers*.

Using the TCP/IP Administration Menu

The TCP/IP Administration menu (Figure 9) is a starting point for the configuration tasks. To display the menu, enter GO TCPADM from the AS/400 Main Menu.

```
TCPADM                      TCP/IP Administration                      System:  RC
Select one of the following:

    1. Configure TCP/IP
    2. Configure TCP/IP applications
    3. Start TCP/IP
    4. End TCP/IP
    5. Start TCP/IP servers
    6. End TCP/IP servers
    7. Work with TCP/IP network status
    8. Verify TCP/IP connection
    9. Start TCP/IP FTP session
   10. Start TCP/IP TELNET session
   11. Send TCP/IP spooled file

    20. Work with TCP/IP jobs in QSYSWRK subsystem

Selection or command
====>

F3=Exit  F4=Prompt  F9=Retrieve  F12=Cancel
```

Figure 9. TCP/IP Administration Menu

Following are descriptions of the menu options.

- **Option 1. Configure TCP/IP:** Displays the Configure TCP/IP menu. Use the options on this menu to configure your local AS/400 system to communicate with other systems in a TCP/IP network.
- **Option 2. Configure TCP/IP applications:** Displays the Configure TCP/IP Applications menu. Use the options on this menu to configure the TCP/IP licensed program (5769-TC1) applications installed on your system.
- **Option 3. Start TCP/IP:** Select this option to issue the Start TCP/IP (STRTCP) command. This command initializes and activates TCP/IP processing, starts the TCP/IP interfaces, and starts the TCP/IP server jobs.
- **Option 4. End TCP/IP:** Select this option to issue the End TCP/IP (ENDTCP) command. This command is used to end all TCP/IP processing on this system.

- **Option 5. Start TCP/IP servers:** Select this option to issue the Start TCP/IP Server (STRTCPSVR) command. This command is used to start the TCP/IP application servers that are shipped with OS/400 or the TCP/IP licensed program (5769-TC1). This command starts the TCP/IP application server jobs in the QSYSWRK subsystem.
- **Option 6. End TCP/IP servers:** Select this option to issue the End TCP/IP Server (ENDTCPSVR) command. This command is used to end the TCP/IP application servers that are shipped with OS/400 or the TCP/IP licensed program (5769-TC1). This command ends the TCP/IP application server jobs in the QSYSWRK subsystem.
- **Option 7. Work with TCP/IP network status:** Select this option to issue the Work with TCP/IP Network Status (WRKTCPSTS) command. This command is used to view and manage the status information of your TCP/IP and IP over Systems Network Architecture (SNA) interfaces, routes, and connections. This command is the AS/400 version of the TCP/IP NETSTAT (Network Status) command. NETSTAT is also shipped as an AS/400 command.
- **Option 8. Verify TCP/IP connection:** Select this option to issue the Verify TCP/IP Connection (VFYTCPCNN) command. This command tests the TCP/IP connection between your system and a remote system. The VFYTCPCNN command is the AS/400 version of the TCP/IP PING (Packet InterNet Groper) command. PING is also shipped as an AS/400 command.
- **Option 9. Start TCP/IP FTP session:** Select this option to issue the Start TCP/IP FTP (STRTCPFTP) command. This command is used to start a file transfer using TCP/IP. This command is the AS/400 version of the TCP/IP FTP (File Transfer Protocol) command. FTP is also shipped as an AS/400 command.
- **Option 10. Start TCP/IP TELNET session:** Select this option to issue the Start TCP/IP TELNET (STRTCPTELN) command. This command is used to start a TELNET client session with a remote system. This command is the AS/400 version of the TCP/IP TELNET command. TELNET is also shipped as an AS/400 command.
- **Option 11. Send TCP/IP spooled file:** Select this option to issue the Send TCP/IP Spooled File (SNDTCPSPLF) command. This command sends a spooled file to be printed on a remote system. The remote system must be running TCP/IP. The SNDTCPSPLF command is the AS/400 version of the TCP/IP LPR (line printer requester) command. LPR is also shipped as an AS/400 command.
- **Option 20. Work with TCP/IP jobs in QSYSWRK subsystem:** Select this option to work with the status and performance information for the active TCP/IP jobs in the QSYSWRK subsystem. This option issues the Work with Active Jobs (WRKACTJOB) command with these parameters:
WRKACTJOB SBS(QSYSWRK) JOB(QT*)

Using the Configure TCP/IP Menu

The Configure TCP/IP menu is shown here (Figure 10 on page 28) so that you are familiar with all of the options available during configuration of the TCP/IP network. To get to this menu, select option 1 on the TCPADM menu or enter the Configure TCP/IP (CFGTCP) command.

```

CFGTCP                               Configure TCP/IP                               System:  SYSNAM890
Select one of the following:

    1. Work with TCP/IP interfaces
    2. Work with TCP/IP routes
    3. Change TCP/IP attributes
    4. Work with TCP/IP port restrictions
    5. Work with TCP/IP remote system information

    10. Work with TCP/IP host table entries
    11. Merge TCP/IP host table
    12. Change TCP/IP domain information

    20. Configure TCP/IP applications
    21. Configure related tables
    22. Configure point-to-point TCP/IP

Selection or command
====>

F3=Exit  F4=Prompt  F9=Retrieve  F12=Cancel

```

Figure 10. Configure TCP/IP Menu

Following are descriptions of the Configure TCP/IP menu options.

- **Option 1. Work with TCP/IP interfaces:** Select this option to add TCP/IP interface information to the list of current interfaces or to display, change, print, or remove TCP/IP interface information that you have already added. Select this option to start or end a TCP/IP interface.
- **Option 2. Work with TCP/IP routes:** Select this option to add route information or to display, change, print, or remove route information that you have already added.
- **Option 3. Change TCP/IP attributes:** Select this option to run the Change TCP/IP Attributes (CHGTCPA) command.
 With this option you can change User Datagram Protocol (UDP) checksum processing, IP datagram forwarding, IP time-to-live values, and other attributes that relate to the TCP/IP protocol stack.
- **Option 4. Work with TCP/IP port restrictions:** Select this option to add port restrictions or to display, remove, or print port restrictions that you have already added.
- **Option 5. Work with TCP/IP remote system information:** Select this option to add or remove X.25 data network addresses or to print the list.
- **Option 10. Work with TCP/IP host table entries:** Select this option to add host IP addresses and their associated host names to the host table or to display, change, print, rename, or remove items that you have already added.
- **Option 11. Merge TCP/IP host table:** Select this option to merge or replace a local host table by using the Merge TCP/IP Host Table (MRGTCPHT) command.
- **Option 12. Change TCP/IP domain information:** Select this option to change TCP/IP domain information.

Note: Prior to Version 4 Release 2, the Configure TCP/IP menu contained both an option 12 and an option 13. In Version 4 Release 2, the functions of options 12 and 13 were combined, and option 13 (Change Remote name

server) was removed from the menu. Option 12, formerly Change local domain and host names, was renamed to Change TCP/IP domain information.

- **Option 20. Configure TCP/IP applications:** Select this option to configure the TCP/IP applications that are installed on your system. The list of applications varies depending on whether the TCP/IP licensed program is installed on your system. If the TCP/IP licensed program is not installed on your system, you can configure **only** the following server applications:
 - Simple Network Management Protocol (SNMP)
 - Bootstrap Protocol (BOOTP) server
 - Trivial File Transfer Protocol (TFTP) server
 - Route Daemon (Routed)

If the TCP/IP licensed program is installed on your system, you can configure the following server applications:

- Simple Mail Transfer Protocol (SMTP)
- File Transfer Protocol (FTP), TELNET
- Post Office Protocol (POP) Version 3 mail server
- Line Printer Daemon (LPD)
- Remote Execution (REXEC) server
- Workstation gateway applications
- Simple Network Management Protocol (SNMP)

For information about configuring SNMP, see the *Simple Network Management Protocol (SNMP) Support* book.

- **Option 21. Configure related tables:** Select this option to configure the tables related to TCP/IP. These tables are:
 - **Protocol table**
Contains a list of protocols used in the Internet.
 - **Services table**
Contains a list of services and the specific port and protocol a service uses.
 - **Network table**
Contains a list of networks and the corresponding IP addresses for that network.
- **Option 22. Configure point-to-point TCP/IP:** Select this option to define, change, or display your TCP/IP point-to-point (SLIP) configuration.

Configuring TCP/IP using the Command Line Interface

The following steps using the command line interface will guide you through configuring TCP/IP on your AS/400 system:

1. Configuring line descriptions
2. Configuring TCP/IP interfaces
3. Configuring TCP/IP routes
4. Configuring TCP/IP attributes
5. Configuring remote system information (X.25)
6. Configuring host table entries

Note: For information about using a DNS server to manage entries, in place of host tables, refer to 421.

7. Configuring local domain and host name
8. Starting TCP/IP
9. Verifying TCP/IP connection
10. Saving the TCP/IP configuration

Important Note:

To perform the configuration steps discussed throughout this chapter, you need the special authority of *IOSYSCFG defined in your user profile.

Step 1—Configuring a Line Description

AS/400 TCP/IP supports various local area network (LAN) and wide area network (WAN) connection types: Ethernet, token-ring, SDDI and FDDI, wireless LAN, X.25 SVC, and permanent virtual circuit (PVC), Async (for SLIP), Point-to-Point (PPP) and frame relay. Refer to Appendix A. Configuring a Physical Line for TCP/IP Communication for information about how to configure an Ethernet line for TCP/IP communications.

These are the important parameters for configuring a line description:

- Line description name
- Resource name
- Local adapter address
- Ethernet standard
- Source service access point (SSAP) list.

The SSAP X'AA' required for an IEEE 802.3 Ethernet is automatically allocated if you use the *SYSGEN special value.

When TCP/IP starts an interface, the line, controller, and device descriptions are varied on automatically. If the controller and device descriptions for a line do not exist, TCP/IP creates them automatically when it attempts to start an interface using that line. This happens at TCP/IP startup time if the TCP/IP interface that is associated with the newly configured line is set to AUTOSTART *YES.

Step 2—Configuring a TCP/IP Interface

In an AS/400 system, each line that connects to a TCP/IP network must be assigned to at least one Internet address. You do this by configuring, or *adding* a TCP/IP interface. The additional interfaces are logical interfaces, not physical ones. These logical interfaces are associated with a line description.

An **interface** identifies a direct connection to a network using TCP/IP and a physical medium (communications line). You must consider the following when defining an interface:

Internet address

A 32-bit address assigned to hosts using TCP/IP. It is associated with the line description.

Subnet mask

Defines which part of an Internet address forms the subnet (subnetwork) field of an Internet address. An example of a single-network subnet mask is: 255.255.255.128.

Line description

Contains information describing a communications line that is attached to the AS/400 system, as defined previously in “Step 1—Configuring a Line Description” on page 30.

To find the names of the currently defined line descriptions, use the Work with Line Descriptions (WRKLIND) command.

Associated local interface

Allows the network to which this interface is attached appear to be part of the same network that the associated local interface is attached to. This is referred to as *transparent subnetting*.

Transparent subnetting allows TCP/IP traffic to flow between the two physical networks without defining additional routing. This is only valid for broadcast-capable networks. This also requires the Internet address for Add TCP/IP Interface (ADDTCPIFC) to be configured in the same network as the associated local interface. An additional requirement is for the subnet mask that is defined for the associated local interface.

Automatic start

Refers to whether the TCP/IP interface is started automatically whenever TCP/IP is started. The default setting is *YES. If you choose *NO, you must start the interface yourself by using the STRTCPIFC command or by selecting option 9 (Start) on the Work with TCP/IP Interfaces display, as shown in Figure 12 on page 32.

To add a TCP/IP interface, do the following:

1. Enter GO TCPADM to get the TCP/IP Administration menu.
2. Select option 1 to get to the Configure TCP/IP menu.
3. Select option 1 on the Configure TCP/IP menu.

The Work with TCP/IP Interfaces display is shown in Figure 12 on page 32.

4. Type option 1 (Add) at the input-capable top list entry on this display to go to the Add TCP/IP Interfaces (ADDTCPIFC) display, as shown in Figure 11 on page 32.

(You can go directly to this display by typing ADDTCPIFC command on any command line and pressing F4.)

AS/400 TCP/IP supports multihoming, which allows you to specify multiple interfaces for each line description. For example, multihoming can be used to assign multiple Internet addresses to a single physical line description. See “Multihoming Function” on page 79 for further information.

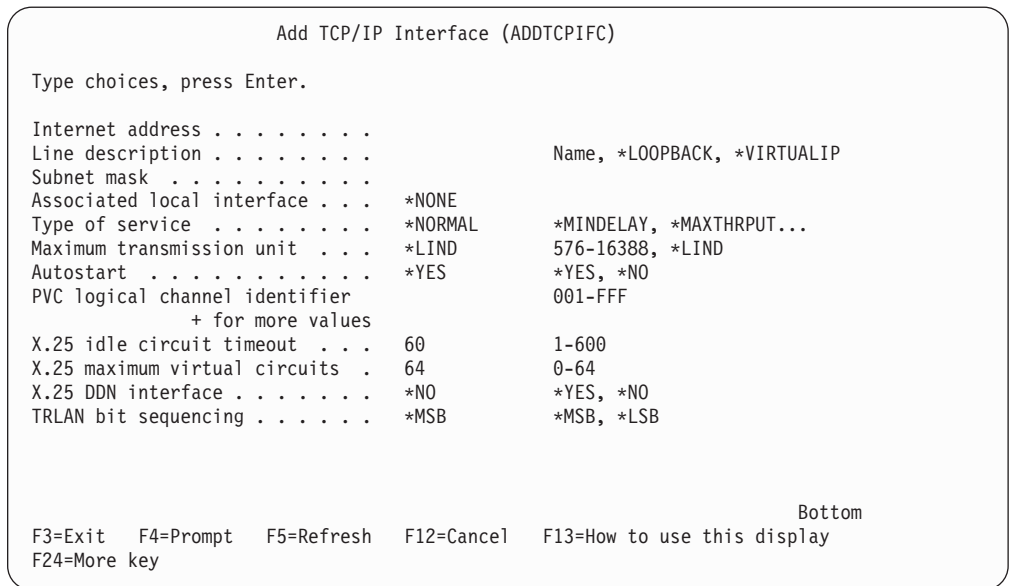


Figure 11. Add TCP/IP Interfaces Display

When you are finished adding entries, the Work with TCP/IP Interfaces display looks like Figure 12.

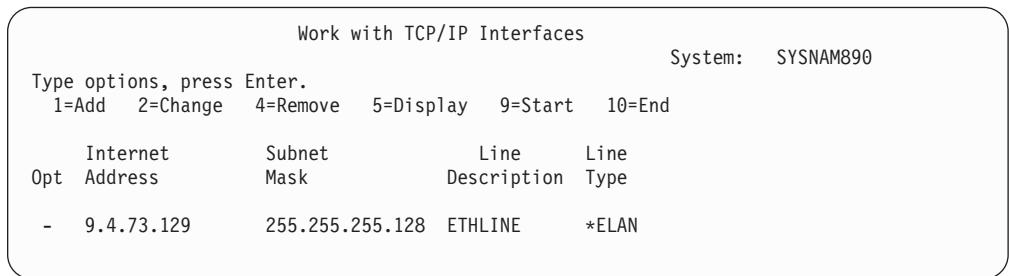


Figure 12. Work with TCP/IP Interfaces Display

Note: Any change to the TCP/IP interfaces configuration, except for the automatic start parameter, takes effect immediately.

Step 3—Configuring TCP/IP Routes

Do you need to add routes at all?

If you have several individual networks to which the AS/400 is not directly attached, you must add routing entries to allow the AS/400 to reach these remote networks.

If your AS/400 is attached to a single network, if there are no IP routers in your network, you do not need to add routes.

To reach remote networks, at least one routing entry is required. If no routing entries are manually added, your AS/400 cannot reach systems that are not on the

same network that the AS/400 is attached to. You must also add routing entries to allow TCP/IP clients that are attempting to reach your AS/400 system from a remote network to function correctly.

For example, suppose that someone using a PC is using the TELNET application to start a remote terminal session on your AS/400 system. The application on the PC must know the route or path to reach the AS/400 system. Your AS/400 system must also be able to determine the route back to the PC. If the PC and your AS/400 system are not on the same network, a routing entry must exist on the PC and on AS/400.

Note: You should plan to have the routing table defined so that there is always an entry for at least one default route (*DFTRROUTE). If there is no match on any other entry in the routing table, data is sent to the IP router specified by the first available default route entry. The only exception to this is if you intend to dial out over a SLIP link to an Internet Service Provider or another remote host. See “Using SLIP with an Asynchronous Line Description” on page 126 for details.

Before adding routing entries, familiarize yourself with the following terms:

Route destination

The network ID portion of an Internet address. The network ID portion is composed of the first byte, the first two bytes, or the first three bytes of the Internet address (depending on the network class). The remaining bytes define the host ID portion of the Internet address.

If subnetting is used, route destination includes the subnet part as well. In other words, **the route destination equals the address of a TCP/IP network to be reached.**

Subnet mask

A bit mask that defines which part of an Internet address forms the network and the subnetwork.

The technique known as **subnet addressing, subnet routing, or subnetting** allows a single network ID to be used on multiple physical networks. This technique lets you define separate routes to different sets of Internet addresses within a specific network.

For more information about subnet masks and subnetworks, refer to “Subnetworks and Subnet Masks” on page 6.

Next hop

The Internet address of the first system in the route between your system and the destination network. The next hop value is always an Internet address. Next hops need to be hosts on a directly connected TCP/IP network defined by the TCP/IP interfaces.

Maximum Transmission Unit (MTU) size

The maximum size (in bytes) of IP datagrams sent on a route. If you specify *IFC, the size is calculated for you based on values found in the AS/400 line description. The maximum size specified for a particular route must not be larger than the smallest MTU supported by any router or bridge in that route. If you specify a larger size, some datagrams may be lost.

In addition, the MTU specified for a particular route should not be larger than the smallest MTU supported by any system used as an IP router for that route. If you specify a larger size, performance may degrade as systems attempt to divide the IP datagrams into smaller fragments.

For additional information about setting the MTU, see Appendix A. Configuring a Physical Line for TCP/IP Communication.

Preferred binding interface

The preferred binding interface allows administrators to choose which of the TCP/IP interfaces that they prefer the route to be bound to or on. This provides the administrator with more flexibility to route traffic over a specific interface. The interface is preferred because the route is bound to the indicated interface if the interface is active. If the indicated interface is not active, then a best-match-first algorithm is used in determining which interface the route is bound.

In Figure 13, a preferred binding interface of *NONE has been defined. By using this definition, the user allows the TCP/IP protocol stack to choose an interface to bind this route to, using a best-match-first algorithm.

Adding TCP/IP routes

You must define routes for any TCP/IP network, including subnetworks, with which you want to communicate. You do not need to define routes for the TCP/IP network that your AS/400 system is directly attached to when you are using an AS/400 adapter.

Manual configuration of the routes that tell TCP/IP how to reach the local networks is not required. AS/400 TCP/IP generates these routes automatically from the configuration information for the interfaces every time TCP/IP is started. In other words, the direct route to the network, which has an interface attached, is automatically created when you add the interface.

To display all routing entries, including direct routes, use the Network Status (NETSTAT) command after starting TCP/IP.

To add a route, type option 2 on the Configure TCP/IP menu. The Work with TCP/IP Routes display (Figure 13) is shown.

Work with TCP/IP Routes				System: SYSNAM890
Type options, press Enter.				
1=Add 2=Change 4=Remove 5=Display				
Opt	Route Destination	Subnet Mask	Next Hop	Preferred Interface
-	*DFTRROUTE	*NONE	9.4.73.193	*NONE

Figure 13. Work with TCP/IP Routes Display

Type option 1 (Add) at the input-capable top list entry on that display to go to the Add TCP/IP Route (ADDTCPRTE) display, as shown in Figure 14 on page 35.

(To go directly to this display, type the ADDTCPRTE command on any command line and press F4.)

```

Add TCP/IP Route (ADDTCPRTE)

Type choices, press Enter.

Route destination . . . . . > '9.4.6.128'
Subnet mask . . . . . > '255.255.255.128'
Type of service . . . . . *NORMAL *MINDELAY, *MAXTHRPUT...
Next hop . . . . . > '9.4.73.193'
Preferred binding interface . . *NONE
Maximum transmission unit . . . 576          576-16388, *IFC
Route metric . . . . . 1          1-16
Route redistribution . . . . . *NO      *NO, *YES
Duplicate route priority . . . . 5          1-10

Bottom
F3=Exit  F4=Prompt  F5=Refresh  F12=Cancel  F13=How to use this display
F24=More keys

```

Figure 14. Add TCP/IP Routes Display

Note: Any changes that you make to the routing information take effect immediately.

```

Work with TCP/IP Routes

Type options, press Enter.
  1=Add  2=Change  4=Remove  5=Display

Opt  Route Destination      Subnet Mask      Next Hop      Preferred Interface
--  -----
-   *DFTRROUTE             *NONE           9.4.73.193    *NONE
-   9.4.6.128             255.255.255.128 9.4.73.193
-

```

Figure 15. Work with TCP/IP Routes Display

Multiple Default Routes

Default routes are used to route data that is being addressed to a remote destination and that does not have a specific route defined. Default routes are based on the availability of the next hop router and the type of service (TOS). If no specific TOS is requested, the first available default route with TOS of *NORMAL is used.

If a default route is not defined, only the networks explicitly defined by any non-default routes appear as though TCP/IP can reach them, and datagrams bound for any undefined networks are not sent.

Note: A default route cannot have a subnetwork; therefore, you must leave the subnet mask at the default value of *NONE.

Consult “Multiple Routes” on page 84 for further information about multiple default routes and the type of service (TOS) parameter.

Step 4—Configuring TCP/IP attributes

To configure the TCP/IP attributes, type option 3 on the Configure TCP/IP menu. The Change TCP/IP Attributes (CHGTCPA) display is shown (Figure 16).

```
Change TCP/IP Attributes (CHGTCPA)

Type choices, press Enter.

TCP keep alive . . . . . 120          1-40320, *SAME, *DFT
TCP urgent pointer . . . . . *BSD      *SAME, *BSD, *RFC
TCP receive buffer size . . . . . 8192  512-8388608, *SAME, *DFT
TCP send buffer size . . . . . 8192    512-8388608, *SAME, *DFT
UDP checksum . . . . . *YES          *SAME, *YES, *NO
IP datagram forwarding . . . . . *YES  *SAME, *YES, *NO
IP source routing . . . . . *YES      *SAME, *YES, *NO
IP reassembly time-out . . . . . 10    5-120, *SAME, *DFT
IP time to live . . . . . 64          1-255, *SAME, *DFT
ARP cache timeout . . . . . 5         1-1440, *SAME, *DFT
Log protocol errors . . . . . *YES    *SAME, *YES, *NO
```

Figure 16. Change TCP/IP Attributes Display

For information about the various parameters for this command, see the online help. In this step only the IP Datagram Forwarding (IPDTGFWD) parameter is discussed.

IP Datagram Forwarding

Specifies whether your system should forward datagrams destined for other networks. The default value is *NO.

Step 5—Configuring TCP/IP Remote System Information (X.25)

Note: If you are not using use X.25, then proceed to “Step 6—Configuring TCP/IP Host Table Entries” on page 38.

If you use an X.25 connection to reach TCP/IP hosts with a public or private packet switched data network (PSDN), you need to add remote system information for each remote TCP/IP host. You must define the X.25 network address of each system if you use a switched virtual circuit (SVC). If a permanent virtual circuit (PVC) is set up by the network connecting your system with your remote TCP/IP partner, you need to know the local logical channel identifier of this PVC.

Adding Remote System Information (X.25)

To add an X.25 remote system address, type option 5 on the Configure TCP/IP menu. The Work with the TCP/IP Remote System Information display appears, as shown in Figure 17 on page 37.


```

Work with TCP/IP Remote System Information
System: SYSNAM890
Type options, press Enter.
  1=Add  4=Remove  5=Display

Opt      Internet      Network      PVC      Reverse
  Address      Address
-----
(No remote system information)

```

Figure 17. Work with Remote System (X.25) Information

Type option 1 (Add) at the input-capable top list entry to go to the Add TCP/IP Remote System (ADDTCPRSI) display, as shown in Figure 18.

```

Add TCP/IP Remote System (ADDTCPRSI)
Type choices, press Enter.
Internet address . . . . . > '9.4.73.66'
Network address . . . . . > 40030002
PVC logical channel identifier      001-FFF
X.25 reverse charge . . . . . *NONE      *NONE, *REQUEST, *ACCEPT

Additional Parameters
Default packet size:
  Transmit packet size . . . . . *LIND      *LIND, 64, 128, 256, 512...
  Receive packet size . . . . . *LIND      *LIND, *TRANSMIT, 64, 128...
Default window size:
  Transmit window size . . . . . *LIND      1-15, *LIND
  Receive window size . . . . . *LIND      1-15, *LIND, *TRANSMIT

Bottom
F3=Exit  F4=Prompt  F5=Refresh  F12=Cancel  F13=How to use this display
F24=More keys

```

Figure 18. Add Remote System (X.25) Information

The network controller used by AS/400 TCP/IP does not allow you to specify X.25 user facilities. However, some of the values usually configured on a controller, using the ADDTCPRSI command, allow you to configure each X.25 remote system. These values include reverse charging, packet sizes, and window sizes.

Use the following CL command is used to enter the information as shown in the display above:

```

ADDTCPRSI INTNETADR('9.4.73.66')
NETADR(40030002)

```

Notes:

1. Specifying remote system information for an X.25 DDN interface causes that information to be used instead of the DDN conversion algorithm. The DDN conversion algorithm is used only for a connection with DDN specified as *YES when you try to connect to a host that is not defined in the remote system

information. If DDN is specified as *YES on the X.25 connection, you should not specify remote system information for that interface or its associated DDN network systems.

2. A routing error occurs when both of the following are true:
 - The remote system information associated with the Internet address is an extended data terminal equipment (DTE) address.
 - The configured X.25 interface's line does not support X.25 extended addressing.

Note: Any changes that you make to the remote system information take effect immediately.

Step 6—Configuring TCP/IP Host Table Entries

Each computer system in your network is called a **host**. The host table allows you to associate a host name to an Internet address. This step gives instruction for configuring a host table and host table entries. However, you should determine early in the configuration planning if a host table or a Domain Name System (DNS) server is the best option for you in managing host name and IP address translations.

Whenever possible, a DNS server should be used as a replacement for, or in addition to, the local host table. The DNS server is a single source for host names, which is one reason that it is often preferred over host tables, especially for larger networks. For more information about DNS server support for your AS/400 system, see "Chapter 18. AS/400 Domain Name System (DNS)" on page 421.

The local host table on your AS/400 system contains a list of the Internet addresses and related host names for your network. Host tables map Internet addresses to TCP/IP host names. Host tables allow users to use an easily remembered name for a system in a network without having to remember the Internet address.

To configure the mapping of host names to Internet addresses, you can use three different options on the Configure TCP/IP menu. You can use only one or a combination of all three to obtain the host name processing you need for your network. The three options on the Configure TCP/IP menu related to Internet address mappings are:

1. Option 10 (Work with TCP/IP host table entries) to create your own host table. The Work with Host Table Entries display is shown in Figure 19 on page 39.
2. Option 11 (Merge TCP/IP host table) to merge or convert a host table sent from another system.
For more information about merging and converting host tables, see "Merging TCP/IP Host Tables" on page 75.
3. Option 12 (Change TCP/IP domain information) to call the following new command, CHGTCPDMN.

Note: You can start TCP/IP client functions, such as FTP, by specifying the Internet address directly without using the host table.

For more information about managing host tables, including host file formats, and merging host tables, see "Managing TCP/IP Host Tables" on page 73.

Adding an Entry to the Host Table

The Add TCP/IP Host Table Entry display provides fields for an Internet address, associated host name, and an optional text description.

To add an entry to your local host table, type option 10 on the Configure TCP/IP menu. The Work with TCP/IP Host Table Entries display is shown in Figure 19.

```
Work with TCP/IP Host Table Entries                               System: SYSNAM890
Type options, press Enter.
 1=Add  2=Change  4=Remove  5=Display  7=Rename

Opt      Internet      Host
        Address      Name
-----
-        127.0.0.1    LOOPBACK
-                                     LOCALHOST
```

Figure 19. Work with TCP/IP Host Table Entries Display

Note: Just as AS/400 TCP/IP automatically creates a LOOPBACK interface, it also automatically adds an entry to your local host table to associate the IP address 127.0.0.1 with the host names LOOPBACK and LOCALHOST. Type option 1 (Add) at the input-capable top list entry to show the Add TCP/IP Host Table Entry display.

Work with TCP/IP Host Table Display

Figure 20 and Figure 21 on page 40 show how the host table looks after you enter all hosts explicitly known.

```
Work with TCP/IP Host Table Entries                               System: SYSNAM890
Type options, press Enter.
 1=Add  2=Change  4=Remove  5=Display  7=Rename

Opt      Internet      Host
        Address      Name
-----
-        9.4.6.129    ROUTER2
-        9.4.6.134    HPUX
-        9.4.6.138    SPARKY
-        9.4.6.252    MVAX
-        9.4.73.65     XSYSNAM890
-        9.4.73.66     XSYSNAM456
-        9.4.73.129    ESYSNAM890
-        9.4.73.130    ESYSNAMRS
-        9.4.73.193    ROUTER1
-        9.4.73.198    SYSNAMRS
-        9.4.73.206    ITALY
-        9.4.73.207    HOLLAND
-        9.4.73.208    ENGLAND

More...
```

Figure 20. Work with Host Table Entries, Display 1 of 2

```

Work with TCP/IP Host Table Entries
System:  SYSNAM890
Type options, press Enter.
  1=Add  2=Change  4=Remove  5=Display  7=Rename

  Internet      Host
Opt  Address      Name
--  -
-   9.4.73.211    BERN
-   9.4.73.212    SYSNAM890
-   9.4.73.214    MACIAN
-   9.4.191.76    DNS
-   127.0.0.1     LOOPBACK
-                   LOCALHOST

```

Figure 21. Work with Host Table Entries, Display 2 of 2

The AS/400 TCP/IP host table is shipped with the LOOPBACK entry. The LOOPBACK entry has an Internet address of 127.0.0.1 and two host names: LOOPBACK and LOCALHOST.

The 127.0.0.1 Internet address can be changed (CHGTCPHTE) and a different one can be added (ADDTCPHTE). The local table command processing programs ensure that any LOOPBACK host name added or changed in the host table is in the range of 127.0.0.1 to 127.255.255.254. Multiple loopback host table entries are allowed in the AS/400 host table.

You may alter the LOOPBACK host name or add additional host names using the (CHGTCPHTE) command.

If the LOOPBACK or LOCALHOST name is changed or removed from the host table, the name is not valid, unless the domain name server has a LOOPBACK entry that specifies this value as a host name.

You can define up to four names for each Internet address. If the TCP/IP host is in your local domain, then it is not necessary to qualify the host with the domain name. As long as a TCP/IP host is in your local domain, you need only to enter the host name with the host table entry.

However, if you would like to add TCP/IP hosts that are outside of your local domain, you need to add these TCP/IP hosts as fully qualified. The fully qualified host name of SYSNAMEND.ENDICOTT.IBM.COM shows this as an example in Figure 22 on page 41.

```

Work with TCP/IP Host Table Entries
System:  SYSNAM890
Type options, press Enter.
  1=Add  2=Change  4=Remove  5=Display  7=Rename

  Internet      Host
Opt  Address      Name
--  -
  9.4.73.211    BERN
  9.4.73.212    SYSNAM890
  9.4.73.214    MACIAN
  9.4.191.76    DNS
  9.125.87.127  SYSNAMEND.ENDICOTT.IBM.COM
  127.0.0.1     LOOPBACK
  -            LOCALHOST

```

Figure 22. Example of a Fully Qualified Host Table Entry

Additional host names are useful as alternative nicknames. See the examples in Figure 23.

Host names need not be unique. When searching the host table with a duplicate host name, the result is random. However, IP addresses have to be unique. The uniqueness of the IP address is enforced at the time you try to add a new entry to the host table.

Note: An IP address cannot be used as a host name.

```

Work with TCP/IP Host Table Entries
System:  SYSNAM890
Type options, press Enter.
  1=Add  2=Change  4=Remove  5=Display  7=Rename

  Internet      Host
Opt  Address      Name
--  -
  9.4.73.211    BERN
  9.4.73.212    SYSNAM890
  -            M03
  -            F25
  -            MYSYSTEM
  9.4.73.214    MACIAN
  9.4.191.76    DNS
  9.4.73.198    SYSNAMRS

```

Figure 23. Multiple Host Names

To remove one of the additional host names, select option 2 to change the selected host table entry. Type *BLANK over the host name to remove it.

Note: The fully qualified host name is used when sending mail between two TCP/IP hosts.

Notice in the example that the name of AS/400 system SYSNAM890 is in the host table too. There are several reasons to put your host name in the host table:

- You may want to use your host name when using FTP, TELNET, or PING to test your own system's configuration.

- Simple Mail Transfer Protocol (SMTP) requires your host name to be in the host table or on a domain name server.
- You may want to use your host table on other systems in the network. Your host name must be in the host table on those systems so they can refer to your system by name.
- Applications written to use host table lookup routines may require this information.

When you are finished working with the host table, press F3 (Exit) or F12 (Cancel).

AnyNet/400: APPC over TCP/IP

Advanced program-to-program communication (APPC) over TCP/IP support allows Common Programming Interface (CPI) Communications or Intersystem Communications Function (ICF) applications to run over TCP/IP with no changes. To use the APPC over TCP/IP support, the logical unit (LU) name or the remote location that your application uses must be mapped to an Internet address. For APPC over TCP/IP support, the host table is configured to map Internet addresses to LU names. To do this, you can update the TCP/IP host table using the configuration menus. The format for the host name is:

```
LUNAME.NETID.SNA.IBM.COM
```

Step 7—Configuring the Local Domain and Host Name

Within TCP/IP, the primary name associated with your system (your system can have more than one name) is called your **local domain and host name**. The combination of the local domain and host name forms a fully-qualified host name. The fully qualified host name is the name by which your system is known and identified in the TCP/IP domain. The local domain name is also used by sockets to help in host name resolution at the Domain Name System (DNS) server. The Post Office Protocol (POP) and Simple Mail Transfer Protocol (SMTP) mail servers require that the local domain and host name be configured. It is used, but not required, by line printer requester (LPR), File Transfer Protocol (FTP), and Simple Network Management Protocol (SNMP).

A domain name consists of labels that are separated by periods, for example, SYSNAM890.ROCHESTER.IBM.COM. For hosts, the first label in a domain name is the name of a host that belongs in the domain identified by the other labels. In this example, host SYSNAM890 belongs to the domain ROCHESTER.IBM.COM. SYSNAM890.ROCHESTER.IBM.COM is known as the host's fully qualified domain name.

To define a local domain name and a host name, use option 12 (Change TCP/IP domain information) from the Configure TCP/IP menu (Figure 10 on page 28). Refer to "Domain and Host Name" on page 9 for a more detailed discussion of domain names.

You may need to configure the local domain name if you use a DNS server that requires a fully qualified host name to resolve an Internet address. "Chapter 18. AS/400 Domain Name System (DNS)" on page 421 provides more information on how to do that.

The AS/400 TCP/IP applications concatenate the local domain name to the host name if a period is not used at the end of the domain name. See "Concatenating the Domain Name to the Host Name" on page 437 for an example.

To change the local domain name, type option 12 on the Configure TCP/IP menu. The Change TCP/IP domain information display is shown in Figure 24.

```
Change TCP/IP Domain (CHGTCPDMN)

Type choices, press Enter.

Host name . . . . . SYSNAM890

Domain name . . . . . SYSNAM123.IBM.COM

Host name search priority . . . *LOCAL      *REMOTE, *LOCAL, *SAME
Domain name server:
Internet address . . . . . '9.4.73.129'
```

Figure 24. Change TCP/IP Domain Information (CHGTCPDMN)

Notes:

1. Changes that you make using the Change TCP/IP domain information (CHGTCPDMN) command take effect immediately.
2. The local domain name is used by many applications including PING. PING appends the local domain to a host name **if** a domain is not specified or if a period (.) does not appear at the end of the specified host name.

Domain Name System (DNS) Server

The conversion from host name to Internet address can be performed by using the host table on the local system or by defining a Domain Name System server, or DNS server.

In large networks with large host tables, it is more convenient to have DNS servers than to have a complete copy of the host table on every host in the network.

A DNS server maintains the host table for an entire TCP/IP domain. This prevents each single host from having to maintain its own local host table.

You can configure your AS/400 system to use both a DNS server and your local host table, but they are not mutually exclusive. You can also specify whether the domain name server or your local host table is searched first.

For more information about how the Domain Name System works and how to configure a DNS server, see “Chapter 18. AS/400 Domain Name System (DNS)” on page 421.

Step 8—Starting TCP/IP and TCP/IP Servers

Before any TCP/IP services are available on the AS/400 system, TCP/IP processing must be initialized and activated. To start TCP/IP, you have two options:

1. Select option 3 from the TCP/IP Administration menu (GO TCPADM),
2. Enter the Start TCP/IP (STRTCP) command.

The STRTCP command initializes and activates TCP/IP processing, starts the TCP/IP interfaces, and starts the TCP/IP server jobs. Only TCP/IP interfaces with AUTOSTART *YES are started at STRTCP time. Allow a few moments for TCP/IP to start, and then check to see if the QTCPIP job has started.

Option 20 of the TCP/IP Administration menu allows you to display the jobs related with TCP/IP. You can also use the following command:

```
WRKACTJOB SBS(QSYSWRK) JOB(QT*)
```

The job QTCPIP should be displayed.

Messages indicating that TCP/IP has been started are sent to the QTCP and QSYSOPR message queues. To check for the successful start of TCP/IP, enter either of these commands:

```
DSPMSG QSYSOPR
DSPMSG QTCP
```

Figure 25 contains a sample of the messages that are issued.

```
STRTCP issued by job 007138/DJONES/DSP02.
QTCPIP job started.
127.0.0.1 interface started.
QTCPIP job starting 9.5.5.162 interface.
127.0.0.2 interface started.
SNMP Server starting.
TELNET Server starting
FTP Server starting
SMTP Server starting
POP Server starting
LPD Server starting
9.5.5.162 interface started.
STRTCP completed successfully.
```

Figure 25. Sample Messages from STRTCP with All Applications Autostarted

If the QTCPIP job does not start, look for spooled job logs. Generally, the user for these job logs is QTCP. Use the Work with Spooled Files (WRKSPLF) command and specify QTCP for the user (WRKSPLF QTCP) to find the logs.

Application Servers: The TCP/IP application server jobs run under subsystem QSYSWRK. Several types of TCP/IP server jobs run in the QSYSWRK subsystem. They are the server jobs for TELNET, POP, FTP, SMTP, LPD, BOOTP, TFTP, RouteD, REXEC, and SNMP.

The STRTCP command starts the server jobs for an application if the automatic start attribute for that server is equal to *YES. To change the autostart attribute for an application, do either of the following:

- Select option 2 from the TCP/IP Administration menu
- Option 20 from the TCP/IP Configuration menu

Using the Start TCP/IP Server (STRTCPSVR) command starts the servers individually or together. You can monitor the jobs with option 20 (Work with TCP/IP jobs in QSYSWRK subsystem) from the TCP/IP Administration menu.

If you want TCP/IP processing and any related TCP/IP servers to start automatically at the initial program load (IPL), add STRTCP to the QSTRUP CL program.

Note: If they are installed, the Client Access host servers are automatically started when TCP/IP is started.

Changing the IPL Start-Up Program The autostart job in the controlling subsystem transfers control to the program specified in the system value QSTRUPPGM. You can tailor this program. For instructions on how to create your own IPL start-up program, see the *OS/400 Work Management* book located in the AS/400 Online Library at the following URL address: <http://www.as400.ibm.com/infocenter>.

REMINDER: Host Table Conversion: If you had a pre-V3R1M0 version of TCP/IP installed on your AS/400 and you had a local host table with more than 75 entries, use one of the host table configuration commands, such as CHGTCPHTE or MRGTCPHT before you run the STRTCP command. Using the host table configuration commands converts pre-V3R1M0 host tables to the new format without affecting the performance of the STRTCP command processing.

TCP/IP Jobs

Jobs started by the Start TCP/IP (STRTCP) command are listed in Table 2.

Table 2. Jobs Used by TCP/IP

Job Name	Description
QAPPCTCP	APPC over TCP/IP applications
QTBOOTP	BOOTP server
QTCPIP	Main TCP/IP job
QTFTPxxxxx	FTP server (there may be several)
QTGTELNETS	TELNET server (there may be several)
QTRTDxxxxx	RouteD server
QTRXCxxxx	REXEC server (there may be several)
QTSMTPLNT	SMTP client
QTSMTPSRVR	SMTP server
QTSMTPBRCCL	SMTP bridge client
QTSMTPBRSR	SMTP bridge server
QTTFTxxxxx	TFTP server (there may be several)
QTMSNMP	SNMP server
QTMSNMPCV	SNMP server
QSNMPSA	SNMP server
QTLPDxxxxx	LPD server (there may be several)
QTPOxxxxxx	POP server (there may be several)
QTPPANSxxx	Dial-in (*ANS) support (PPP)
QTPPDIALxx	Dial-out (*DIAL) support (PPP)
ADMIN and DEFAULT	ICS (HTTP) server
QTSWGxxxxx	Workstation gateway (there may be several)

Table 2. Jobs Used by TCP/IP (continued)

Job Name	Description
Note:	
1. There may be other jobs running in the QSYSWRK subsystem that have nothing to do with TCP/IP.	
2. The TCP/IP jobs in QSYSWRK run under the QTCP user profile, with two exceptions: the TFTP server runs under the QTFTP profile, and the workstation gateway server runs under the QTMTWGS profile.	
3. To use APPC over TCP/IP applications, you must set the network attribute Allow AnyNet (ALWANYNET) to *YES.	

End TCP/IP (ENDTCP):

ATTENTION!

No confirmation display appears when you enter ENDTCP is entered. Therefore, you must use the ENDTCP command carefully. The default for the ENDTCP command is to immediately end all TCP/IP processing on the AS/400 system that you are working on.

Use the End TCP/IP (ENDTCP) command to end all TCP/IP processing.

The command can be issued from the command line or by using option 4 on the TCP/IP Administration menu. To display this menu, enter GO TCPADM on the command line.

Step 9—Verifying the TCP/IP Connection

To verify the TCP/IP connection from your AS/400 system to the network, use the PING (VFYTCPCNN) function.

1. To test the TCP/IP code without sending anything out of the token-ring adapter, specify the special host name LOOPBACK as follows:

```
PING LOOPBACK
```

2. To test the TCP/IP code, token-ring adapter, and token-ring connection, specify the Internet address of the local adapter, as defined in the host table, as follows:

```
PING RMTSYS(*INTNETADR)
INTNETADR('9.4.73.212')
```

Or you may enter:

```
PING RMTSYS(SYSNAM890)
```

This command sends data out onto the token-ring line, which the local adapter receives again as if the data is from the TCP/IP network.

Figure 26 on page 47 shows the results from a successful connection verification.

```

> ping '9.4.73.212'
Verifying connection to host system 9.4.73.212.
PING request 1 from 9.4.73.212 took 24 ms. 256 bytes. TTL 64.
PING request 2 from 9.4.73.212 took 11 ms. 256 bytes. TTL 64.
PING request 3 from 9.4.73.212 took 31 ms. 256 bytes. TTL 64.
PING request 4 from 9.4.73.212 took 11 ms. 256 bytes. TTL 64.
PING request 5 from 9.4.73.212 took 12 ms. 256 bytes. TTL 64.
Round-trip (in milliseconds) min/avg/max = 11/17/31
Connection verification statistics: 5 of 5 successful (100 %).

```

Figure 26. Successful PING Messages

3. If the PING operation is successful, you should see messages similar to those in Figure 26.

If the PING operation is unsuccessful, you should see messages similar to those in Figure 27.

If you receive an unsuccessful PING message, check your configuration steps. Also check that the configuration at the remote system is correct and that the remote system is not powered down. For additional information about identifying the cause for an unsuccessful connection verification, see 437.

```

> ping '9.4.73.198'
Verifying connection to host system 9.4.73.198.
No response from host within 1 seconds for connection verification 1.
No response from host within 1 seconds for connection verification 2.
No response from host within 1 seconds for connection verification 3.
No response from host within 1 seconds for connection verification 4.
No response from host within 1 seconds for connection verification 5.
Connection verification statistics: 0 of 5 successful (0 %).

```

Bottom

Figure 27. Unsuccessful PING Messages

Note: A datagram sent by TCP or UDP to a system with the name LOOPBACK does not actually leave the system. The IP layer, instead, returns the datagram to the TCP or UDP layer from which it came. The other layers then treat the datagram as a normal incoming datagram. The LOOPBACK host name can be used with any TCP/IP command requiring a system name, such as PING or FTP (or any TCP or UDP application including user-written applications). Using the LOOPBACK default host name provides an ability to test TCP/IP applications without actually connecting to a physical network.

The AS/400 system defines LOOPBACK as the default host name by automatically creating an entry in the local host table.

Verifying Additional TCP/IP Connections

Once TCP/IP is configured on the AS/400 system, and the initial connection is verified, you will probably want to add more systems to your network. When you connect additional systems to your network, you also need to verify their TCP/IP connection. The examples in the following paragraphs show you how to verify a remote TCP/IP connection.

Use the system menus or the Verify TCP/IP Connection (VFYTCPCNN or PING) command to verify your system's ability to communicate with a remote system using TCP/IP.

Note: PING uses the Internet Control Message Protocol (ICMP) to send data to a host's Internet address and waits for a response. The user command to perform this verification is called PING (Packet InterNet Groper) on non-AS/400 systems. On an AS/400 system, use either the PING command or the VFYTCPCNN command.

To verify TCP/IP connections, perform the three steps below in the order in which they are listed:

1. Type VFYTCPCNN and then press F4.
The display for the VFYTCPCNN command appears (Figure 28).
2. Type the name of a remote system as defined in your host table or as defined by your domain name server.
If you prefer to use an Internet address, type the address enclosed in apostrophes. You can also type *INTERNETADR to be prompted for the Internet address.
3. Press F10 to view or change the additional parameters.
As you can see in Figure 29 on page 49, the system defaults are to send five packets of 256 bytes each and to wait 1 second for a response on each packet.

Verify TCP/IP Connection (VFYTCPCNN)

Type choices, press Enter.

Remote system _____

Figure 28. Verify TCP/IP Connection

```

Verify TCP/IP Connection (PING)

Type choices, press Enter.

Remote system . . . . . sysnam36.sysnam123.ibm.com
_____
_____
_____
Remote internet address . . . . . _____

Additional Parameters

Message mode:
  Response message detail . . . *VERBOSE      *VERBOSE, *QUIET
  Summary, if response errors . *COMP      *COMP, *ESCAPE
  Packet length (in bytes) . . . 256          8-512
  Number of packets . . . . . 5              1-999
  Wait time (in seconds) . . . . 1           1-120
  Local internet address . . . . *ANY _____
  Type of service . . . . . *NORMAL        *MINDELAY, *MAXTHRPUT...
  IP time to live . . . . . *DFT          1-255, *DFT

More...
F3=Exit  F4=Prompt  F5=Refresh  F12=Cancel  F13=How to use this display
F24=More keys

```

Figure 29. Verify TCP/IP Connection, Additional Parameters

Verifying TCP/IP Connections with Host Name—Example

In this example, sending five packets of 256 bytes each verifies the connection to the remote system SYSNAM36. The local system waits 1 second for a response to each packet that is sent.

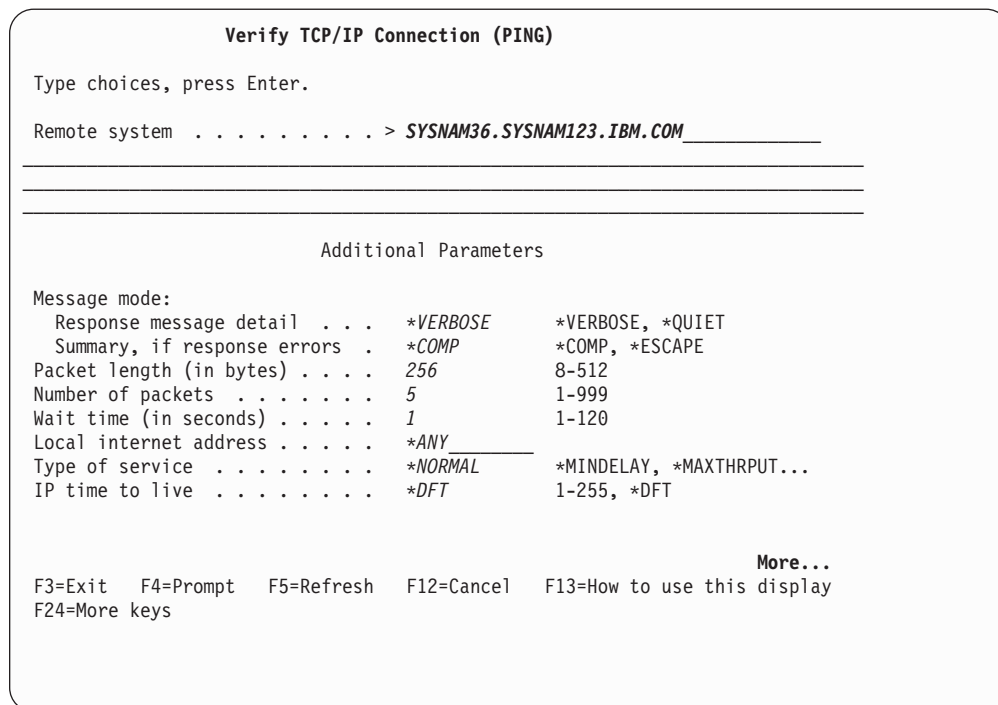


Figure 30. Verifying Connection to Remote System SYS1

Verifying TCP/IP Connections with Internet Address—Example

In this example, (Figure 30) the connection to the remote system at Internet address 9.4.191.76 is verified using the system defaults for packet length, number of packets, and wait time.

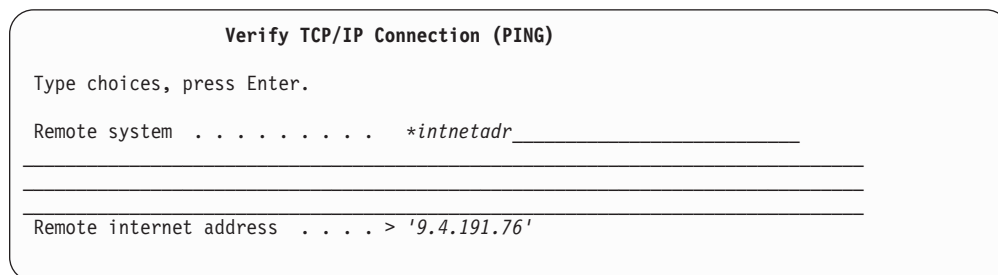


Figure 31. Verifying Connection to Remote System at Internet Address 9.4.191.76

Step 10—Saving Your TCP/IP Configuration

To save your TCP/IP configuration files, use the following command:

```
SAVOBJ OBJ(QATOC* QATM*) LIB(QUSRSYS)
      DEV(TAP01) OBJTYPE(*FILE)
```

The associated line descriptions are not saved with this command. Configuration objects are saved with the system.

To maintain consistency, save all TCP/IP configuration files together.

Note: You do not have to end TCP/IP in order to save the configuration files. However, you should end TCP/IP before any TCP/IP configuration files are restored.

TCP/IP Planning Checklists

The following checklists (Table 3 and Table 4) can help you prepare for the installation and configuration of TCP/IP on your network

- Line description parameters
- Local TCP/IP host information

Line Description Parameters Checklist

Table 3. Line Description Parameters

Line Type	*ELAN	*TRLAN	*WLS	*DDI	*FR	*X25	*ASYNC	*PPP	*TDLC
Resource name	R	R	R	R		R	R	R	
Local adapter address	O	O	O	O					
Speed		O	O	O	O	O	O	O	
SSAP (session services access point)	O	O	O	O	O				
Maximum frame size	O	O	O	O	O	O	O	O	
Local manager mode				O					
Attached non-switched NWI name					R				
Data link connection ID					R				
Network controller					R				
Connection type						R			
Logical channel identifier						R			
Logical channel type						R			
PVC (permanent virtual circuit) controller						R			
Local network address						R			
Physical interface type						O			
Packet size						O			
Window size						O			
Attached workstation controller									R

Note:

R means the parameter is required

O means OS/400 suggests a default value

Local TCP/IP Host Information Checklist

Table 4. Local TCP/IP Host Information

Interfaces to Local TCP/IP Networks			
	Interface #1	Interface #2	Interface #3

Table 4. Local TCP/IP Host Information (continued)

Internet address			
Line description name			
Subnet mask			
Interface MTU			
Local host name			
Local domain name			
Domain name server (Internet address)			
Default route/next hop (Internet address)			
IP datagram forwarding (yes or no)			
Explicit Routes to Remote TCP/IP Networks			
	Route #1	Route #2	Route #3
Internet address			
Subnet mask			
Next hop (Internet address)			
MTU size			
Local Host Table Entries: Remote TCP/IP Hosts			
Internet address	Host Name #1	Host Name #2	Host Name #3
X.25 / Remote System Information			
	Host #1	Host #2	Host #3
Internet address			
X.25 network address			
PVC channel ID			
Packet or window size			

Sample Network Drawing

The following network illustration can help you as you begin to draw your own network diagram. By performing this exercise, you will have a clear idea of how to attach your AS/400 system to the other systems in the network.

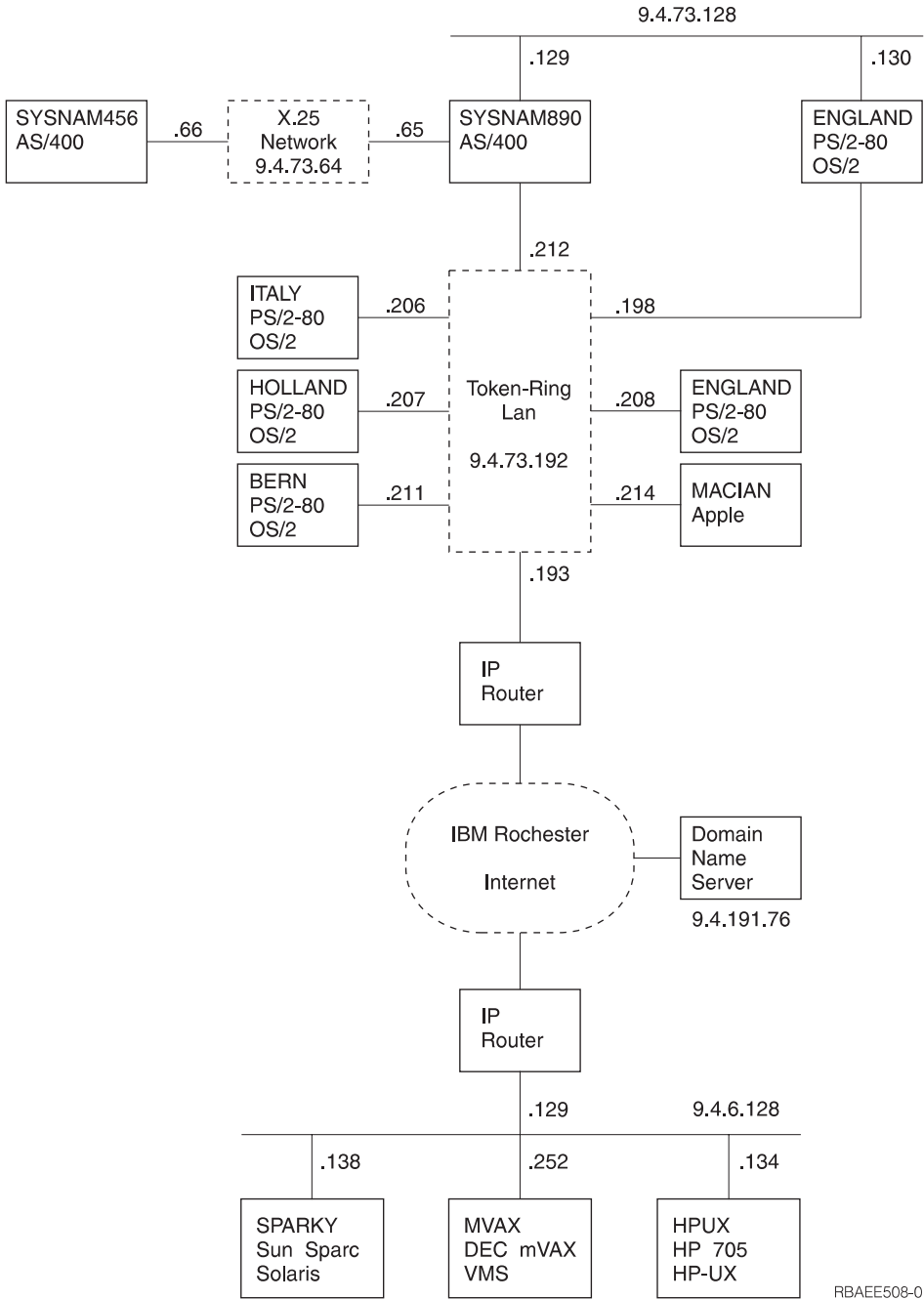


Figure 32. Sample TCP/IP Network

Chapter 3. TCP/IP: Operation, Management, and Advanced Topics

This chapter discusses managing your network by using the NETSTAT command, and the maintenance of host tables. In addition, this chapter covers other topics beyond those that are required to configure and use TCP/IP on AS/400. This information may help you to understand and maximize your usage of the AS/400 TCP/IP support.

TCP/IP on an AS/400 system can also be managed by Simple Network Management Protocol (SNMP). For information about SNMP, see the book, *Simple Network Management Protocol (SNMP) Support*, SC41-5412-00.

Network Status

The network status function on the AS/400 system allows you to get information about the status of TCP/IP network interfaces, routes, and connections on your local system. This function also allows you to end TCP/IP connections and to start or end TCP/IP interfaces.

NETSTAT displays the current TCP/IP protocol stack information. This information does not necessarily match the configuration data you see when using the Configure TCP/IP (CFGTCP) menu. In most cases, the NETSTAT command displays more information than the configuration data. In some cases, the configuration data might even change.

The reason for such a change is that the AS/400 TCP/IP dynamically creates some information, such as *DIRECT routes, when TCP/IP starts. A change may also occur if the configuration data that was sent to TCP/IP when it starts is changed dynamically by TCP/IP applications that run after you start TCP/IP. Several types of processing alter the initial TCP/IP configuration:

- Internet Control Message Protocol (ICMP) requests
- Sockets `ioctl` system calls
- Simple Network Management Protocol (SNMP) requests
- AS/400 TCP/IP internal processing

Work with TCP/IP Network Status Menu

The Work with TCP/IP Network Status menu allows you to work with the various network status functions.

To display the Work with TCP/IP Network Status menu, take these steps:

1. Type the WRKTCPSTS (Work with TCP/IP Network Status) command or the NETSTAT (Network Status) command.
2. Press the Enter key. (See Figure 33 on page 56.)

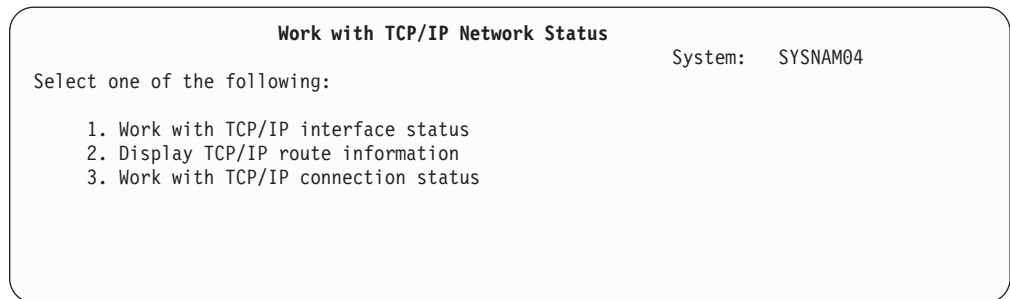


Figure 33. Work with TCP/IP Network Status

Work with TCP/IP Interface Status

The Work with TCP/IP Interface Status display, as shown in Figure 34, provides the most current summary of interface activity. This display allows you to view TCP/IP interface information for selected interfaces and to start or end TCP/IP interfaces. To view the Work with TCP/IP Interface Status display, take these steps:

1. Type 1 on the command line of the Work with TCP/IP Network Status menu or enter the WRKTCPSTS *IFC command.
2. Press the Enter key.

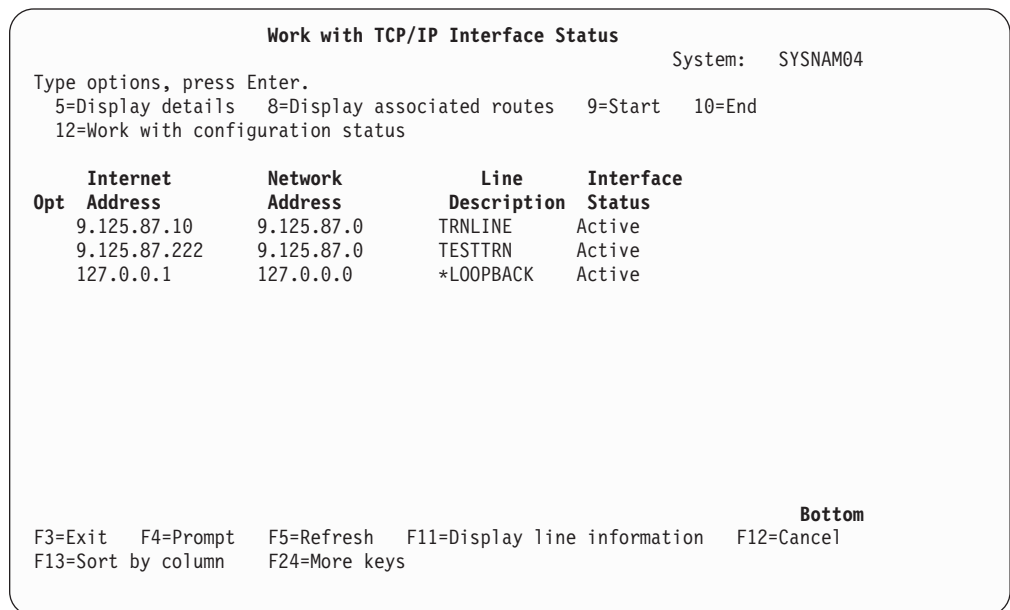


Figure 34. Work with TCP/IP Interface Status, Display 1 of 2

Press F11 to change the contents of the display to include the subnet mask, type of service, maximum transmission unit (MTU), and line type, as shown in Figure 35 on page 57.

```

Work with TCP/IP Interface Status
System:  SYSNAM04
Type options, press Enter.
5=Display details  8=Display associated routes  9=Start  10=End
12=Work with configuration status

  Internet      Subnet      Type of      Line
Opt  Address    Mask        Service      MTU  Type
  9.125.87.10   255.255.255.0 *MAXTHRPUT   1989 *TRLAN
  9.125.87.222  255.255.255.0 *NORMAL      1989 *TRLAN
  127.0.0.1     255.0.0.0    *NORMAL      576  *NONE

```

Figure 35. Work with TCP/IP Interface Status, Display 2 of 2

Starting TCP/IP Interfaces

TCP/IP interfaces are started in one of the following ways:

- The Work with TCP/IP Interface Status displays are reached by:
 - Option 1 on the Configure TCP/IP (CFGTCP) menu
 - Option 1 on the Network Status (NETSTAT or WRKTCPSTS) menu
- The Start TCP/IP Interface (STRTCPIFC) command
- Using the Operations Navigator interface

Note: You can start TCP/IP interfaces through the Operations Navigator interface wizard. However, this chapter does not document any of the Operations Navigator functions. See the online help in Operations Navigator for this information.

To start a TCP/IP interface from the Work with TCP/IP Interface Status menu, type 9 in the option field for each interface that you want to start and press the Enter key.

To start a TCP/IP interface using the STRTCPIFC command, take these steps:

1. Type STRTCPIFC on the command line and press F4 (Prompt).
2. Type the Internet address of the interface that you want to start and press the Enter key.

Option 9 on the Work with TCP/IP Interface Status display is used to start both TCP/IP interfaces and Internet Protocol (IP) over Systems Network Architecture (SNA) interfaces. For information about starting IP over SNA interfaces, see the STRIPSIFC (Start IP over SNA Interface) command in the *CL Reference (Abridged)*, SC41-5722.

Note: When starting the first TCP/IP interface associated with an Integrated Netfinity Server for AS/400 (also known as File Server Input/Output Processor and FSIOP) network server description, a considerable amount of time may pass before the interface becomes active. This is because TCP/IP activation includes starting the network server. The amount of time that is required depends mainly on machine use and the size of the processor. To determine whether the interface has started, view the messages in the QTCPIP job log and the QSYSOPR message queue.

Ending TCP/IP Interfaces

The ENDTCPIFC (End TCP/IP Interface) command ends an existing TCP/IP interface immediately. As a result, all TCP/IP connections using this interface also end immediately. However, the operation of any other TCP or IP over SNA interface, using the same line description as the interface that is ending, is not affected.

TCP/IP interfaces can be ended in one of two ways:

- Using the Work with TCP/IP Interface Status display, which is reached by:
 - Option 1 on the Configure TCP/IP (CFGTCP) menu
 - Option 1 on the Network Status (NETSTAT or WRKTCPSTS) menu
- Using the ENDTCPIFC (End TCP/IP Interface) command

To end a TCP/IP interface from the Work with TCP/IP Interface Status menu:

1. Type 10 in the option field for each interface that you want to end.
2. Press the Enter key.

To end a TCP/IP interface using the ENDTCPIFC command:

1. Type ENDTCPIFC on the command line.
2. Press F4 (Prompt).
3. Type the Internet address of the interface that you want to end.
4. Press the Enter key.

Option 10 on the Work with TCP/IP Interface Status display is used to end both TCP/IP interfaces and IP over SNA interfaces. For information about ending IP over SNA interfaces, see the ENDIPSIFC (End IP over SNA Interface) command in the *CL Reference (Abridged)*, SC41-5722-03.

Route-to-Interface Binding: Interfaces define direct paths to networks or subnetworks to which an AS/400 system is directly attached. Routes define indirect paths. A route identifies the first hop on the path to a network or subnetwork to which an AS/400 system is not directly attached.

Routes are bound to interfaces through the use of a best-match-first algorithm. This algorithm is based on the state of the interface, and on the type of service (TOS) specified for the route and interface. When you end an interface, the routes associated with the interface can move to another existing active interface if the following conditions are satisfied:

- If the TOS for the route is something other than *NORMAL, the algorithm looks for an interface with the same TOS. If an interface with the specified TOS is not found, an interface with TOS *NORMAL is sought. Again, if one is not found, that route will not be moved.
- The MTU value for the route that is being moved must be less than or equal to the MTU value for the active interface.
- The network ID of the interface must be equal to the logical AND of the next hop for the route and the subnet mask for the interface.

Notes:

1. If the next hop of a route is identical to an interface's IP address, that route will never be bound to another interface.
2. When starting interfaces (if all interfaces are currently inactive) routes are bound to the interfaces with the same best-match-first algorithm. An exception is if the

route is defined with a preferred binding interface. In this case, an attempt is made to bind the route to the interface that is indicated. If the binding attempt fails, then the best-match-first algorithm is used.

Display TCP/IP Route Information

The display TCP/IP route information function allows you to view information about TCP/IP routes.

To display TCP/IP route information:

1. On the Work with TCP/IP Network Status menu, type 2 on the command line or enter the WRKTCPSTS *RTE command.
2. Press the Enter key.

The first of the two Display TCP/IP Route Information displays appears, as shown in Figure 36.

Display TCP/IP Route Information					System: SYSNAM04
Type options, press Enter.					
5=Display details					
Opt	Route Destination	Subnet Mask	Next Hop	Route Available	
	9.125.87.0	255.255.255.0	*DIRECT	*YES	
	9.125.87.0	255.255.255.0	*DIRECT	*YES	
	9.125.109.3	*HOST	9.125.87.17	*YES	
	127.0.0.0	255.0.0.0	*DIRECT	*YES	
	*DFTRROUTE	*NONE	9.125.87.169	*YES	
	*DFTRROUTE	*NONE	9.125.87.250	*YES	
					Bottom
F3=Exit	F5=Refresh	F6=Print list	F11=Display route type	F12=Cancel	
F13=Sort by column		F17=Top	F18=Bottom		

Figure 36. Display TCP/IP Route Information, Display 1 of 2

To view the second display, press F11 (Display route type). The route information is presented as shown in Figure 37 on page 60. To return to the first display, press F11 (Display next hop).

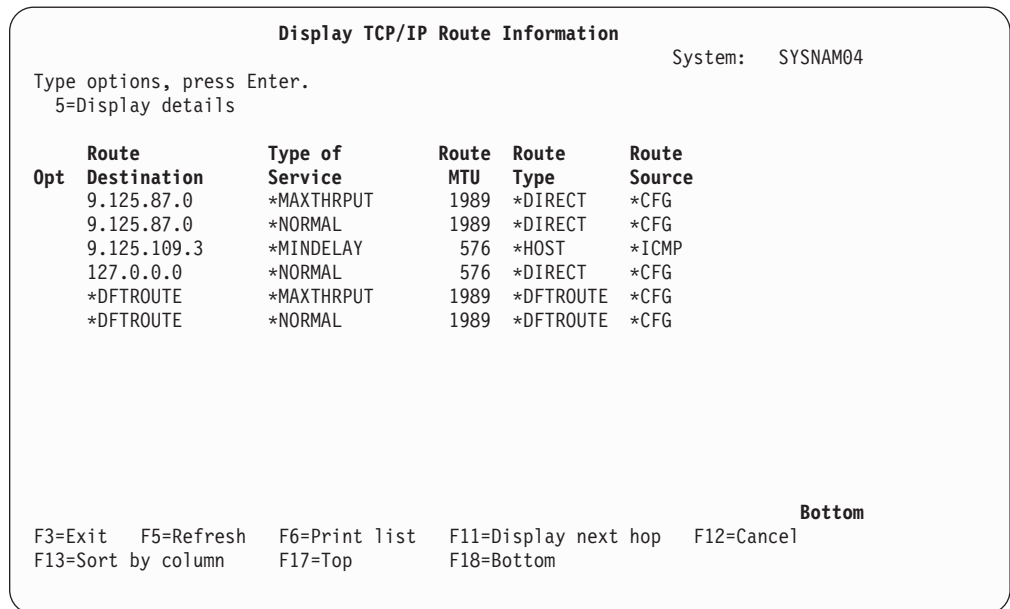


Figure 37. Display TCP/IP Route Information, Display 2 of 2

To view detailed information about a specific route, type 5 in the option field next to the route and press the Enter key.

Routes listed on the Display TCP/IP Route Information display differ from the routes that are displayed on the Work with TCP/IP Routes display. Only routes with a route source of *CFG and a route type that is not *DIRECT can be changed with the Work with TCP/IP Routes display. Similarly, only routes that meet these conditions can be changed or removed with the CHGTCPRTE or RMVTCPRTE commands. *CFG means the route was added using AS/400 configuration commands or is a *DIRECT route. *DIRECT means that the route is to a network or subnetwork to which this system has a direct physical connection. This route is not defined with an add route command.

Work with TCP/IP Connection Status

The Work with TCP/IP Connection Status display allows you to display or end a TCP/IP connection between a local system and a remote system.

To display the Work with TCP/IP Connection Status display:

1. Type 3 on the command line of the Work with TCP/IP Network Status menu or enter the WRKTCPSTS *CNN command.
2. Press the Enter key.

The first of the three Work with TCP/IP Connection Status displays, as shown in Figure 38 on page 61.

To display the second and third Work with TCP/IP Connection Status displays, press F11 (see Figure 39 on page 61 and Figure 40 on page 62). To display port numbers instead of port service names, press F14.

In Figure 38 on page 61, the connections indicate that the FTP server, SMTP server, and TELNET server are active and ready to receive connection attempts.

Because no connection has been established yet, the *Remote Address* and *Remote Port* fields contain an asterisk (*). When an application requests a connection to a listening socket, a new connection is created. The remote Internet address and remote port are shown for the new connection. The listening socket always remains in the list of connections.

```

Work with TCP/IP Connection Status
System:  SYSNAM04
Local internet address . . . . . : *ALL

Type options, press Enter.
  4=End  5=Display details

  Opt  Remote      Remote   Local   Idle Time  State
  Address      Port     Port
  *      *          *      ftp-con > 000:20:41 Listen
  *      *          *      telnet    001:39:00 Listen
  *      *          *      telnet    000:14:27 Listen
  *      *          *      smtp     000:55:23 Listen
  *      *          *      lpd      002:36:29 Listen
  *      *          *      1049     001:31:01 *UDP
  *      *          *      1050     001:28:02 *UDP
  *      *          *      1051     001:12:05 *UDP
  *      *          *      1052     001:09:52 *UDP
  *      *          *      1070     000:35:53 Listen
  9.5.1.180    1211     telnet   000:10:17 Established

  More...

F5=Refresh  F11=Display byte counts  F13=Sort by column
F14=Display port numbers  F22=Display entire field  F24=More keys

```

Figure 38. Work with TCP/IP Connection Status, Display 1 of 3

```

Work with TCP/IP Connection Status
System:  SYSNAM04
Local internet address . . . . . : *ALL

Type options, press Enter.
  4=End  5=Display details

  Opt  Remote      Remote   Local   User      Bytes Out  Bytes In
  Address      Port     Port
  *      *          *      ftp-con > QTCP      0          0
  *      *          *      telnet    QTCP      0          0
  *      *          *      telnet    QTCP      0          0
  *      *          *      lpd      QTCP      0          0
  *      *          *      1070     BILANSKY 0          0
  9.5.1.131    1954     telnet    QTCP      48583     815
  9.5.1.180    1211     telnet    QTCP      32319     4704
  9.5.15.134   1024     telnet    QTCP      403415    226141
  9.5.15.141   1027     telnet    QTCP      3831      236
  9.130.38.18  2099     telnet    QTCP      509788    15394
  9.130.38.74  1125     telnet    QTCP      680       34

  More...

F5=Refresh  F11=Display connection type  F13=Sort by column
F14=Display port numbers  F22=Display entire field  F24=More keys

```

Figure 39. Work with TCP/IP Connection Status, Display 2 of 3

```

Work with TCP/IP Connection Status
System: SYSNAM04
Local internet address . . . . . : *ALL
Type options, press Enter.
4=End 5=Display details

  Remote      Remote      Local      Local
Opt Address    Port      Address    Port      Type
*            *          *          ftp-con > *TCP
*            *          *          telnet    *TCP
*            *          *          telnet    *TCP
*            *          *          lpd       *TCP
*            *          9.125.87.222 1070     *TCP
9.5.1.131    1954      9.125.87.10  telnet    *TCP
9.5.1.180    1211      9.125.87.10  telnet    *TCP
9.5.15.134   1024      9.125.87.10  telnet    *TCP
9.130.38.18  2099      9.125.87.222 telnet    *TCP
9.130.38.74  1125      9.125.87.10  telnet    *TCP
9.130.38.74  1126      9.125.87.222 telnet    *TCP

F5=Refresh F11=Display connection state F13=Sort by column
F14=Display port numbers F22=Display entire field F24=More keys
More...

```

Figure 40. Work with TCP/IP Connection Status, Display 3 of 3

Ending TCP/IP Connections

TCP/IP connections and User Datagram Protocol (UDP) sockets can be ended from the Work with TCP/IP Connection Status display. To do so:

1. Type 4 in the option field for the lines containing the connections that you want to end.
2. Press the Enter key.

The Confirm End of TCP/IP Connections displays is then presented as shown in Figure 41 on page 63.

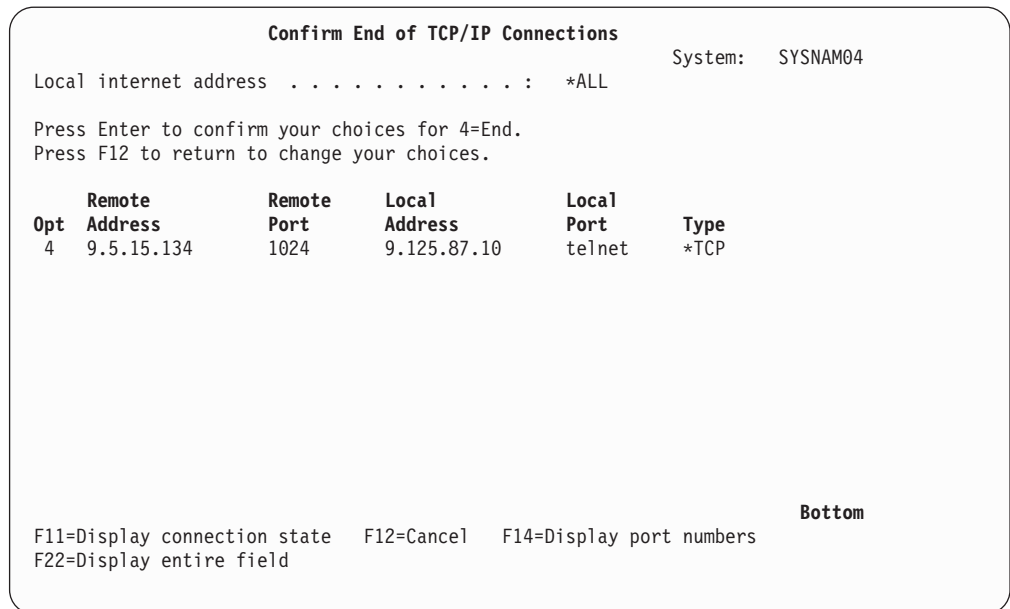


Figure 41. Confirm End of TCP/IP Connections

To end the TCP/IP connections, press the Enter key from the Confirm End of TCP/IP Connections display.

If you decide not to end a TCP/IP connection or if you want to change your choices, press F12 (Cancel).

Working with Configuration Status

To work with the line description used by an interface:

1. On the Work with TCP/IP Interface Status menu, type 12 in the option field for each interface that you want to work with.
2. Press the Enter key.

This option issues the WRKCFGSTS (Work with Configuration Status) command for the line description associated with the interface. Using the options shown in Figure 42 on page 64 you can vary a line description on or off, display the Work with Job menu, and display the line description or mode status.

This option cannot be used for IP over SNA interfaces because IP over SNA does not use specific line descriptions.

```

Work with Configuration Status                                SYSNAM04
                                                            04/26/94 15:55:58
Position to . . . . . Starting characters
Type options, press Enter.
 1=Vary on   2=Vary off  5=Work with job  8=Work with description
 9=Display mode status ...

Opt  Description      Status      -----Job-----
TRNLINE          ACTIVE
TRNLINET         ACTIVE
TRNLITCP         ACTIVE      QTCPIP      QTCP      007936

```

Figure 42. Work with Configuration Status

Displaying TCP/IP Network Status Information

In addition to working with network status functions, the Work with TCP/IP Network Status menu allows you to display current information about your TCP/IP network, including multicast groups, TCP/IP interfaces, and associated routes, to name a few.

Display Multicast Groups

To display the multicast groups associated with an interface:

1. On the Work with TCP/IP Interface Status display, type 14 in the option field for each interface for which you want to see the associated multicast groups.
2. Press the Enter key.

Figure 43 on page 65 illustrates the display of the multicast groups for an Ethernet interface.

If you have requested multicast group information for more than one interface, press the Enter key to review the remaining displays.

```

Display Multicast Host Groups
System:  SYSNAM04
Interface internet address . . . . . : 10.5.5.55

Host Group      Hardware Address      Host Group      Hardware Address
224.0.0.1       01:00:5E:00:00:01
225.4.5.6       01:00:5E:04:05:06
233.32.40.51    01:00:5E:20:28:33
224.0.0.9       01:00:5E:00:00:09
229:200:100:1   01:00:5E:48:64:01

Bottom
F3=Exit  F5=Refresh  F6=Print  F9=Command line  F11=Hide hardware address
F12=Cancel

```

Figure 43. Display Multicast Host Groups

Displaying TCP/IP Interfaces

To display more detailed information about the TCP/IP interface status for specific interfaces:

1. On the Work with TCP/IP Interface Status display, type 5 in the option field for each interface about which you want more information.
2. Press the Enter key.

If you requested status for a token-ring interface, the information displays, as shown in Figure 44 on page 66.

If you have requested interface status information for more than one interface, press the Enter key to view the remaining displays.

```

Display TCP/IP Interface Status
System:  SYSNAM04
Interface host name . . . . . : sysnam04.endicott.ibm. >
Internet address . . . . . : 9.125.87.10
Subnet mask . . . . . : 255.255.255.0
Network address . . . . . : 9.125.87.0
Host address . . . . . : 0.0.0.10
Directed broadcast address . . . . . : 9.125.87.255

Interface status . . . . . : Active
Change date/time . . . . . : 04/26/94 14:32:32
Line description . . . . . : TRNLINE
Line type . . . . . : *TRLAN
Type of service . . . . . : *MAXTHRPUT
Maximum transmission unit . . . . . : 1989
Automatic start . . . . . : *YES

TRLAN bit sequencing . . . . . : *MSB

```

Figure 44. Display TCP/IP Interface Status for a Token-Ring Interface

Displaying Associated Routes

To display information about the routes associated with a specific interface:

1. On the Work with TCP/IP Interface Status display, type 8 in the option field for each interface for which you want to see the associated routes information.
2. Press the Enter key.

The first of two displays with associated route information is shown in Figure 45 on page 67.

If you have requested associated route information for more than one interface, press the Enter key to view the remaining displays.

```

Display Associated Routes
System:  SYSNAM04
Interface internet address . . . . . : 9.125.87.10

Type options, press Enter.
5=Display details

  Route      Subnet      Next      Route
Opt Destination Mask      Hop      Available
    9.125.87.0 255.255.255.0 *DIRECT    *YES
    *DFTRROUTE *NONE          9.125.87.169 *YES

Bottom
F3=Exit   F5=Refresh  F6=Print list  F11=Display route type  F12=Cancel
F13=Sort by column  F17=Top      F18=Bottom

```

Figure 45. Associated Route Information, Display 1 of 2

Press F11 to show the display that includes the type of service (TOS), maximum transmission unit (MTU), type, and source.

Displaying Route Details Option

To display detailed information about the route:

1. On the Display Associated Routes display, type 5 in the option field for each route about which you want more information.
2. Press the Enter key.

Figure 46 on page 68 and Figure 47 on page 68 are examples.

```

                                Display TCP/IP Route Details
                                System:  SYSNAM04

Route information:
Route destination . . . . . : 9.125.87.0
Subnet mask . . . . . : 255.255.255.0
Next hop host name . . . . . : sysnam04.endicott.ibm. >
Next hop . . . . . : *DIRECT
Type of service . . . . . : *MAXTHRPUT
Route available . . . . . : *YES
Route type . . . . . : *DIRECT
Route source . . . . . : *CFG
Change date/time . . . . . : 04/26/94 14:32:32
Route maximum transmission unit . . . . . : 1989
Reference count . . . . . : 0

Local interface information:
Internet address . . . . . : 9.125.87.10
Subnet mask . . . . . : 255.255.255.0
Network address . . . . . : 9.125.87.0

Press Enter to continue.

F3=Exit  F6=Print  F12=Cancel  F22=Display entire field
More...

```

Figure 46. Display TCP/IP Route Details, Display 1 of 2

```

                                Display TCP/IP Route Details
                                System:  SYSNAM04

Interface status . . . . . : Active
Line description . . . . . : TRNLIN
Line type . . . . . : *TRLAN

```

Figure 47. Display TCP/IP Route Details, Display 2 of 2

Displaying TCP/IP Route Information

To display TCP/IP route information:

1. On the Work with TCP/IP Network Status menu, type 2 on the command line or enter the WRKTCPSTS *RTE command.
2. Press the Enter key.

The first of the two Display TCP/IP Route Information displays is presented as shown in Figure 48 on page 69.

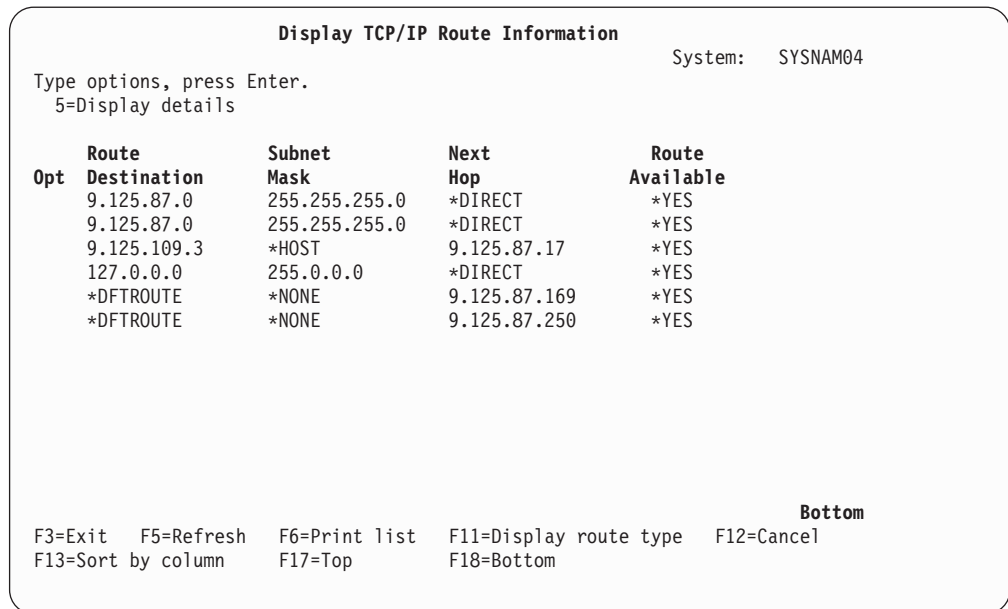


Figure 48. Display TCP/IP Route Information, Display 1 of 2

To view the second Display TCP/IP Route Information display, press F11 (Display route type). The route information is presented in Figure 49. To return to the first display, press F11 (Display next hop).

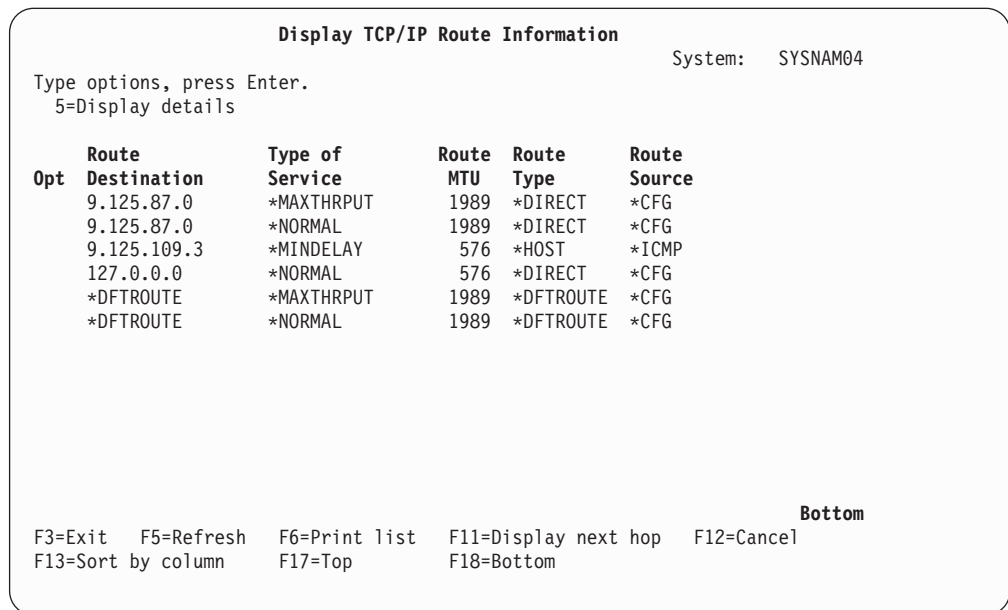


Figure 49. Display TCP/IP Route Information, Display 2 of 2

To view detailed information about a specific route, type 5 in the option field next to the route and press the Enter key. See Figure 46 on page 68 and Figure 47 on page 68.

Displaying TCP/IP Connections

You can request more detailed information about TCP/IP connections shown on the Work with TCP/IP Connection Status display. This information includes timing information and transmission statistics for the connection displayed.

To display more information about the listed TCP/IP connections:

1. Type 5 in the option field for each connection about which you want more information.
2. Press the Enter key.

A series of up to three displays for each connection appears. Press the Page Down key to view the remaining displays.

The contents of the displays vary depending on the type of connection, whether *TCP, *UDP, or *IPS. (Figure 50, Figure 51 on page 71, and Figure 52 on page 71 show displays for a TCP connection.)

```

                                     Display TCP Connection Status
                                     System:  SYSNAM04
Connection identification:
Remote host name . . . . . : drfun.rchland.ibm.com
Remote internet address . . . . . : 9.5.15.134
Remote port . . . . . : 1025
Local host name . . . . . : sysnam04.endicott.ibm. >
Local internet address . . . . . : 9.125.87.143
Local port . . . . . : telnet
Associated user profile . . . . . : QTCP
TCP programming interface information:
State . . . . . : Established
Connection open type . . . . . : Passive
Timing information:
Idle time . . . . . : 000:00:00.381
Last activity date/time . . . . . : 05/25/94 14:38:11
Round-trip time . . . . . : .133
Round-trip variance . . . . . : .016

                                     More...

Press Enter to continue.
F3=Exit  F5=Refresh  F6=Print  F10=Display IP options  F12=Cancel
F14=Display port numbers  F22=Display entire field
```

Figure 50. Display TCP/IP Connection Status, Display 1 of 3

```

Display TCP Connection Status
System:  SYSNAM04
Bytes out . . . . . : 57692
  Outgoing bytes buffered . . . . . : 0
  User send next . . . . . : 3270868150
  Send next . . . . . : 3270868150
  Send unacknowledged . . . . . : 3270868150
  Outgoing push number . . . . . : 3270868149
  Outgoing urgency number . . . . . : 3270868149
  Outgoing window number . . . . . : 3270896558
Bytes in . . . . . : 1021
  Incoming bytes buffered . . . . . : 0
  Receive next . . . . . : 1545153023
  User receive next . . . . . : 1545153023
  Incoming push number . . . . . : 1545153023
  Incoming urgency number . . . . . : 1545153022
  Incoming window number . . . . . : 1545160742

More...

Press Enter to continue.
F3=Exit  F5=Refresh  F6=Print  F10=Display IP options  F12=Cancel
F14=Display port numbers  F22=Display entire field

```

Figure 51. Display TCP/IP Connection Status, Display 2 of 3

```

Display TCP Connection Status
System:  SYSNAM04
Retransmission information:
  Total retransmissions . . . . . : 8
  Current retransmissions . . . . . : 0
Send window information:
  Maximum size . . . . . : 28672
  Current size . . . . . : 28408
  Last update . . . . . : 1545153004
  Last update acknowledged . . . . . : 3270868150
  Congestion window . . . . . : 2704
  Slow start threshold . . . . . : 1281
Precedence and security:
  Precedence . . . . . : 0
Initialization information:
  Maximum segment size . . . . . : 536
  Initial send sequence number . . . . . : 3270810457
  Initial receive sequence number . . . . . : 1545152001

Bottom

Press Enter to continue.
F3=Exit  F5=Refresh  F6=Print  F10=Display IP options  F12=Cancel
F14=Display port numbers  F22=Display entire field

```

Figure 52. Display TCP/IP Connection Status, Display 3 of 3

Displaying Connection Totals

To display a summary of TCP and UDP counts, press F10 on the Work with TCP/IP Connection Status display. The counts provided are a cumulative summary of all TCP and UDP activity since the last time the STRTCP (Start TCP) command was issued.

The information in Figure 53 and Figure 54 shows TCP and UDP counts that are maintained for Simple Network Management Protocol (SNMP). For additional information about SNMP, see the *Simple Network Management Protocol (SNMP) Support* book.

```

Display TCP/IP Connection Totals
System:  SYSNAM04

TCP connection information:
Currently established . . . . . : 1
Active opens . . . . . : 0
Passive opens . . . . . : 0
Attempted opens that failed . . . . . : 0
Established and then reset . . . . . : 0

TCP send information:
Segments sent . . . . . : 108
Retransmitted segments . . . . . : 10
Reset segments . . . . . : 0

TCP receive information:
Segments received . . . . . : 117
Segments received in error . . . . . : 0

More...

Press Enter to continue.

F3=Exit  F5=Refresh  F6=Print  F12=Cancel

```

Figure 53. Display TCP/IP Connection Totals, Display 1 of 2

```

Display TCP/IP Connection Totals
System:  SYSNAM04

UDP send information:
Datagrams sent . . . . . : 0

UDP receive information:
Datagrams received . . . . . : 0
Datagrams not delivered . . . . . : 0
Application port not found . . . . . : 0
Other datagrams in error . . . . . : 0

```

Figure 54. Display TCP/IP Connection Totals, Display 2 of 2

TCP/IP Host Tables

Host tables are a method for mapping host names to IP addresses. This is done by using a hosts file for name-to-address resolution. Because the host table lacks the structure to list names in any hierarchical order, names assigned to hosts must be unique. In the topics that follow, you will find discussions about the overall management of TCP/IP host tables. Instructions for merging host tables and managing a host table from a central site are included.

Successful TCP/IP host table maintenance also includes periodically evaluating whether or not to use a DNS server to manage your network. The DNS server is often the preferred alternative to host tables for the purpose of managing IP addresses and host names, particularly in large network environments. However,

even some small organizations that access the Internet require a DNS server to meet their name-service needs. See “Chapter 18. AS/400 Domain Name System (DNS)” on page 421 for more information.

Managing TCP/IP Host Tables

In a large network, it can be more efficient to administer AS/400 TCP/IP from a central site. Working with the host table would be time consuming if each system is individually updated with the TCP/IP configuration menu. Updates can be made more quickly on one system and then copied to others.

AS/400 TCP/IP is designed to protect configuration files, including the host table. You cannot change the host table file unless you use the Configure TCP/IP menu or the MRGTCPHT, ADDTCPHT, RNMTCPHT, CHGTCPHT, or RMVTCPTHTE commands. However, you can still import and use a host table from a central site by using the MRGTCPHT command.

The following host table file types can be imported and merged with the AS/400 host table:

- Host table type ***AS400**, generated by AS/400 TCP/IP Version 3 Release 1 Modification 0 (V3R1M0) or later
- Host table type ***AIX**, generated by AS/400 TCP/IP Version 3 Release 0 Modification .5 (V3R0M5), Version 2 Release 3 (V2R3) or earlier, or many other IBM and non-IBM systems
- Host table type ***NIC**, host table format used by public domain systems

You can merge or replace the local AS/400 host table with the imported host table. The name of the database file containing the local host table is QATOCHOST with member HOSTS in library QUSRSYS. This file is used directly by AS/400 TCP/IP; no conversion into an internal version takes place.

Host File Formats

If you receive a host file and want to use it on your system, the MRGTCPHT (Merge TCP/IP Host Table) command allows you to specify which format you are using. You can use host information files that are in either the *NIC format, the *AIX format, or the *AS400 format. The record length of the imported host table file is not limited.

Host Table Information with *AIX Files

Table 5 shows the *AIX format supported on the AS/400 system.

*Table 5. *AIX Supported on the AS/400 System*

Delimiter	Meaning
# (pound sign)	Indicates the beginning of a comment. The text following the pound sign is a comment and is not part of the host table.
blank, tab	Indicates a field delimiter.

Host Table Information with *NIC Files

The *NIC format is often used by hosts in the public domain. A record in a *NIC file has the following format:

```
HOST : 128.12.19.1 : Host2.lan.ibm.com,Host2 : PC-AT : DOS : TCP/IP
```

This entry describes one host (at address 128.12.19.1) with two names (Host2.lan.ibm.com) and (Host2). The host is an IBM Personal Computer AT computer running MS-DOS and supporting TCP/IP.

A complete description of the *NIC format is found in Request for Comment (RFC) 952, *Internet Host Table Specification*. The subset supported on the AS/400 system is shown in Table 6. The *NIC continuation characters are not supported because the record length of the file can be up to 512 bytes.

Table 6. *NIC Subset Supported on the AS400 System

Delimiter	Meaning
; (semicolon) ¹	Indicates the beginning of a comment. The text following the semicolon is a comment and is not part of the host table.
NET ²	A keyword introducing a network entry.
GATEWAY	A keyword introducing a gateway entry.
HOST	A keyword introducing a host entry.
: (colon)	A field delimiter.
:: (two colons)	Indicates a null field.
, (comma)	A data element delimiter.
Notes:	
1. If any line in the *NIC table contains a semicolon as the first column value, then that line is not merged into the AS/400 host table.	
2. These entries are not merged into the AS/400 host table.	

Host Table Information with *AS400 Files

The *AS400 file format is the format of the local AS/400 host table file used by AS/400 TCP/IP directly. The name of the file is QATOCHOST with member HOSTS in library QUSRSYS. A single record contains an Internet address, up to four host/domain names and a text description field. For more details regarding record and file formats, use the DSPFFD (Display File Field Description) command.

This file can be exchanged between AS/400 systems. However, there is no function to convert from *AS400 to *AIX or *NIC format.

Tips for Merging Host Tables

A maximum of four host names per IP address is allowed when host tables are merged. For example, if the local host table already has three host names and the physical file member to be merged has two additional host names, only the first host name in the physical file is merged into the final host table.

Host names that exist for the same Internet address are not duplicated. If the same host name is found for Internet addresses that are different, then that host name is accepted, but a warning message is displayed.

The original copy of the local host table is not saved by the MRGTCPTH (Merge TCP/IP Host Table) command. To save the original host table, create a copy of the file QUSRSYS/QATOCOST.HOSTS by using the Copy File (CPYF) command. Do this before issuing the MRGTCPTH command.

Merging TCP/IP Host Tables

You can use imported host tables in two ways:

- Overwrite the current host table. To do this, specify Replace Host Table (*Yes) on the Merge Host Table display.
- Merge the information of the imported host table with the information that was entered by using option 10 (Work with TCP/IP host table entries) from the Configure TCP/IP menu. To merge the information, specify Replace Host Table (*No) on the Merge Host Table display.

You can merge an imported host table with the local host table while TCP/IP is running by using the CFGTCP (Configure TCP/IP) command. The changes take affect the next time a TCP/IP application accesses the host table.

Select option 11 to merge an imported host table with the local AS/400 host table.

You can also use the Merge TCP/IP Host Table (MRGTCPTH) command from any command line.

Example: Successful Host Table Merge

The following example shows the command to merge an imported host table with the local host table.

```
MRGTCPTH FROMFILE(QUSRSYS/M02HOSTS) FILEFMT(*AS400) REPLACE(*NO)
```

```
File M02HOSTS, member *FIRST, successfully merged with host table.
```

Example: Partly Successful Host Table Merge

The following example shows the command to merge an imported host table with the local host table.

```
MRGTCPTH FROMFILE(QUSRSYS/M03HOSTS) FILEFMT(*AS400) REPLACE(*NO)
```

```
Duplicate host name SPARKY.SYSNAM123.IBM.COM at address 9.4.6.138 found host table.
```

```
Duplicate host name MVAX.SYSNAM123.IBM.COM at address 9.4.6.252 found host table.
```

```
File M03HOSTS, member *FIRST, merged with host table: however, error occurred.
```

In this example, the host table contains entries with the same host name, which shows in the message as duplicate host names.

Managing the Host Table from a Central Site

If your network has multiple AS/400 systems, you can define the TCP/IP host table on one system and share that table with the other systems. This saves you the effort of having to define the host table on each system. To do this, follow these steps:

Step 1—Create the Host Table on Your Central System

Use the CFGTCP command to configure your host table. Select option 10 (Work with TCP/IP host table entries). Your system's host table is stored in member HOSTS of file QATOCHOST in library QUSRSYS.

Step 2—Start FTP to a Remote System

For example, if your host table defines the remote system as SYSNAM02, type the FTP command as follows:

```
ftp sysnam02
```

Step 3—Tell FTP to Send the Host File to the Remote System

Type the following FTP subcommand:

```
put qusrsys/qatochost.hosts qusrsys/m03host.hosts
```

Note: Do not use FTP to put the host file directly into file QATOCHOST containing the AS/400 host table.

Step 4—Merge the File

Type the following FTP subcommand:

```
quote rcmd mrgtcpht fromfile(qusrsys/m03host) frommbr(host)
```

Domain Name System (DNS) Server

The conversion from host name to Internet address can be performed by using the host table on the local system or by defining a Domain Name System server, or DNS server.

In large networks with large host tables, it is more convenient to have DNS servers than to have a complete copy of the host table on every host in the network. As a single source for host names, a DNS server is capable of storing the name and address translations for all of the computers on your network. The DNS server is the preferred alternative to managing host tables.

This chapter does not document DNS server functions. However, if you are interested in more detailed information about Domain Name System server (DNS) support on your AS/400 and configuring a DNS server see "Chapter 18. AS/400 Domain Name System (DNS)" on page 421.

IP Routing and Internet Control Message Protocol (ICMP) Redirecting

Internet routing tables usually remain static for long periods. TCP/IP generates routing tables at activation time from configuration data and adjusts the routing tables based on ICMP redirects, SNMP manager requests, dead gateway processing and socket routing requests.

If network interconnections change, routing tables in a particular host may become incorrect. Because gateways exchange routing information periodically to accommodate network changes and to keep their routes up to date, a gateway usually knows better routes than a host. When a gateway detects that a host is using a route that is not optimum, the gateway sends an ICMP redirect message to

that host. It also forwards the original datagram on to its destination. Redirect messages are limited to interactions between a gateway and a host on the same network.

If the host that sends the original datagram is an AS/400, it receives the ICMP redirect message from the gateway and uses this information to update its internal routing table. The next datagram is then sent using the more optimum route received from the gateway. You can see the updated routing table by using NETSTAT, option 2. A route created by the ICMP redirect mechanism is recorded in the IP dynamic routing table and remains there as long as an upper level protocol is using it. When the last upper-level protocol user has completed its unit of work using a route created by the ICMP redirect mechanism, the route is then removed from the routing table. When TCP/IP is restarted, this process is repeated.

In Figure 55, host A1 in network 2 is an AS/400 system that sends a message to host A2 in network 3. The routing table in host A1 indicates that the first hop to host A2 is through gateway G1, which connects networks 1 and 2. When this gateway receives the datagram, it forwards the datagram to gateway G2, which sends it to the host A2. Gateway G1 then sends an ICMP redirect message to host A1 to inform it that a better route to host A2 is to use gateway G2 as the first hop. This information updates the internal routing table in host A1, and the next datagram to host A2 in network 3 is sent to gateway G2 as the first hop. The gateway then sends the datagram to host A2. When the TCP/IP services are stopped, the collected routing information is deleted and host A1 starts the learning process again.

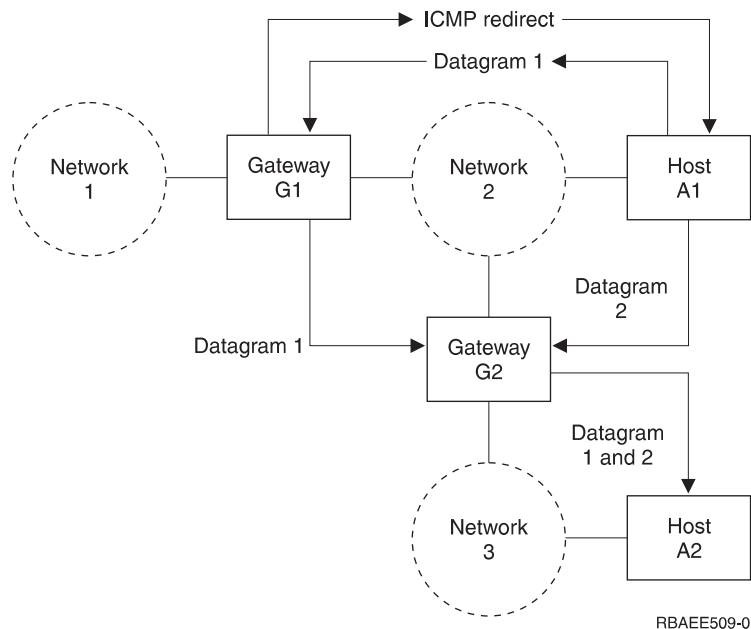


Figure 55. Example of ICMP Redirect

To see routing changes due to ICMP redirect messages, select NETSTAT menu 2 or NETSTAT *RTE and then press PF11. Comparing the next hop in this display with the next hop present in the routing table, you can verify whether a route has been dynamically changed.

Dead Gateway Processing

RFC-1122, *Requirements For Internet Hosts - Communication Layers*, requires the IP layer to include a dead gateway algorithm to manage suspected gateway failures. This section is intended to give you an overview of dead gateway processing.

Two types of gateway failures can occur:

- Failure of a first-hop gateway. A first-hop gateway is the gateway that is specified in an IP route. First-hop gateways must be on a directly-connected network. This type of failure can be detected by either TCP or the data link layer.
- Failure of a gateway other than the first-hop gateway. The path between source and destination TCP/IP hosts can traverse multiple gateways. This type of failure can be detected only by TCP.

Dead gateway processing is initiated when IP receives a negative advice indicator from either TCP or the data link layer. These indicators from TCP and the data link layer are referred to as advice since they may result from transient conditions as well as from a serious gateway failure.

Negative Advice from TCP or the Data Link Layer

Retransmissions on a TCP connection occur as a result of transient or non-transient problems somewhere along the path to a destination host. When TCP notices excessive retransmissions on a TCP connection, a TCP negative advice indicator is sent to IP.

The data link layer passes a negative advice indicator to IP when it is unable to transmit data to a first-hop (directly-connected) gateway. In most cases, negative advice from the data link layer means that the Address Resolution Process (ARP) processing performed by the data link layer was unable to resolve the location of first-hop gateway on the directly connected physical network. (ARP is not performed on all physical network types. Some physical network types, such as X.25, use an alternative scheme for this purpose.)

Negative advice, whether from TCP or the data link layer, is always expressed in terms of the first-hop gateway. Dead gateway processing on a given host only attempts to verify the first-hop gateway. However, gateways also carry out their own dead gateway processing for other adjacent gateways. In this way, all of the gateways along the path to a destination host are taken care of.

How IP Responds to Negative Advice

When receiving negative advice from TCP or the data link layer concerning a next hop gateway, IP marks all routes that use this gateway as suspect. IP attempts to deliver data destined for the suspect gateway via routes that use other gateways (if any are configured). Next, an IP process is started that uses periodic PING requests to attempt to contact the suspect next-hop gateway. If the suspect gateway continues to be unresponsive for an extended period of time, the frequency of the PING requests is reduced.

When any PING response is received from a suspect gateway, the gateway is considered active and the routes are restored.

Notes about IP Responses to Negative Advice:

1. If an ICMP redirect message is received during dead gateway processing, routes to a suspect gateway may be temporarily restored. However, dead gateway PING processing is not interrupted, and subsequent negative advice forces the IP routing table back to its previously adjusted state.
2. Responses from user-initiated PINGs can also indicate that a suspect gateway is active.
3. Negative advice is not passed from the UDP or RAW IP protocol machines. Applications using these protocols must use other mechanisms to detect and respond to apparent network problems. However, data link layer-negative advice is still used to manage problems with the first-hop gateway.

Multihoming Function

A multihomed host has multiple IP addresses, which we may think of as logical interfaces. These logical interfaces may be associated with one or more physical interfaces, and these physical interfaces may be connected to the same or different networks.

The AS/400 TCP/IP implementation supports multihoming. This allows you to specify either a single interface or multiple interfaces for a line description. You can have your AS/400 appear as any one or combination of the following scenarios:

- A single host on a network over a communications line
- Multiple hosts on the same network over the same communications line
- Multiple hosts on the same network over multiple communications lines
- Multiple hosts on different networks over the same communications line
- Multiple hosts on different networks over multiple communications lines

Note: The maximum number of interfaces that can be active on a line description at any given time is 128. This is true for all line types (for example, token-ring, Ethernet, frame relay, and so forth).

Example: A Single Host on a Network over a Communications Line

Your AS/400 system uses one adapter for TCP/IP to attach to a LAN or WAN network. You add one TCP/IP interface. This TCP/IP interface includes the Internet address of your AS/400 system. With this single Internet address, your AS/400 system is part of a single TCP/IP network (Figure 56).

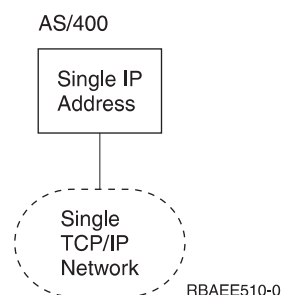


Figure 56. Multihoming - Single Host, Single Network, Single Line

Example: Multiple Hosts on the Same Network over the Same Communications Line

Your AS/400 system uses one adapter for TCP/IP to attach to a LAN or WAN network. You add multiple TCP/IP interfaces. Each of these TCP/IP interfaces includes an Internet address of the same TCP/IP network. With these multiple Internet addresses your AS/400 system appears as multiple hosts in a single TCP/IP network (Figure 57).

This can be a migration scenario.

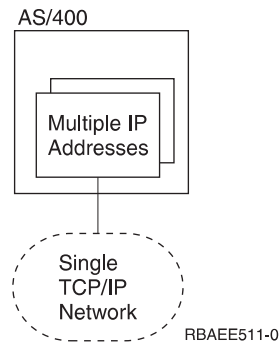


Figure 57. Multihoming - Multiple Hosts, Single Network, Single Line

Example: Multiple Hosts on the Same Network over Multiple Communications Lines

Your AS/400 system uses more than one adapter for TCP/IP to attach to the same LAN or WAN network. You add multiple TCP/IP interfaces. At least one interface is assigned to each adapter/line description. Each of these TCP/IP interfaces includes an Internet address of the same TCP/IP networks. With these multiple Internet addresses, your AS/400 system appears as multiple TCP/IP hosts in the same TCP/IP network (Figure 58).

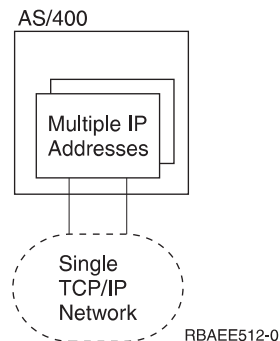


Figure 58. Multihoming - Multiple Hosts, Single Network, Multiple Lines

This scenario can be helpful for backup or to improve performance. However, there is no dynamic backup or performance balance function.

Example: Multiple Hosts on Different Networks over the Same Communications Line

Your AS/400 system uses one adapter for TCP/IP to attach to a LAN or WAN network. You add multiple TCP/IP interfaces. Each of these TCP/IP interfaces includes an Internet address of different TCP/IP networks. With these multiple Internet addresses, you participate in different TCP/IP networks (Figure 59).

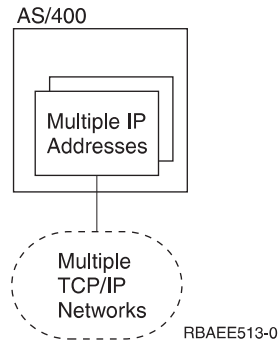


Figure 59. Multihoming - Multiple Hosts, Multiple Networks, Single Line

Imagine a public X.25 network. With this physical network, you can run multiple different TCP/IP networks, for example the company intranet, and connections with business partners and service providers. For each of these different TCP/IP networks, your AS/400 system must configure a unique Internet address.

Running multiple TCP/IP networks within a single local area network (LAN) is also supported. In most situations, however, one designs a single TCP/IP network per physical LAN only.

Example: Multiple Hosts on Different Networks over Multiple Communications Lines

Your AS/400 system uses more than one adapter for TCP/IP to attach to multiple LAN or WAN networks. You add multiple TCP/IP interfaces. At least one interface is assigned to each adapter/line description. Each of these TCP/IP interfaces includes an Internet address of different TCP/IP networks. With these multiple Internet addresses, you take part in different TCP/IP networks (Figure 60 on page 82).

This example is a combination of all of the previous examples discussed.

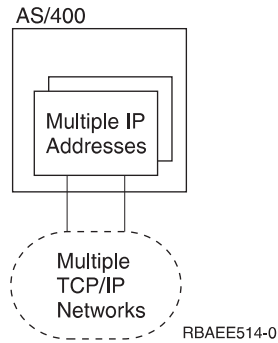


Figure 60. Multihoming - Multiple Hosts, Multiple Networks, Multiple Lines

Example: The Multihoming function

Assume AS/400 systems SYSNAM02 and SYSNAM03 are connected with a public or private X.25 network. The Internet address of this network is 9.4.73.64.

In this example, the AS/400 system SYSNAM03 connects with a service provider by using TCP/IP and the same X.25 network attachment (Figure 61). The Internet address assigned by the service provider for the AS/400 system is 223.1.1.17.

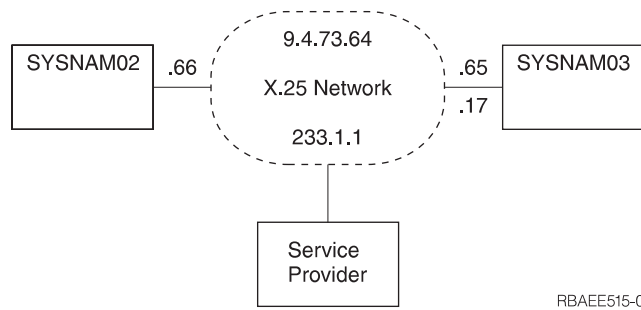


Figure 61. Multihoming TCP/IP Network

The multihoming function supports multiple networks with the same adapter. AS/400 system SYSNAM03 must handle two different Internet addresses on the same attachment. To do this, an additional TCP/IP interface needed to be specified (Figure 62 on page 83).

```

Work with TCP/IP Interfaces
System:  SYSNAM03
Type options, press Enter.
  1=Add  2=Change  4=Remove  5=Display  9=Start  10=End

  Internet      Subnet      Line      Line
  Opt  Address      Mask      Description  Type
  ---  -
  ---  9.4.73.65      255.255.255.192  X25LINE      *X25
  ---  127.0.0.1      255.0.0.0      *LOOPBACK    *NONE
  ---  223.1.1.17     255.255.255.0  X25LINE      *X25

F3=Exit      F5=Refresh  F6=Print list  F11=Display interface status
F12=Cancel   F17=Top     F18=Bottom

```

Figure 62. Work with TCP/IP Interfaces Display, Multihoming

Type of Service (TOS)

Type of Service (TOS) is a parameter defined to indicate a quality of the service desired by an application program. It is specified within a single octet of the IP datagram header, and it is used to select Internet service. It denotes how the Internet hosts and routers should make trade-offs between throughput, delay, reliability, and cost.

TOS is used to identify and select the actual transmission characteristics for a particular network, the interface, and the route to be used when routing an Internet datagram. The TOS values are mapped into the actual TOS value of the particular network a datagram is going through. All of the values are mutually exclusive.

The TOS values are defined through the Add TCP/IP Interface (ADDTCPIFC) and Add TCP/IP Route (ADDTCP RTE) commands. The possible selections are as follows:

***NORMAL**

Normal service is used for delivery of datagrams.

***MINDELAY**

Minimize delay means that prompt delivery is important for datagrams with this indication.

***MAXTHRPUT**

Maximize throughput means that high data rate is important for datagrams with this indication.

***MAXRLB**

Maximize reliability means that a higher level of effort to ensure delivery is important for datagrams with this indication.

***MINCOST**

Minimize monetary cost means that lower cost is important for datagrams with this indication.

The following table shows which type of services AS/400 uses for some of the TCP/IP applications:

Table 7. AS/400 TCP/IP applications and Type of Services

Protocol or Application	Type of Service Used
TELNET	Normal
FTP (control connection)	Minimize delay
FTP (data connection)	Maximize throughput
SMTP (command phase)	Minimize delay
SMTP (data phase)	Maximize throughput
POP (all phases)	Maximize throughput
SNMP	Maximize reliability

Thus, TOS is a suggestion, not a demand, to the interface (if more than one is present in the system) and to the routing algorithms. If a TCP/IP subsystem knows more than one interface and more than one possible route to a given destination, it uses the TOS to select one with characteristics closest to that desired.

TOS Example

For example, suppose the system can select between a low-capacity nonswitched line or a high-bandwidth (but high delay) satellite connection:

- Datagrams carrying keystrokes from a user to a remote computer could have the type of service set to *MINDELAY, requesting that they be delivered as quickly as possible.
- Datagrams carrying a bulk file transfer could have the type of service set to *MAXTHRPUT, requesting that they travel across the high-capacity satellite path.

It is up to the network administrator to define TOS values when defining interfaces and routes in the TCP/IP configuration. Based on the administrator's knowledge of the hardware technologies available on systems and networks used, TOS values for the routes must also be defined according to the interface's TOS value. This means that if a *MINDELAY value is defined in the interface definition, at least one route definition must have the *MINDELAY TOS value defined.

Note: A TCP/IP network does not guarantee the TOS requested. However, datagram transmission is never denied.

Multiple Routes

You can have multiple routes in your routing table (by using the ADDTCPRTE command). You can have more than one route for the same destination Internet address with the same type of service or a different type of service. If you have multiple routes with the same types of service, they are used in the order specified. If a particular next hop router is not available, the subsequent specified next hop router is used. This continues until an entry that is active is found or the list of next hop values is exhausted. If you have multiple routes with different TOS, the one with the TOS equal to the one requested by applications with TOS octet in IP datagram is used. If no match is found in any specified routes, the route with the closest TOS or *NORMAL TOS is used.

You can have *DFTRROUTE, and specific route destination addresses. Default routes are used only when data is sent to a remote destination system that does

not have a specific route defined. The system allows up to eight default routes, but each route must have a unique next hop value.

An example of a multiple route table can be found in Figure 63.

```

Work with TCP/IP Routes
System:  SYSNAM003
Type options, press Enter.
  1=Add  2=Change  4=Remove  5=Display

Opt  Route      Subnet      Next      Preferred
     Destination  Mask        Hop        Interface
-----
-   *DFTRROUTE  *NONE      9.4.73.193 *NONE
-   *DFTRROUTE  *NONE      9.4.73.197 *NONE
-   *DFTRROUTE  *NONE      9.4.73.196 *NONE
-   9.4.70.0    255.255.255.0 9.4.73.194 *NONE
-   9.4.70.0    255.255.255.0 9.4.73.195 *NONE
-   9.4.70.0    255.255.255.0 9.4.73.198 *NONE

Bottom
F3=Exit  F5=Refresh  F6=Print list  F10=Work with IP over SNA routes
F11= Display type of service  F12=Cancel  F17=Top  F18=Bottom

```

Figure 63. Work with TCP/IP Routes Display

TCP/IP Port Restriction

TCP and UDP protocols use **ports** to identify a unique origin or destination of communication with an application. Each port is assigned a small integer. You can configure port information if you want to restrict the use of a TCP or UDP port to one or more user IDs.

The range of port numbers is from 1 to 65535. However, ports 0-1023 are reserved as well-known port numbers, which are controlled and assigned by the Internet Assigned Numbers Authority (IANA). Only those applications that have been assigned one of these ports should use a number within this range. Refer to the current Assigned Numbers RFC for a list of the port assignments.

Because this range of port numbers, 0-1023, is reserved for the well-known ports, they should not be used by user application programs because it could affect the operation of TCP/IP. For example, restricting the use of ports 21, 23, or 25, prevents other users from using FTP, TELNET, or SMTP, respectively.

The AS/400 Add TCP/IP Port Restriction (ADDTCPPORT) command allows you to restrict usage of a single port or a range of ports to a particular AS/400 user profile.

Restricting ports is like allocating ports to a specific user profile. When a socket application issues the bind() system call, or when a TCP/UDP Pascal API application issues a call to the TcpOpen, TcpWaitOpen, or UdpOpen function, the job's user profile is checked against the list of user profiles that are associated with

the specified port. If no match is found, the requesting program is not allowed to use the specified port. If any port in the 1-1023 range is restricted, the following message is posted:

Port restriction added but may affect TCP/IP processing

If no user profiles are associated with a specific port, there are no restrictions.

It is not necessary to configure port restrictions unless you are writing your own TCP/IP applications and you want to reserve the use of the applications to certain user profiles.

Note: For an installation in which user-written programs use ports other than the well-known ports, you can consider restricting the use of the well-known ports to the user profiles running the server application. As an example, for File Transfer Protocol (FTP), this would be user profile QTCP.

Configuring TCP/IP Port Restrictions

To configure TCP/IP port restrictions, type option 4 on the Configure TCP/IP menu. The Work with TCP/IP Port Restrictions display is shown (Figure 64).

```
Work with TCP/IP Port Restrictions                               System:  SYSNAM03
Type options, press Enter.
 1=Add  4=Remove

  Opt  --Port Range---  Protocol  User Profile
      Lower  Upper  *ONLY
  -    1050   1059   *TCP      PAOLO

F3=Exit  F5=Refresh  F6=Print list  F12=Cancel  F17=Top  F18=Bottom
Bottom
```

Figure 64. Work with TCP/IP Port Restrictions Display

Type option 1 (Add) at the input-capable top list entry to get to the Add TCP/IP Port Entry (ADDTCPPORT) display shown in Figure 65 on page 87. You can go directly to this display by typing ADDTCPPORT on any command line and pressing F4.

```

Add TCP/IP Port Restriction (ADDTCPPORT)

Type choices, press Enter.

Range of port values:
  Lower value . . . . . 1060          1-65535
  Upper value . . . . . > *ONLY      1-65535, *ONLY
  Protocol . . . . . *tcp             *UDP, *TCP
  User profile . . . . . gerry        Character value

F3=Exit  F4=Prompt  F5=Refresh  F12=Cancel  F13=How to use this display
F24=More keys
Bottom

```

Figure 65. Add TCP/IP Port Restriction Display

Let us assume we have an application that uses Port 1060 in the TCP layer and we want to restrict its use to user profile GERRY. Type the information as shown in Figure 65.

Figure 66 shows what the display looks like after you enter port information for both user profiles PAOLO and GERRY.

Changes to the port restrictions take effect immediately. However, applications that are already active are not affected until they are restarted.

```

Work with TCP/IP Port Restrictions
System:  SYSNAM03

Type options, press Enter.
  1=Add  4=Remove

Opt  --Port Range---
     Lower  Upper  Protocol  User
     -----
-    1050   1059   *TCP     PAOLO
     1060   *ONLY  *TCP     GERRY

F3=Exit  F5=Refresh  F6=Print list  F12=Cancel  F17=Top  F18=Bottom
Bottom

```

Figure 66. Work with TCP/IP Port Restrictions Display

Related Tables and the Host Table

Socket applications require a set of tables from which they can retrieve specific TCP/IP network data when needed. These are as follows:

- Host table
- Service table
- Protocol table
- Network table

The host table contains a list of host names and corresponding Internet addresses. Socket applications requesting host data obtain it either from the AS/400 host database file or from the domain name server.

The service table contains a list of services and the specific port and protocol a services uses. The protocol table contains a list of protocols used in the TCP/IP network. The network table contains a list of networks and the corresponding Internet addresses.

UNIX** systems traditionally store this information in the following files:

- /etc/hosts - host table
- /etc/protocols - protocol table
- /etc/services - service table
- /etc/networks - network table

AS/400 TCP/IP maintains the service, protocol, and network tables as database files. AS/400 TCP/IP refers to these three tables as related tables. To configure or view the protocol, services, or network tables, select option 21 (Configure Related Tables) on the Configure TCP/IP menu. You are shown the display in Figure 67 .

```

                                Configure Related Tables
                                System:  SYSNAM03

Select one of the following:

    1. Work with service table entry
    2. Work with protocol table entry
    3. Work with network table entry

Selection or command
===> _____
F3=Exit  F4=Prompt  F9=Retrieve  F12=Cancel
```

Figure 67. Configure Related Tables Menu

You can change the services, protocols, and network files using the options from this display.

The services table stores the mapping of services to ports or ports to services as shown in Figure 68. The mapping information is usually accessed with the `getservbyname()` and `getservbyport()` socket functions.

```
Work with Service Table Entry                               System:  SYSNAM03
Type options, press Enter.
 1=Add  4=Remove  5=Display

Opt  Service                                     Port  Protocol
-----
     echo                                       7    udp
     finger                                    79    tcp
     finger                                    79    udp
     ftp-control                               21    tcp
     ftp-control                               21    udp
     ftp-data                                  20    tcp
     ftp-data                                  20    udp
     gopher                                    70    tcp
     gopher                                    70    udp
     graphics                                  41    tcp
     graphics                                  41    udp
     pop3                                       110   tcp

Parameters for options 1 and 4 or command
===>
F3=Exit  F4=Prompt  F5=Refresh  F6=Print list  F9=Retrieve  F12=Can
F17=Top  F18=Bottom

More...
```

Figure 68. Work with Service Table Entry Display

The protocol table stores the mapping of protocol names to protocol numbers and protocol numbers to protocol names. Socket applications use `getprotobyname()` and `getprotobynumber()` functions to access this table (Figure 69 on page 90).

```

Work with Protocol Table Entry
System:  SYSNAM03
Type options, press Enter.
  1=Add  4=Remove  5=Display

Opt      Protocol
-----
-        icmp          1
-        ip            0
-        tcp           6
-        udp           17

Parameters for options 1 and 4 or command
===>
F3=Exit  F4=Prompt  F5=Refresh  F6=Print list  F9=Retrieve  F12=Cancel
F17=Top  F18=Bottom
Bottom

```

Figure 69. Work with Protocol Table Entry Display

The network table contains the networks and the Internet address associated with the network. Socket applications use the `getnetbyname()` and `getnetbyaddr()` functions to access the information in the network table (Figure 70).

```

Work with Network Table Entry
System:  SYSNAM03
Type options, press Enter.
  1=Add  4=Remove  5=Display

Opt      Network
-----
-        IBM          9.0.0.0

Parameters for options 1 and 4 or command
===>
F3=Exit  F4=Prompt  F5=Refresh  F6=Print list  F9=Retrieve  F12=Cancel
F17=Top  F18=Bottom
Bottom

```

Figure 70. Work with Network Table Entry Display

The protocols and services tables that are shipped contain standard information. The network tables do not contain any information. The network IBM information has been added in Figure 70, as an example.

For additional information about sockets, refer to the *Sockets Programming*, SC41-5422-03 book.

Using X.25 PVC instead of SVC

In “Step 5—Configuring TCP/IP Remote System Information (X.25)” on page 36 you were shown how to define the X.25 network address of each system that uses a switched virtual circuit (SVC).

To replace the X.25 SVC with an X.25 permanent virtual circuit (PVC) connection, the example below is helpful. The following CL commands will look different: CRTLINX25, ADDTCPIFC, and ADDTCPRSI.

Use the same X.25 line description, but replace the first of the four SVCs with a PVC.

```
CRTLINX25 LIND(X25LINE) RSRNAME(LIN051)
LGLCHLE((001 *PVC) (002 *SVCBOTH)
(003 *SVCBOTH) (004 *SVCBOTH))
NETADR(40030003) CNNINIT(*LOCAL)
TEXT('ITSO X.25 Network')
```

The TCP/IP interface now points to a specific PVC instead of a pool of SVCs.

```
ADDTCPIFC INTNETADR('9.4.73.65') LIND(X25LINE)
SUBNETMASK('255.255.255.192') PVCLGLCHLI(001)
MAXSVC(0)
```

The TCP/IP remote system information no longer includes the X.25 address to be called. Instead, the entry points to the PVC channel ID.

```
ADDTCPRSI INTNETADR('9.4.73.66')
PVCLGLCHLI(001)
```

IP Multicasting

IP multicasting is the process of transmitting an IP datagram to a host group. The hosts that are in the group may reside on a single subnet or on different subnets that are connected by multicast-capable routers. Hosts may join and leave groups at any time. There are no restrictions on the location or number of members in a host group. For more information about IP multicasting, refer to RFC 1112, *Host Extensions for IP Multicasting*.

Note: The AS/400 cannot act as a multicast-capable router.

Multicast Application Programming Information

An application program can send or receive multicast datagrams by using the Sockets API and connectionless, SOCK_DGRAM type sockets. Multicasting is a one-to-many transmission method. You cannot use connection-oriented sockets of type SOCK_STREAM for multicasting. When a socket of type SOCK_DGRAM is created, an application can use the setsockopt() function to control the multicast characteristics associated with that socket. The setsockopt() function accepts the following IPPROTO_IP level flags:

- IP_ADD_MEMBERSHIP: Joins the multicast group specified.

- IP_DROP_MEMBERSHIP: Leaves the multicast group specified.
- IP_MULTICAST_IF: Sets the interface over which outgoing multicast datagrams should be sent.
- IP_MULTICAST_TTL: Sets the time to live (TTL) in the IP header for outgoing multicast datagrams.
- IP_MULTICAST_LOOP: Specifies whether or not a copy of an outgoing multicast datagram should be delivered to the sending host as long as it is a member of the multicast group.

For additional information about sockets, including sample programs, see the *Sockets Programming*, SC41-5422-03 book. The *System API Reference*, SC41-5801-03 documents the sockets API.

Multicast Restrictions

Multicast does not map well to all types of physical lines. For this reason, it is not supported on all lines. For example, a switched network such as X.25 does not lend itself to multicast applications because no mechanism exists for transmitting a single packet to all systems in the network that have joined a group. IP multicast is supported on broadcast capable networks and on SLIP/PPP interfaces, but it is not supported on multi-access nonbroadcast networks. IP multicast is also not currently supported on Frame Relay, FDDI/SDDI, or ATM networks. To determine whether an interface supports multicast, enter option 14 on the Work with TCP/IP Interface Status display. If the interface supports multicast, there will be at least one Host Group entry for the All Hosts group 224.0.0.1. Otherwise, the interface does not support multicast.

The 2626 token-ring input-output processor (IOP) requires manual configuration to receive multicast datagrams. In particular, you must specify the token-ring address, C00000040000, on the functional address parameter for the token-ring line description. To add this address to a line description that is named TRNLINE, use the following command:

```
CHGLINTRN LIND(TRNLINE) FCNADR(C00000040000)
```

The 2617 Ethernet IOP also requires manual configuration in order to receive multicast datagrams. The Ethernet group addresses to be received need to be specified on the group address parameter (GRPADR) for the Ethernet line description. A 4-byte IP multicast address is mapped to a 6-byte Ethernet group address by placing the low-order 23 bits of the IP multicast address into the low-order 23 bits of the Ethernet group address 01005E000000. For example, to receive multicast datagrams with a destination address of 224.255.0.2, the GRPADR parameter for the 2617 Ethernet line description must include 01005E7F0002.

Chapter 4. Configuring Point-to-Point TCP/IP (PPP and SLIP)

Important note: A thorough and in-depth explanation of Point-to-Point is beyond the scope and purpose of this document. The majority of material on Point-to-Point is covered in the **AS/400e Information Center** under the **TCP/IP** topic. For more information see "TCP/IP Topics in the Information Center" on page xv.

This chapter focuses on conceptual and reference information for the TCP/IP Point-to-Point Protocol (PPP) and briefly describes its predecessor, the Serial Line Internet Protocol (SLIP). This chapter does not document the procedures for AS/400 configuration and implementation of the PPP protocol. Those procedures are covered in the **AS/400e Information Center** under the **TCP/IP** topic.

Networks and Point-to-Point Connections

When only two systems are physically connected, this is typically referred to as a point-to-point connection or link. Several different protocols, such as PPP, SLIP, X.25, and frame relay, can be viewed as point-to-point protocols. Support for Point-to-Point (PPP) protocol is included in AS/400 as part of wide area network (WAN) connectivity. A common example is a PPP connection that is established periodically over a phone line from a remote office to a central office in order to exchange data between the locations. This connection could be from a laptop computer. Remote systems can access AS/400 applications such as Lotus Notes over a PPP TCP/IP link. Figure 71 illustrates AS/400 responding to incoming calls.

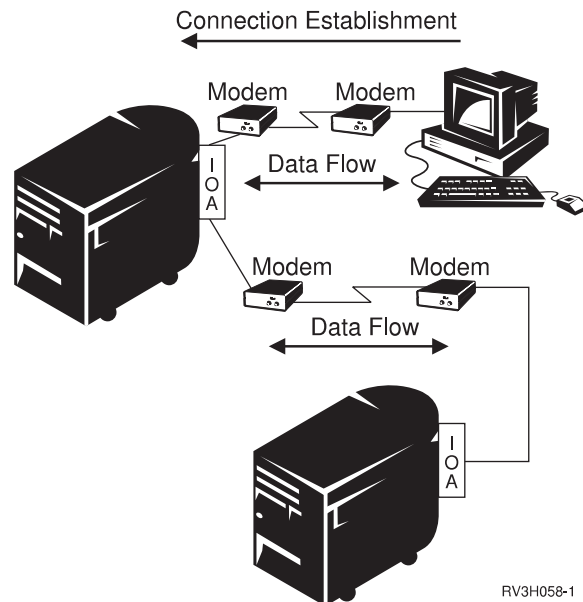


Figure 71. Remote Systems Dial-in to an AS/400

Another common example of a PPP WAN connection is a dial-out connection that your system establishes to an Internet Service Provider (ISP), such as the IBM Global Network (IGN) (see Figure 72 on page 94.)

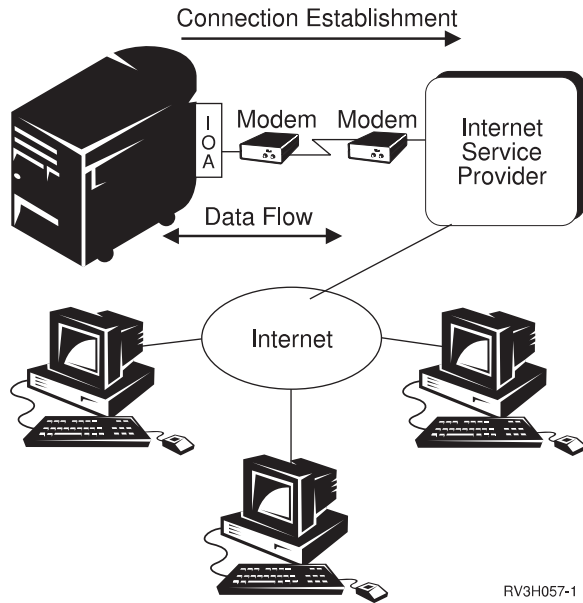


Figure 72. AS/400 Dial-out to an ISP

A wide area network (WAN) is usually distinguished from a local area network (LAN) in that physical communications is limited to two systems. The communications line between the two systems can be a direct connection such as a leased telephone line. However, it is more common to connect two systems by a dial-up telephone connection. A dial-up connection between systems is also known as a switched connection. For a switched line, the connection turns on or off depending on whether a phone connection has been established. In the case of a leased line, the connection between the two systems is always available.

PPP versus SLIP

Serial Line Internet Protocol (SLIP) is the result of early attempts to connect two systems using TCP/IP over an asynchronous line. SLIP is described in Request for Comment (RFC) 1055, *A Nonstandard For Transmission of IP Datagrams Over Serial Lines: SLIP*. This RFC defines a simple framing method for IP packets flowing over a serial line. SLIP is not an Internet standard.

The SLIP RFC never became an Internet standard because it has several deficiencies that are discussed in the RFC. Some of those deficiencies are as follows:

- No standardized mechanism for hosts to communicate addressing information
- No support for network protocols other than TCP/IP
- No support for system authentication
- No support for packet error detection, error correction, or compression

While SLIP is still used today, IBM does not encourage you to use it. PPP corrects all of the SLIP deficiencies. PPP is an Internet standard and the predominant connection protocol used today among Internet Service Providers.

One goal of PPP is to allow interoperability among the remote access software of different manufacturers. Another goal is to allow the same physical communication line to be used by multiple network communication protocols.

Requirements for AS/400 SLIP

You need two or more computers that support the SLIP protocol. The following is a summary of AS/400 requirements for Serial Line Internet Protocol (SLIP):

- The correct communications ports and adapters must be installed on AS/400.
- A connection must be established using either a switched line or a direct leased line.
- A modem to send and receive data over the connection. “Connection Alternatives” on page 111 contains a brief overview of data transmission equipment available at the time of this writing.
- If you plan to connect to the Internet, you must have a dial-up account with an Internet Service Provider (ISP).

Note: Many PCs have an internal modem that is equipped with the serial port. AS/400 does not support an internal modem. You must use an external modem connected to your AS/400 by one of the required I/O Adapters (IOA).

Point-to-Point Request for Comments (RFC)

A sample of related Request for Comments (RFC) are:

- RFC 1661, *The Point-to-Point Protocol (PPP)*, describes the base structure for PPP communications.
- RFC 1662, *PPP in HDPC-like Framing*, describes PPP packet encapsulation over asynchronous and synchronous communication lines.
- RFC 1994, *PPP Challenge Handshake Authentication Protocol (CHAP)*, describes system authentication.

Line Pools

In the Point-to-Point Protocol (PPP), a *line pool* is a list of lines that are used by a connection profile.

You can use a line pool instead of defining a particular line description to a connection profile. The connection profile selects an available line from the line pool when the profile is started.

Advantages of using line pools:

- You are not committing a line resource to a connection profile until it is started. For connection profiles using a specific line, the connection profile ends if the line is not available. For connection profiles that use a line pool, only one line in the line pool must be available when the profile is started.
- You can use dial-on-demand profiles with line pools to use resources more efficiently.

A line is selected from the line pool only when a dial-on-demand connection is required. At other times, the same line can be used by other connection profiles.

- You can start more profiles than you have resource to support.

Let us say your environment needs four dial-on-demand connection profiles, but you only need two lines available at one time. You could create a line pool for two of the four lines, so two connections profiles are active at any time. By using a line pool, you do not need to have four lines available at the same time.

Configuring Point-to-Point Network Connections

This material is covered in the **AS/400e Information Center** under the **TCP/IP** topic. For more information see “TCP/IP Topics in the Information Center” on page xv.

Configuring PPP Connection Profiles

This material is covered in the **AS/400e Information Center** under the **TCP/IP** topic. For more information see “TCP/IP Topics in the Information Center” on page xv.

Accessing Point-to-Point functions through Operations Navigator

You can access Point-to-Point (PPP) functions from a command line interface or Operations Navigator (graphical user interface). Not all PPP functions are available on both interfaces, and most PPP functions are available only through Operations Navigator. Operations Navigator functions are not documented here.

To access the **Point-to-Point Profile Properties** dialogs, perform the following tasks:

1. Start Operations Navigator.
2. Double-click your AS/400 server in the main tree view of Operations Navigator.
3. Double-click **Network**.
4. Double-click **Point-to-Point**.
5. Right-click **Connection Profiles** to open a context menu.
6. Select **New Profile** to open the New Point-to-Point Profile Properties dialog. Click the desired tab to define properties.

Checking for existing PPP Connection Profiles

This material is covered in the **AS/400e Information Center** under the **TCP/IP** topic. For more information see “TCP/IP Topics in the Information Center” on page xv.

PPP Configuration Scenarios

Example: Configuring Windows 95/98 to an AS/400 using a PPP Connection

This material is covered in the **AS/400e Information Center** under the **TCP/IP** topic. For more information see "TCP/IP Topics in the Information Center" on page xv.

Example: Connecting to the Internet using an ISP

You can connect an AS/400 to the Internet by using an Internet Service Provider (ISP). While there are many ISPs in service today, this example uses the IBM Global Network (IGN) as a service provider. IGN requires an account, user ID, and password. Other Internet Service Providers (ISP) may have different requirements. The connection to the ISP is switched line-dial.

Access Point-to-Point functions through Operations Navigator.

From the **General** page, perform the following tasks:

- Specify DialIGN in the "Name" text box.
- Specify IGN (IGN-ISP) in the "Description" text box to describe the connection profile in more detail.
- Select PPP under Type.
- Select Switched line-dial under Mode. This allows you to dial out to the remote system.

Click the **Connection** tab.

- To specify the remote phone number for the IGN or another ISP, click **Add** and specify the phone number.

If you need to dial a special number to reach an outside line, typically the outside line number is followed by several commas. The commas determine how many seconds the modem must wait for the dial tone before it begins dialing. You can also use dashes in the phone number.

A remote phone number can be any of the following:

- Local company extension (for example, 34567)
- Outside phone number (for example, 9,1234567)
- Long distance number (for example, 1-800-1234567)
- In the "Name" field, select the appropriate line name, such as IGN_Dial.
If you need to configure a new line description, specify the name and click **Open**. For more information, see "Configuring Point-to-Point Network Connections" on page 96.
- You can disconnect from the ISP after a certain period of time by overriding the line inactivity timeout. To do this, select "Override line activity timeout" and specify the "Timeout" value in seconds. In this way, you will not be charged by the ISP for idle time.

Click the **TCP/IP Settings** tab.

- Select Dynamically assign to indicate that you want IGN to dynamically assign both the local and remote IP addresses.

- Click **Routing** to open the **Routing** dialog.
- For static routing, specify Add remote system as the default route. Since you do not know the range of destination IP addresses, you need a default route. This is the desired configuration when AS/400 dials in to an ISP such as IGN. The default route forwards packets to the ISP router, which in turn routes the packets throughout the Internet.

Click the **Authentication** tab.

- Check "Enable local system identification." IGN requires customers to use the Password Authentication Protocol (PAP) for authentication. This verifies the identity of the customer who is using an unencrypted password.
- Specify both a user ID and a password. In this example, the user name is internet.account.userid. Other ISPs may use CHAP, which uses an encrypted password. Consult your ISP to learn its requirements for specifying a user name and password.

Click the **Domain Name Server** tab.

- Specify the IP address for the Domain Name Server. In this example, the IP address is 10.11.27.5. When you use an Internet Service Provider (ISP) to connect to the Internet, you will receive a name server address from the ISP. This allows you to connect using host names instead of IP addresses for remote sites. A host name is a way of identifying a system and its IP address.

Example: Connecting two AS/400s using dial-on-demand

In this example, System 1 is a local system with IP address 10.11.25.1 and System 2 is a remote system with IP address 10.11.25.2. To establish a connection:

1. Configure a Dial-only connection profile on System 1.
2. Configure a Switched line-answer connection profile on System 2.
3. Start connection profiles on System 1 and System 2 by using Operations Navigator.
4. Verify configurations on System 1 and System 2 by using TELNET.

Configure a Dial-only connection profile on System 1

1. Access Point-to-Point functions through Operations Navigator.
2. From **New Point-to-Point Profile Properties** dialog, click the **General** tab and do the following:
 - Type **DODBP** in the **Name** text box.
 - Type Basic Dial on Demand Profile in the **Description** text box.
 - Under **Type**, select **PPP**.
 - Under **Mode**, select **Dial-on-demand (dial only)**.
3. Click the **Connection** tab and do the following:
 - Type the phone number of the remote system.
 - Under **Line**, select a line pool from the **Name** list.
 - Select **Override line activity timeout** and specify 300 seconds, or five minutes, as the **Timeout** value.
4. Click the **TCP/IP Settings** tab and then do the following:
 - Under **Local IP Address**, click the radio button and type 10.11.25.1 in the text box.
 - Under **Remote IP Address**, the Remote IP Address is 10.11.25.2.

- Check **Allow IP forwarding**.
5. Click the **Authentication** tab and then ensure that local and remote system identification are not checked.
 6. Click **OK** to save the configuration.

Configure a Switched line-answer connection profile on System 2

1. Access Point-to-Point functions through Operations Navigator.
2. From **New Point-to-Point Profile Properties** dialog, click the **General** tab and then do the following:
 - Type **DODANS** in the **Name** text box.
 - Type Basic Answer Profile in the **Description** text box.
 - Under **Type**, select **PPP**.
 - Under **Mode**, select **Switched line-answer**.
3. Click the **Connection** tab and then select a line description from the **Name** list.
4. Click the **TCP/IP Settings** tab and do the following:
 - Under **Local IP Address**, click the radio button and type 10.11.25.2 in the text box.
 - Under **Remote IP Address**, the Remote IP Address is 10.11.25.1.
 - Check **Allow IP forwarding**.
5. Click the **Authentication** tab and then ensure that local and remote system identification are not checked.
6. Click **OK** to save the configuration.

Start connection profiles on System 1 and System 2

Once you have started the connection profiles, each connection profile has the following status:

- System 1: DODBP connection profile: Waiting for dial-Switched line-dial on demand
- System 2: DODANS connection profile: Waiting for incoming call

Verify configurations on System 1 and System 2 using TELNET

Use TELNET to verify that you can dial and connect to System 2 from System 1 using the DODBP and DODANS connection profiles.

- From the command line on System 1, type

```
TELNET '10.11.25.2'
```

and press **Enter**.

The AS/400 logon screen on System 2 displays if the connection is successful.

The TELNET session ends after 5 minutes of inactivity. Both connection profiles are reset and wait for the next call.

While the DODBP profile is in the *Waiting for dial — Switched line-dial on demand* status, System 1 will dial System 2 when IP packets arrive for System 2.

Example: AS/400 Office-to-Office Scenarios

In this scenario, AS400 (LCL400) is set up to answer incoming calls from one of the following:

- Another AS400 (RMT400), along with its Ethernet and token-ring network
- A remote gateway, along with the Ethernet network attached to it
- An individual remote user

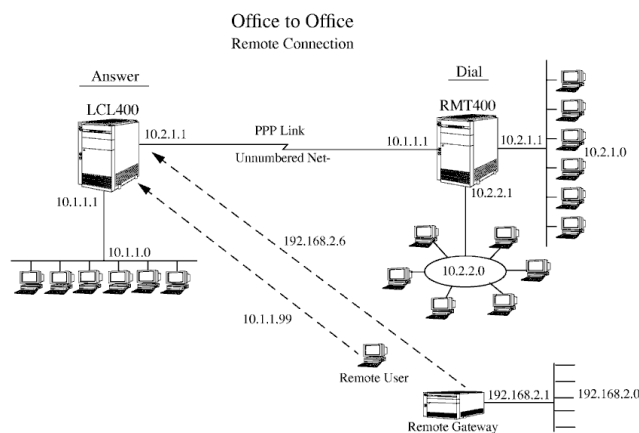


Figure 73. Office-to-Office Remote Connection

This can be accomplished using one connection, such as a phone line with an attached modem, and only one point-to-point connection profile on LCL400. This is done by configuring LCL400 to administer different remote IP addresses and routes based on the incoming caller's user ID. Since the user ID is used, the remote system dialing in must use some sort of authentication. For SLIP, this would be done with a connection script.

This example uses a PPP connection using CHAP authentication. This authentication protocol was chosen to ensure that only encrypted authentication information is passed between the systems.

You can configure LCL400 to accept calls from many other user IDs as well, giving it the flexibility of having tailored IP address and routes for each caller's environment. Figure 73 shows an example of this configuration.

Scenario Definitions

Office-to-Office Connection (RMT400 - LCL400)

RMT400 is a remote office with a Token Ring and Ethernet network attached to it. Either periodically or for a specified time period during the day, RMT400 dials into LCL400 using a PPP connection. The users in the remote office need to be able to access LCL400 as well as the 10.1.1.0 Ethernet network attached to LCL400. Conversely, the users on the 10.1.1.0 network need access to the 10.2.1.0 and 10.2.2.0 networks attached to RMT400. Information about how to configure both LCL400 and RMT400 to accomplish this task is covered shortly.

Remote User to Office (LCL400)

A remote user could be any user who dials into LCL400 from a PC or workstation. For example, a remote user is someone who is travelling and wishes to connect to the home office. The user dialing in also needs access to the 10.1.1.0 Ethernet network that is attached to LCL400.

Remote Network to Office (LCL400)

Remote Gateway (non-AS400) is a gateway that is attached to an Ethernet network. Remote Gateway could call LCL400 during the night to allow systems on the Remote Gateway's Ethernet network (192.68.2.0) to transfer files to either LCL400 or its attached network. Remote Gateway also allows the forwarding of mail to a mail server on the 192.68.2.0 network.

Configuring LCL400 for all Scenarios

Access Point-to-Point functions through Operations Navigator.

From the **General** page, perform the following tasks:

- Specify a profile name (LCL400 for this example).
- Select PPP as the "Type."
- Specify Switched line-answer as the "Mode."
- (Optional) Specify a description.

From the **Connection** page, perform the following tasks:

- Select an existing PPP line from the drop-down box and click **Open**. In this example, the line 400ANSWER is used.

If an existing line does not exist, you can create a new one by specifying the name and clicking **New**.

- From the **Modem** page on the **Line Properties** dialog, select a modem from the modem name pull-down box and click **OK**.

From the **Authentication** page, perform the following tasks:

- Select "Require remote system identification."
- Ensure that CHAP only is selected.
- Select an existing validation list from the validation list drop-down box and click **Open**. In this example, the validation list OFFICE_400 is used.

If a new validation list is required, create a new one by specifying the name and clicking **New**.

Figure 74 shows the **Validation List** dialog.

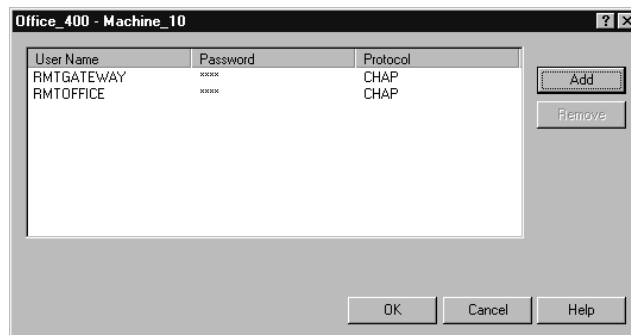


Figure 74. Validation List

The following values hold true for this example of the **Validation List** dialog:

- The user IDs RMTOFFICE and RMTGATEWAY have been added for CHAP authentication.
- RMTOFFICE is the user ID for RMT400 to dial in to LCL400.

- RMTGATEWAY is the user ID that is used by the remote gateway to dial in to LCL400.

From the **TCP/IP Settings** page, perform the following tasks:

- Set your local IP address. In this example, the existing Ethernet interface 10.1.1.1 is selected from the "Local IP address" pull-down box.
- Set your remote IP address. In this example, "Route specified" is selected. This signifies that remote IP addresses will be defined from entries that are defined from the **Routing** dialog.

Note: "Route specified" is valid only for Switched Line-Answer profiles.

- Check "Allow IP forwarding." IP packets that originate from the remote system will be allowed to flow through LCL400 to the 10.1.1.0 network.
- Click **Routing** to add the "Route specified" IP addresses.

Figure 75 shows the **Routing** dialog.

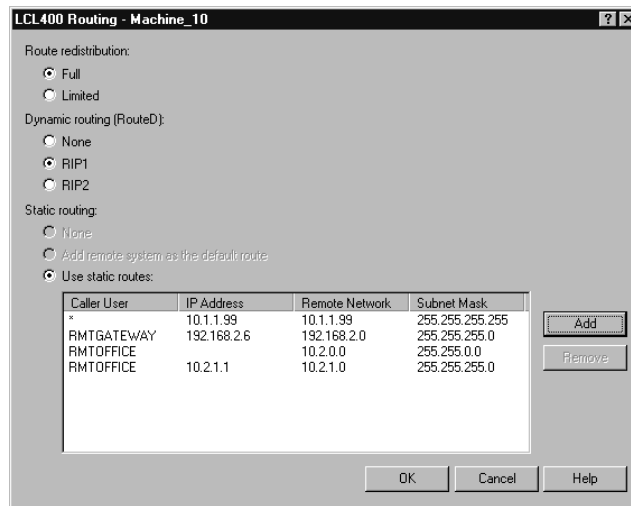


Figure 75. Routing

The following values hold true for this example of the **Routing** dialog:

- RMTGATEWAY user ID (from the Remote Gateway)
When user RMTGATEWAY dials in, he or she receives IP address 192.168.2.6. A subnet mask of 255.255.255.0 was used to allow LCL400 to add a direct route to the 192.168.2.0 network that is attached to the Remote Gateway.
- RMTOFFICE user ID (from system RMT400)
When user RMTOFFICE dials in, he or she receives IP address 10.2.1.1. A subnet mask of 255.255.255.0 was used to allow LCL400 to add a direct route to the 10.2.1.0 network that is attached to RMT400.
A route was defined for RMTOFFICE. The destination network is 10.2.0.0 with a subnet mask of 255.255.0.0. This route is added when RMTOFFICE dials in and allows LCL400 to have access to any 10.2.x.x network that is attached to RMT400. In this scenario, this would include the 10.2.1.0 network, which is also covered by the direct route that was added for the interface, as well as the 10.2.2.0 network. If any other 10.2.x.x networks are added to RMT400, the necessary route to reach them is already defined.

In this scenario, the IP addresses that are used for the PPP connection allow for what is known as an 'Unnumbered Net.' It is called this because no new networks or IP addresses were created for the connection. The IP address of the Ethernet connection to LCL400 (10.1.1.1) is used as the remote address for RMT400. The IP address of the Ethernet connection to RMT400 (10.2.1.1) is used as the remote address for LCL400.

- If a user other than RMTOFFICE or RMTGATEWAY dials in, then he or she receives IP address 10.1.1.99. This is accomplished by using the wildcard user entry * (asterisk), which indicates that if the caller's user ID is not found in the list, then this IP address is used as the default. This action assumes the user has been previously added to the OFFICE_400 validation list.

In the scenario, the remote user receives 10.1.1.99 as the IP address. He or she also has access to the 10.1.1.0 network with no additional routing required.

Note: 10.1.1.99 has a subnet mask of 255.255.255.255 and is a subset of the local 10.1.1.0 network. This is known as "transparent subnetting" (also known as Proxy ARP) and allows for the remote user to appear to be on the same Ethernet network that is attached to LCL400. See the example in "Example: Remote LAN Access with Transparent Subnetting" on page 104 for additional information about transparent subnetting.

If only explicitly defined callers can dial in, then you should omit the * entry. All other callers are then denied access to LCL400.

Configuring RMT400 to Dial into LCL400

Access Point-to-Point functions through Operations Navigator.

From the **General** page, perform the following tasks:

- Specify a profile name (RMT400 for this example).
- Select PPP as the "Type."
- Specify Switched line-dial as the "Mode."
- You have the option of specifying a description.

From the **Connection** page, perform the following tasks:

- In this example, the phone number to connect to LCL400 is 798-1234. The - (dash) is optional.
- Select an existing PPP line from the drop-down box and click **Open**. In this example, the line 400DIAL is used.

If an existing line does not exist, you can create a new one by specifying the name and clicking **New**.

- From the **Modem** page on the **Line Properties** dialog, select a modem from the modem name pull-down box and click **OK**.

From the **Authentication** page, perform the following tasks:

- Select "Enable local system identification."
- Ensure that CHAP only is selected.
- The user ID that will be used to identify RMT400 when it dials a remote system is RMTOFFICE.

Note: The user ID and password *must* be the same as those that you defined in the OFFICE_400 validation list on LCL400.

From the **TCP/IP Settings** page, perform the following tasks:

- Set your local IP address. In this example, the existing Ethernet interface 10.2.1.1 is selected from the "Local IP address" pull-down box.
- Set your remote IP address. In this example, "Dynamically assigned" is selected. The remote system, LCL400, defines the address that RMT440 uses when the connection is established.

The remote address of 10.1.1.1, which is the local Ethernet IP address for LCL400, could have been defined statically for RMT400. If this address ever changes, however, you must update the profile. The value "Dynamically assigned" allows it to work for whatever address LCL400 has specified.

- Since you need to define additional routes, click **Routing**.

Figure 76 shows the **Routing** dialog.

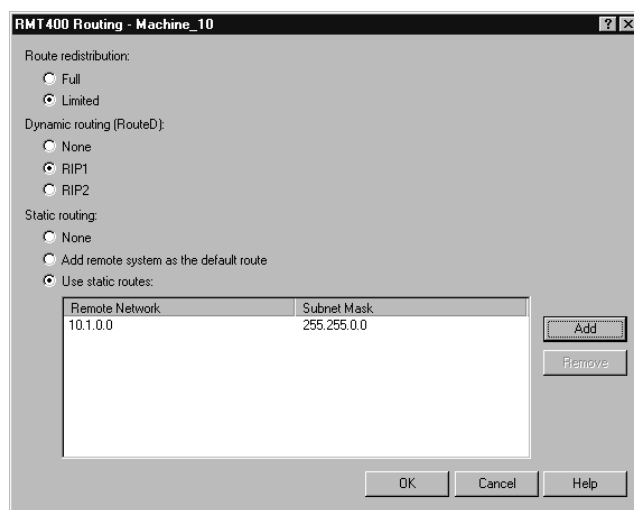


Figure 76. Routing

The following values hold true for this example of the **Routing** dialog:

- Select "Use static routes" Static routing. If RMT400 only ever called LCL400, then you could have selected "Add remote system as the default route." This ensures that *all* TCP/IP traffic would go to LCL400 if not otherwise defined. In this example, however, assume that RMT400 has other point-to-point profiles that can call other remote systems at the same time. Therefore, we want to define network routes to the remote networks.
- A route is defined for remote network 10.1.0.0 with a subnet mask of 255.255.0.0. This route is added when the connection is made with LCL400. All TCP/IP traffic for hosts on the 10.1.x.x network are sent to LCL400. Currently, this is only 10.1.1.0. If other 10.1.x.x networks are added to LCL400 in the future, then no further routing will need to be defined to reach hosts on the other networks.

Example: Remote LAN Access with Transparent Subnetting

Transparent Subnetworking allows remote clients who are on separate LANs to communicate with one another as if they were on the same physical network. To accomplish transparent subnetworking from the home network, you must partition blocks of addresses for each remote site. The subnet and the host are the two levels of hierarchical addressing. The boundary between them is arbitrary.

For example, you can have 255 partitions with each partition having 255 host addresses available. You can then manage which partitions are assigned to remote networks. Each remote network partition is mapped to the subnet (see Figure 77).

AS/400 hides the fact that the remote networks are physically located behind AS/400. It will then act as a proxy and forward the datagram packets to the remote networks. When transparent subnetting is put into effect, the remote host appears to be part of the physical corporate network.

Transparent subnetworking is useful in environments in which it is either impractical or impossible to run routing protocols. Such a situation might occur in a bridged network or in networks that are running unsupported proprietary dynamic routing protocols. This is best suited for leased line connections when remote sites need to communicate with one another. If remote sites do not need to communicate with each other, however switched line connections are viable.

Figure 77 shows an example of how a network configuration could use transparent subnetting for remote LAN access.

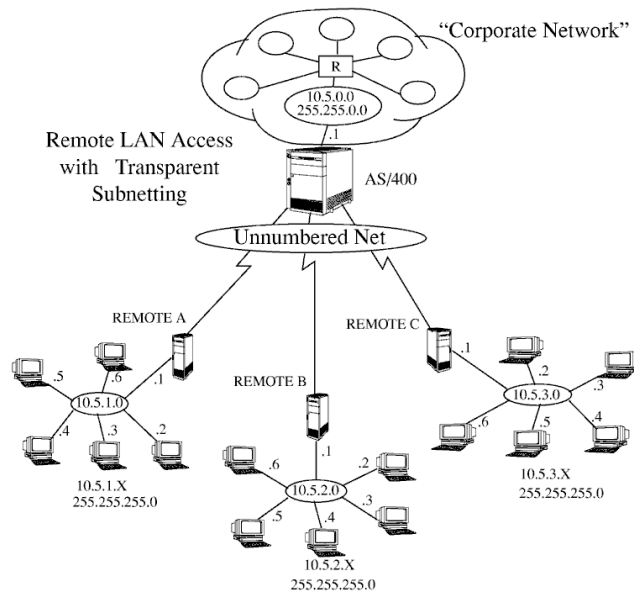


Figure 77. Example Configuration for Remote LAN Access with Transparent Subnetting

Note: This diagram shows PPP links carrying out unnumbered networking. The IP address that is used to connect the home network (10.5.0.0) to AS/400 (10.5.0.1) with a subnet mask of 255.255.0.0 is the same local IP address that is used for all remote dial-up connections. No IP addresses are required to be assigned to the PPP links.

The following steps are an example of how to configure AS/400 to connect to multiple LAN subnets. The example uses a single PPP answer profile to put transparent subnetting into effect, based on Figure 77.

Creating a Point-to-Point Profile for LAN Transparent Subnetting

Access Point-to-Point functions through Operations Navigator.

From the **General** page, perform the following tasks:

- Specify a profile name (REMOTE_ABC for this example).
- Specify a description for the connection profile (Answer profile For Remote Box A, B, or C in this example).
- Select PPP as the "Type."
- Select Switched line-answer as the "Mode."

From the **TCP/IP Settings** page, perform the following tasks:

- Specify your "Local IP address." Since this example is *Unnumbered Net*, an existing token-ring interface, 10.5.0.1, is selected from the "Local IP address" pull-down box.
- Specify your "Remote IP address." In this example, Route specified is selected. Remote IP addresses will be defined from the entries that are defined in the **Routing** dialog.

Note: Route specified is valid only for switched line-answer profiles.

- Check "Allow IP forwarding." IP packets that originate from the remote LAN are allowed to flow through AS/400 to the "Corporate Network" or other remote LANs.

Click **Routing** to add entries to the Route specified IP addresses. From the **Routing** page, perform the following tasks:

- For "Dynamic routing," select None. Routing will be carried out through static routing.
- For "Static routing," select Use static routes:
 - Caller REMOTE_A receives IP address 10.5.1.1. A subnet mask of 255.255.255.0 allows AS/400 to add a direct route to the 10.5.1.0 network.
 - Caller REMOTE_B receives IP address 10.5.2.1. A subnet mask of 255.255.255.0 allows AS/400 to add a direct route to the 10.5.2.0 network.
 - Caller REMOTE_C receives IP address 10.5.3.1. A subnet mask of 255.255.255.0 allows AS/400 to add a direct route to the 10.5.3.0 network.

Note: You can add more routes if additional LANs are attached to AS/400 through subnetting.

Figure 78 on page 107 shows the **Routing** dialog.

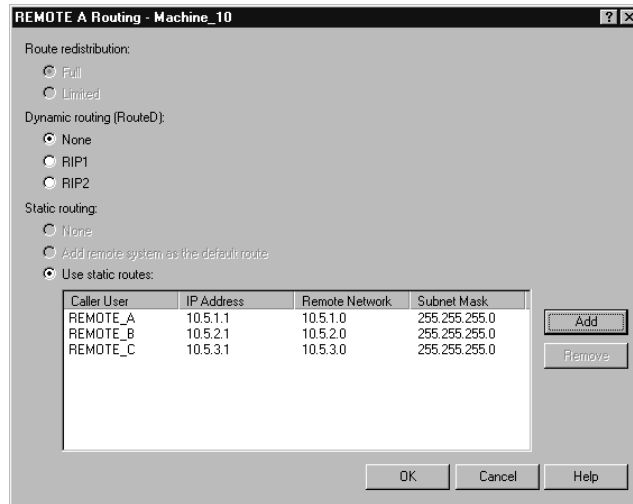


Figure 78. Example Routing for Transparent Subnetting

Validation List: Click the **Authorization** tab; then click **New** or **Open** to enter authorization information.

1. Add a validation list for remote LANs A, B, and C by using PAP or CHAP authentication.
2. Click **OK** when the validation list is completed.
3. Click **OK** to close the connection profile properties page.

Example: Remote LAN Access with Dynamic Routing (RIP)

Remote LAN access with dynamic routing allows remote clients on separate LANs to communicate with one another as if they were on the same routing domain. It enables any host in the corporate network to communicate with any remote LAN host. Dynamic routing from the home network places no address restrictions on the remote network. Further, there is no need for a hierarchical address scheme or relationship from the host to the home address space.

The home network is able to learn about the remote networks by using a dynamic routing algorithm within the home network using RIP 1 or 2. This is accomplished by the home network updating its routing tables as information is extracted from the remote networks static routing table.

The number of remote networks directly affects the size of the dynamic routing tables. No dynamic routing protocols run on the remote PPP links. Instead, they are run by AS/400 and the corporate network. The routing protocol automatically redistributes remote routes throughout the corporate network.

A solution where the corporate network is running RIP 1 or 2 as its routing protocol *and* the remote links are connecting to a small amount of networks would be to implement remote LAN access with dynamic routing. If RIP 1 or 2 is being run in AS/400, the static routes will automatically be redistributed. This eliminates the need to run RIP over the remote links.

Figure 79 on page 108 shows an example of how a network configuration could use RIP for remote LAN access.

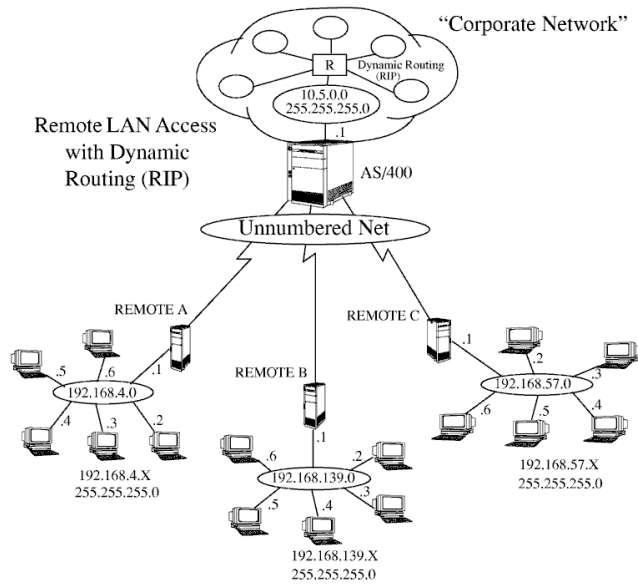


Figure 79. Example Configuration for Remote LAN Access with Dynamic Routing (RIP)

Creating a Point-to-Point Profile for LAN Access with Dynamic Routing (RIP)

Access Point-to-Point functions through Operations Navigator.

From the **General** page, perform the following tasks:

- Specify a profile name (REMOTE_ABC for this example).
- Select PPP as the "Type."
- Select Switched line-dial as the "Mode."

From the **TCP/IP Settings** page, perform the following tasks:

- Specify your "Local IP address." Since this example is *Unnumbered Net*, an existing token-ring interface, 10.5.0.1, is selected from the "Local IP address" pull-down box.
- Specify your "Remote IP address." In this example, Route specified is selected. Remote IP addresses are defined from the entries that are defined in the **Routing** dialog that appears when you click Routing.

Note: Route specified is valid only for switched line-answer profiles.

- Check "Allow IP forwarding." IP packets that originate from the remote LAN are allowed to flow through AS/400 to the "Corporate Network" or other remote LANs.

Click **Routing** to add entries to the Route specified IP addresses. From the **Routing** page, perform the following tasks:

- For "Dynamic routing," select None. Routing will be accomplished through static routing. RIP is used within the "Corporate Network" but **not** on remote links.
- For "Static routing," select Use static routes:
 - Caller REMOTE_A receives IP address 192.168.4.1. A subnet mask of 255.255.255.0 allows AS/400 to add a direct route to the 192.168.4.0 network.

- Caller REMOTE_B receives IP address 192.168.139.1 A subnet mask of 255.255.255.0 allows AS/400 to add a direct route to the 192.168.139.0 network.
- Caller REMOTE_C receives IP address 192.168.57.1. A subnet mask of 255.255.255.0 allows AS/400 to add a direct route to the 192.168.57.0 network.

Note: You can add more routes if additional LANs are attached to AS/400 through subnetting.

Figure 80 shows the Routing dialog.

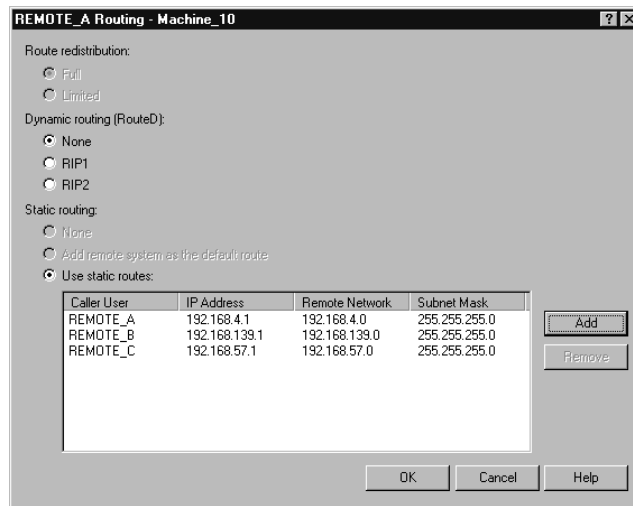


Figure 80. Example Routing for Dynamic Routing

Monitoring Activity

You use Operations Navigator to create, change, view, start or stop a PPP connection profile using the new PPP line type. After you configure a PPP connection profile, you can start, stop, or delete it. To do this, right click the connection profile you want in Operations Navigator’s tree view. Then, select Start, Stop, or Delete from the context menu.

The Connection Profile list also allows you to monitor several things about a point-to-point job by scrolling left or right on the window. The **Status** indicator tells whether the connection is starting, in use, or ending. A value of Inactive specifies that no jobs are currently running for that profile. A value of Ended - information available, on the other hand, indicates that the connection has ended and that information is available for that job. Examples of other status values that occur while the job is running include Session job starting, Waiting for incoming call, and Active.

Scroll to the right in the Connection Profile list to display more information. From here, information can be gathered about a particular job. By using a combination of the values for Job number, Job user, and Job, you can look at additional information such as Point-to-Point **Messages**, **Jobs**, and **Printer Output** information. For additional information, see “Point-to-Point Jobs” on page 110.

Point-to-Point Jobs

You can look at messages and jobs in Operations Navigator's tree view.

Messages

To display a list of messages as a result of a job, click **Messages** in Operations Navigator tree view.

The first time you display this dialog, the messages will be for the current user. However, these jobs run in the QSYSWRK subsystem under user QTCP. Therefore, it is necessary to display the messages for QTCP to see the messages for the job.

To display messages for QTCP, select **Include** from the **Options** menu.

Select "User name" and specify QTCP in the space. Click **OK**. You will be returned to the **Include** dialog. Click **OK**. "Messages for QTCP" will be displayed in the message line. From here, you will be able to see messages pertaining to the profile that you have started or stopped. Find the message that you are interested in and double-click it in the **From user** column. This will bring up the detailed message information dialog with a thorough explanation of the message.

Jobs

To look at information for an active job or a job that has completed, click **Jobs** in Operations Navigator tree view. You will now have to perform a similar search as you did for messages.

Select **Include** from the **Options** menu. For an active job, fill in "User" with QTCP, "Job name" with A11, and click **OK**. In addition for a completed job, select Yes for "Show completed jobs with printer output."

A list of all jobs for QTCP will be displayed on the right side of the window. From here, you can look at the information that is available for the job.

Point-to-point job names start with the string QTPP, where dial-out jobs are QTPPDIALnn and dial-in jobs are QTPPANSnn. The number string nn is sequential from 1 to 99. Find your job name in the list and right-click the desired job name.

From the context menu, select **Job Log** for an active job.

Double-clicking a **Message ID** value displays detailed message information.

Double-clicking the **Output name** value displays a viewer dialog with information on the completed job.

Printer Output

To look at printer output for a job that has completed, click **Printer Output** in Operations Navigator tree view. Next, select **Include** from the **Options** menu. Fill in the user with QTCP as explained in "Messages" and click **OK**. From here, you can look at the **User specified** data column and find your job name. Double-clicking the **Output name** value displays information about the job.

If you wish to narrow your search and do not wish to see all QTCP jobs, you can specify AS/400 job name on the Include dialog. The job name consists of three parts: Job number, Job user, and Job name.

Optional Monitoring of Point-to-Point Activity

An optional method to monitor activity for Point-to-Point connection profiles is to invoke option 14 from the Work with Point-to-Point TCP/IP (WRKTCPTP) command on a 5250 AS/400 console. For more information about this option, see "Monitoring Point-to-Point Activity" on page 134.

Connection Alternatives

This section provides you an overview of the link layer connection alternatives available to you.

Point-to-point links establish a physical connection between a local and remote host. When connected, these links provide dedicated bandwidth and come in a variety of data rates and protocols. With point-to-point links, you can choose from dial-up analog modem connections, leased analog lines, and switched and leased digital services of various kinds.

PPP is a method of transmitting datagrams over serial point-to-point links. PPP enables interconnection of multiple vendor equipment and multiple protocols by standardizing point-to-point communications. The PPP data link layer uses HDLC-like framing for encapsulating datagrams over both asynchronous and synchronous point-to-point telecommunication links.

While PPP supports a wide range of link types, SLIP is limited to asynchronous link types. SLIP is generally employed only for analog links.

Local telephone companies offer traditional telecommunications services in an ascending scale of capabilities and cost. These services use existing telephone company voice network facilities between customer and central office. A comparison of communication services and their relative costs is shown in Table 8. **The costs shown are not intended to reflect current pricing.** Instead, they are intended to show relative differences between services. Cost rates on leased lines are usually distance-based, while the switched lines may also be time-based.

Note: The following table is for illustration purposes only. It does not include all supported AS/400 configurations. The costs shown are not intended to reflect current pricing.

Table 8. Comparison of communication services and relative costs

Service	Line Speed	Required Equipment	DTE/DCE Interface	Relative Cost (per month)
Analog (leased and switched)	33.4Kbps or less	Modem	RS232 asynchronous	\$20-\$150
Digital Data Service (DDS)	56.0Kbps or less	CSU/DSU	X.21/V.35/RS-449 synchronous	\$50-\$500
Switched-56	56.0Kbps	CSU/DSU with V.25bis dial	V.35/RS-449 synchronous	\$50-\$250

Table 8. Comparison of communication services and relative costs (continued)

Service	Line Speed	Required Equipment	DTE/DCE Interface	Relative Cost (per month)
ISDN switched	56, 64, 112, or 128Kbps	ISDN terminal adapter	RS232 asynchronous	\$50-\$250
Fractional T1	56Kbps to 1.544Mbps	CSU/DSU or T1 mux	X.21/V.35/RS449 synchronous	\$100-\$2,000
T1/E1	56Kbps to 1.544/2.048 Mbps	CSU/DSU or T1 mux	X.21/V.35/RS449 synchronous	\$350-\$2,000

Analog Phone Lines

The analog connection, which uses modems to carry data over leased or switched lines, sits at the bottom of the point-to-point scale. Leased lines are full-time connections between two specified locations, while switched lines are regular voice-phone lines.

The fastest modems today operate at an uncompressed rate of 33.6Kbps. Given the signal-to-noise ratio on unconditioned voice-grade telephone circuits, though, this rate is often unattainable. Modem manufacture claims of higher bit-per-second (bps) rates are usually based on a data compression (CCITT V.42bis) algorithm that is utilized by their modems.

Although V.42bis has the potential to achieve as much as four-fold reduction in data volume, compression depends on the data and rarely reaches even 50%. Data already compressed or encrypted may even increase with V.42bis applied.

Emerging technology referred to as X2 or 56Flex extends the bps rate to 56k for analog telephone lines. This is a hybrid technology that requires one end of the PPP link to be digital while the opposite end is analog. Additionally, the 56Kbps applies only when you are moving data from the digital toward the analog end of the link. This technology is well suited for connections to ISPs with the digital end of the link and hardware at their location.

Typically, you can connect to a V.24 analog modem over an RS232 serial interface with an asynchronous protocol at rates up to 115.2Kbps.

Digital Data Service

With digital, data travels all the way from the sender's computer to the central office of the telephone company, to the long distance provider, to the central office, and then to the computer of the receiver in digital form. Digital signaling offers much more bandwidth and higher reliability than analog signaling. A digital signaling system eliminates many of the problems that analog modems must deal with, such as noise, variable line quality, and signal attenuation.

DDS

The most basic of digital services is called Digital Data Services (DDS). DDS links are leased, permanent connections, running at fixed rates of up to 56Kbps. This service is also commonly designated as DS0.

You can connect to DDS using a special box called Channel Service Unit/Data Service Unit (CSU/DSU), which replaces the modem in the analog scenario. DDS has physical limitations that are primarily related to the distance between the CSU/DSU and the Telephone Company Central Office. DDS works best when distance is less than 30,000 feet. Telephone companies can accommodate longer distances with signal extenders, but this service comes at higher cost. DDS is best suited for connecting two sites that are served by the same Central Office. For long distance connections that span different Central Offices, mileage charges can quickly add up to make DDS impractical. In such cases, Switched-56 may be better solution.

Typically, you can connect to a DDS CSU/DSU over V.35, RS449, or X.21 serial interface with synchronous protocol at rates up to 56Kbps.

Switched-56

When you do not need a full-time connection, you can save money by using switched digital service, which is generally called Switch-56 (SW56). An SW56 link is similar to DDS setup in that the DTE connects to the digital service by way of CSU/DSU. An SW56 CSU/DSU, however, includes a dialing pad from which you enter the phone number of the remote host.

SW56 lets you make dial-up digital connections to any other SW56 subscriber anywhere in the country or across international borders. An SW56 call is carried over the long distance digital network just like a digitized voice call. SW56 uses the same phone numbers as the local telephone system, and usage charges are the same as those for business voice calls.

SW56 is available only in North American networks, and it is limited to single channels that can only carry data. SW56 is an alternative for locations where ISDN is unavailable.

Typically, you can connect to a SW56 CSU/DSU over V.35 or RS 449 serial interface with synchronous protocol at rates up to 56Kbps. With a V.25bis call/answer unit, data and call control flow over a single serial interface.

ISDN

Like Switched-56, ISDN also provides switched end-to-end digital connectivity. Unlike other services, however, ISDN can carry both voice and data over the same connection.

There are different types of ISDN services, with Basic Rate Interface (BRI) being the most common. BRI consists of two 64Kbps B channels to carry customer data and a D channel to carry signaling data. The two B channels can be linked together to give a combined rate of 128Kbps. In some areas, the phone company may limit each B channel to either 56Kbps or 112Kbps combined.

There is also a physical constraint in that the customer location must be within 18,000 feet of the central office switch. This distance can be extended with repeaters.

You can connect to ISDN with a device called a terminal adapter. Most terminal adapters have an integrated network termination unit (NT1) that allows direct connection into a telephone jack. Typically, terminal adapters connect to your

computer over an asynchronous RS232 link and use the AT command set for setup and control, much like conventional analog modems. Each brand has its own AT command extension for setting up parameters that are unique to ISDN. In the past, there were many interoperability problems between different brands of ISDN terminal adapters. These problems were due mostly to the variety of rate adaptation protocols that were available in V.110 and V.120 as well as bonding schemes for the two B channels. The industry has now converged to synchronous PPP protocol with PPP multilink for linking two B channels.

Some terminal adapter manufactures integrate V.34 (analog modem) capability into their terminal adapters. This enables customers with a single ISDN line to handle either ISDN or conventional analog calls by taking advantage of the simultaneous voice/data capabilities of ISDN services. New technology also enables a terminal adapter to operate as the digital server side for 56K(X2/56Flex) clients.

Typically, you would like to connect to an ISDN terminal adapter over an RS232 serial interface using asynchronous protocol at rates up to 230.4Kbps. However, the maximum AS/400 baud rate for asynchronous over RS232 is 115.2Kbps. Unfortunately, this restricts the maximum byte transfer rate to 11.5k bytes/sec, while the terminal adapter with multi-linking is capable of 14/16k bytes uncompressed. Some terminal adapters support synchronous over RS232 at 128Kbps, but AS/400 maximum baud rate for synchronous over RS232 is 64Kbps. The AS/400 is capable of running asynchronous over V.35 at rates up to 230.4Kbps, but terminal adapter manufacturers generally do not offer such a configuration. Interface converters that convert RS232 to V.35 interface could be a reasonable solution for the problem, but this approach has not evaluated for AS/400. Another possibility is to use terminal adapters with V.35 interface synchronous protocol at rate of 128Kbps. Although this class of terminal adapters exists, it does not appear that many offer synchronous Multilink PPP.

T1/E1

A T1 connection bundles together twenty-four 64Kbps (DS0) time division multiplexed (TDM) channels over 4-wire copper circuit. This creates a total bandwidth of 1.544Mbps. An E1 circuit in Europe and other parts of the world bundles together thirty-two 64Kbps channels for a total of 2.048Mbps.

TDM allows multiple users to share a digital transmission medium by using pre-allocated time slots. Many digital PBXs take advantage of T1 service to import multiple call circuits over one T1 line instead of having 24 wire pairs routed between the PBX and telephone company.

It is important to note that T1 can be shared between voice and data. A company's telephone service may come over a subset of a T1 link's 24 channels, for instance, leaving remaining channels available for internet connectivity.

A T1 multiplexer device is needed to manage the 24 DS0 channels when a T1 trunk is shared between multiple services. For a single data-only connection, the circuit can be run unchannelized (no TDM is performed on the signal). Consequently, a simpler CSU/DSU device can be used.

Typically, you can connect to a T1/E1 CSU/DSU or multiplexer over V.35 or RS 449 serial interface with synchronous protocol at rates at a multiple of 64Kbps to 1.544Mbps or 2.048Mbps. The CSU/DSU or multiplexer provides the clocking in the network.

Fractional T1

With Fractional T1 (FT1), a customer can lease any 64Kbps sub-multiple of a T1 line. FT1 is useful whenever the cost of dedicated T1 would be prohibitive for the actual bandwidth customer uses.

With FT1 you pay only for what you need. Additionally, FT1 has the following feature that is unavailable with a full T1 circuit: Multiplexing DS0 channels at the telephone company's central office. The remote end of an FT1 circuit is at a Digital Access Cross-Connect Switch that is maintained by the telephone company. Systems that share the same digital switch can switch among each other's DS0 channels. This scheme is popular with ISPs that use a single T1 trunk from their location to the telephone company's digital switch. In these cases, multiple clients can be served with FT1 service.

Typically, you can connect to a T1/E1 CSU/DSU or multiplexer over V.35 or RS 449 serial interface with synchronous protocol at some multiple of 64Kbps. With FT1, you are pre-allocated a subset of the 24 channels. The T1 multiplexer must be configured to fill only the time slots that are assigned for your service.

Using an Asynchronous Modem or ISDN Terminal Adapter

This material is covered in the **AS/400e Information Center** under the **TCP/IP** topic. For more information see "TCP/IP Topics in the Information Center" on page xv.

PPP ISDN Support

This material is covered in the **AS/400e Information Center** under the **TCP/IP** topic. For more information see "TCP/IP Topics in the Information Center" on page xv.

Configuring SLIP Connection Profiles

This section covers sample property sheets for configuring a Serial Line Interface Protocol (SLIP) Point-to-Point network connection profile using Operations Navigator. There is very little difference in how a SLIP connection profile is created and used from how a PPP connection profile is created and used. This section will mainly point out any features that are unique to SLIP. Refer to previous PPP sections for more details on the property pages. You can also refer to the PPP scenarios in the previous section. These scenarios will also work for SLIP.

Note: This section does not cover how to create SLIP point-to-point connection profiles (which use *ASYNC lines). For more information on creating SLIP point-to-point connection profiles, see "Using SLIP with an Asynchronous Line Description" on page 126.

The **General** page allows you to define the following general attributes of a connection profile:

- Name of the profile
- Optional description for the profile
- Which protocol to use (SLIP in this example)

- Operating mode which helps to define the physical connection (switched line-answer in this example)

The **Connection** page allows you to define the line description that will be used for the connection. For switched dial profiles, this page also allows you to specify the remote phone number or numbers required to connect to the remote system. For SLIP, the "Maximum transmission units" is configurable. This determines the size in bytes of each Packet that is sent to the remote system.

The **TCP/IP Settings** page allows you to define the local and remote IP addresses that will be used for this connection. It also allows you to define additional TCP/IP attributes.

For the remote IP address, "Route specified" is enabled only for switched line-answer profiles. This gives the administrator greater flexibility in selection of remote IP addresses. If the IP forwarding option is selected, then this allows the remote user to send IP datagrams to the network that is connected to this AS/400. If this is not desirable, then disable this feature. VJ header compression is not configurable on the answer side since the client determines if VJ header compression will be used.

Dynamically assigned IP addresses require the use of a connection script. For more information, see "Writing Connection Dialog Scripts" on page 118.

Use the **Script** page to specify whether you want to use a connection script for the connection profile.

A Connection script allows AS/400 and remote systems to exchange information. This is the only method of passing user ID, password, and IP address information for SLIP. For more information, see "Writing Connection Dialog Scripts" on page 118.

Use the **Domain Name Server** page to define the domain name server to use for a connection profile. The page is enabled for switched line-dial and leased line-initiator connection profiles only.

You may want to configure the Domain Name Server (DNS) under the following conditions:

- You want to have automatic host-name-to-IP-address resolution
- The remote system has a DNS available

If authentication is required, use the **Authentication** page to define which users may connect to this system.

Figure 81 on page 117 is an example of a switched answer or leased line terminator authentication page for a SLIP connection profile. If authentication is enabled, then a connection script *must* be used to prompt for and accept the user name and password information from the remote system. For more information on connection scripts, see "Writing Connection Dialog Scripts" on page 118.



Figure 81. SLIP Connection Profile Properties - Authentication (Answer)

The validation list name is the name of the validation list object that contains a list of the user names that may connect to this system as well as their passwords. To open a validation list name, click **Open** on the **Authentication** page.

Figure 82 is an example of a switched dial or leased line initiator authentication page for a SLIP connection profile. This page is used to define the user name and password that will be used to identify this system when connecting to a remote system.

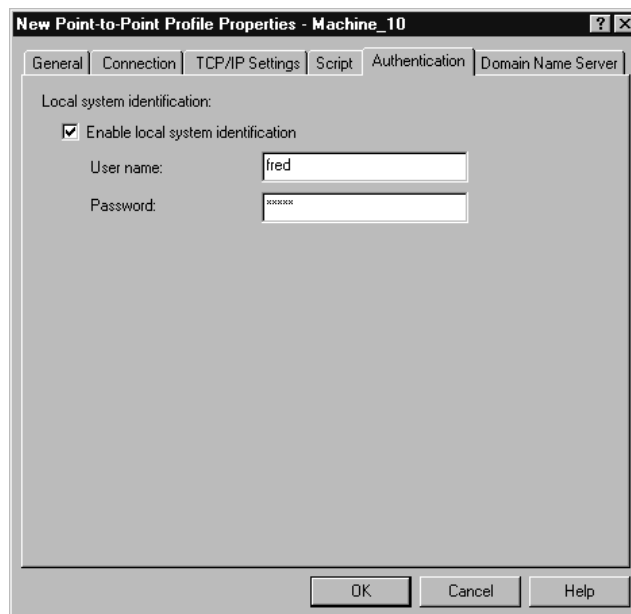


Figure 82. SLIP Connection Profile Properties - Authentication (Dial)

If authentication is enabled, then a connection script *must* be used to pass the user name and password information to the remote system. For more information on connection scripts, see “Writing Connection Dialog Scripts”.

Writing Connection Dialog Scripts

A connection dialog script is information in an AS/400 source file that AS/400 uses to exchange data prior to establishing a SLIP (or in some cases a PPP) connection.

Note: Your AS/400 does *not* support the capability to select SLIP or PPP service from a connection dialog script when a remote system dials AS/400.

The configuration profile determines which connection script is used. The dialog text must exactly match the connection sequence that is exchanged between AS/400 and a remote system.

Each connection script contains the dialog text required for the exchange of authorization and connection parameters with a remote host. The configuration profile determines which connection script is to be used. The dialog text must be an exact match for the connection to be established between the client and server or client and ISP.

Note: AS/400 expects each input line of dialog text from the remote system to end with an end-of-line character. See “Rules for Creating and Changing Connection Scripts” on page 119 for further details.

Connection Script Considerations for SLIP

For SLIP on an AS/400, connection dialog scripts commonly perform the following functions:

- Exchange sign-on and password information to authorize an AS/400 to connect to a remote host or ISP
- Exchange sign-on and password information to authorize a remote host to connect to an AS/400
- Select the type of service requested (for example SLIP or PPP) from an ISP
- Allow the dynamic assignment of an IP address by an ISP to an AS/400
- Allow AS/400 to assign an IP address dynamically to a remote host

If you do not require these functions for SLIP connections to or from an AS/400, you do not need to use a connection script. If you do not use connection dialog scripts, though, AS/400 will establish SLIP connections without user ID or password authentication.

Creating and Changing Connection Scripts

You cannot change the default connection script file QUSRSYS/QATOCPPSCR. You must first create your own connection script file. Do this by copying the default file as follows:

```
CPYF FROMFILE(QUSRSYS/QATOCPPSCR)
      TOFILE(lib/file) FROMMBR(*ALL)
      TOMBR(*FROMMBR) MBROPT(*ADD)
      CRTFILE(*YES)
```

where lib/file represents your own new file, such as QUSRSYS/SLIPSCRIPT.

This creates a new file that uses the existing default connection script file as the base and that contains all of the same data. You can now change the data for your use.

You must refer to the new connection script file name in the TCP/IP point-to-point configuration profile that will use the new connection script.

If you need to use characters in your connection script file that are not supported by EBCDIC CCSID 500, which is the CCSID of the default connection script, then go to "NLS Considerations" on page 125.

Rules for Creating and Changing Connection Scripts

There are many special rules for creating and changing connection scripts. The rules are as follows:

- Column 1 of each line in the connection script may contain a control character. The control character specifies the action to be taken by AS/400 system. Column 2 may be left blank.
 - The AS/400 system ignores comment lines.
Comment lines begin with either an asterisk (*), or are completely blank.
 - Output lines for the remote system begin with an ampersand (&)
 - To send an asterisk (*) to the remote host, put an ampersand (&) in column 1 followed by an asterisk in column 2.
To send an ampersand (&) to the remote host, put an ampersand in column 1 followed by an ampersand in column 2.
 - A blank line with an ampersand in column 1 will not be sent to the remote host. Refer to the (PROMPT) keyword for this operation.
- The AS/400 system interprets any non-blank line that does not contain one of the previously described control characters as input from the remote host.
- You can mix upper-case and lower-case in a line that contains expected input from the remote host.
AS/400 translates all input to upper-case before comparing the dialog match text.
- AS/400 expects each line of dialog text from the remote system to end with an end-of-line character, such as a carriage return, New Line, or Line Feed. If this is not possible, dialog match text may be changed to achieve a match. For example, if the remote server host prompts for a password with "Password:", then the match text in the connection script may be shortened to "Pass" or "Password" (the colon character has been omitted).
- Fields enclosed in parentheses () contain the keywords that describe an input or an output operation.
You cannot define the keywords. However, you can control the operations performed during the connection dialog. To control the operations and the order in which they run, create a unique connection script using appropriately placed keywords.
- You cannot change the default connection scripts.
If you need a connection script that is different from all the default ones, you must copy the default connection scripts and change the copy.

The valid keywords are:

(USERID)

When remote systems dial in, AS/400 uses this keyword to validate the user ID that the remote system is using to connect to AS/400. The keyword is only valid if an access authorization list is configured.

When an AS/400 user dials out to a remote system, AS/400 uses this keyword to pass the user ID to the remote server for connection validation.

If the remote system requires both an account name and a user ID, configure both as described in "Remote System Access Information" on page 152.

(PASSWORD)

For dial-in sessions, AS/400 uses this keyword to validate the password for the user ID that was received on the (USERID) keyword. AS/400 uses this keyword only when an access authorization list is configured.

For dial-out sessions, AS/400 uses this keyword to send the password to the remote server for connection validation.

(IPADDR)

The IP address to be exchanged and activated to set up a session.

(IPGATE)

The gateway IP address to be exchanged and activated to set up a session.

(PROMPT)

Specifies that AS/400 is to either:

- Pause for input on an input line, or
- Send a blank followed by a carriage return and a line feed for an output line.

This keyword is useful for synchronizing a connection script with a remote system. Placing this keyword on the input line of a connection script causes the host to pause until input is received before continuing the dialog. Placing this keyword on the output line of a connection script causes the host to signify readiness to continue the dialog. Typically this keyword is placed on the first line of the connection script to synchronize the host to host connection before beginning the connection dialog.

Some ISPs may look for specific responses to initiate a SLIP connection. For example, specific characters like @ , &, or a carriage return without any text may be required.

To change what AS/400 sends, use the following command:

```
& (PROMPT) 'xxyy'
```

where 'xxyy' is the two-digit hexadecimal representation for any ASCII characters that you wish to send.

For example, & (PROMPT) '4053' would cause AS/400 to send the characters '@S', and & (PROMPT) '0D' would cause AS/400 to send only a carriage return. The hexadecimal representation of the characters to be sent must be two digits each and represent valid ASCII characters.

(WAIT)

Causes AS/400 to wait 1 second before resuming.

You can cause a time delay from 1 to 99 seconds. To change the time delay, use the following command:

(WAIT) 'xx'

where xx is the delay value in seconds. This keyword works the same on both input and output lines, and is useful for controlling the rate and timing of the dialog.

You do not need to include all of the connection dialog in the connection script. For input lines containing a keyword, it is good practice to include some text preceding the keyword to help AS/400 system locate the expected keyword input. Also, as previously described, it may be necessary to shorten input match text in order to obtain a match when connecting to systems that do not send end-of-line characters at the end of a line.

Location of Default Connection Scripts: You can find the default connection scripts contained in Table 9. The members are in file QATOCPPSCR.

SLIP Connection Scripts-Examples

The connection scripts in Table 9 are examples for the following:

- Dialing out from AS/400 to a remote host.
- Dialing in to AS/400 from a remote host.

Each member has a different purpose, stated in the comment. Each script is placed in its own member in file QATOCPPSCR in some connection script library.

Table 9. SLIP Client Connection Scripts-Examples

Client (Dial-Out) Examples	Server (Dial-In) Examples
Member DIAL400 in File QATOCPPSCR <pre>***** * CLIENT CONNECTION SCRIPT EXAMPLE * FOR AS/400 WITH LOGIN AND PASSWORD PROMPT ***** & (PROMPT) login: & (USERID) password: & (PASSWORD) * END OF CLIENT CONNECTION SCRIPT EXAMPLE</pre>	Member ANS400 in File QATOCPPSCR <pre>***** * SERVER CONNECTION SCRIPT EXAMPLE * FOR AS/400 WITH LOGIN AND PASSWORD PROMPT ***** (PROMPT) & login: (USERID) & password: (PASSWORD) * END OF SERVER CONNECTION SCRIPT EXAMPLE</pre>
Member DIAL400I in File QATOCPPSCR <pre>***** * CLIENT CONNECTION SCRIPT EXAMPLE * FOR AS/400 WITH DYNAMIC IP ADDRESS ***** & (PROMPT) login: & (USERID) password: & (PASSWORD) YOUR IP ADDRESS IS (IPADDR) MY IP ADDRESS IS (IPGATE) * END OF CLIENT CONNECTION SCRIPT EXAMPLE</pre>	Member ANS400I in File QATOCPPSCR <pre>***** * SERVER CONNECTION SCRIPT EXAMPLE * FOR AS/400 WITH DYNAMIC IP ADDRESS ***** (PROMPT) & login: (USERID) & password: (PASSWORD) & YOUR IP ADDRESS IS (IPADDR) MY IP ADDRESS IS (IPGATE) * END OF SERVER CONNECTION SCRIPT EXAMPLE</pre>

Table 9. SLIP Client Connection Scripts-Examples (continued)

Client (Dial-Out) Examples

Member DIALAIX in File QATOCPPSCR

```
*****
* CLIENT CONNECTION SCRIPT EXAMPLE
* FOR AIX 4.1 SLIPLOGIN SCRIPT
*****
&(PROMPT)
login:
&(USERID)
password:
&(PASSWORD)
Called system's address is (IPGATE)
Calling system's address is (IPADDR)
discipline
* END OF CLIENT CONNECTION SCRIPT EXAMPLE
```

Member DIALIGN in File QATOCPPSCR

```
*****
* CLIENT CONNECTION SCRIPT EXAMPLE FOR
* IBM GLOBAL NETWORK (IGN) INTERNET SERVICE
*****
& &
Enter dial script version ==>
& 1.1
Enter service ==>
& INTERNET
Enter account userID password
& (USERID) (PASSWORD)
(IPADDR) is your IP address.
& (PROMPT)
(IPGATE) is the gateway IP address.
Begin TCP/IP communication now.
* END OF CLIENT CONNECTION SCRIPT EXAMPLE
```

Server (Dial-In) Examples

Member ANSAIX in File QATOCPPSCR

```
*****
* SERVER CONNECTION SCRIPT EXAMPLE
* FOR AIX 4.1 SLIPCALL SCRIPT
*****
(PROMPT)
& login:
(USERID)
& password:
(PASSWORD)
& Called system's address is (IPGATE)
& Calling system's address is (IPADDR)
& The netmask is 255.255.255.0
& Activating slip line discipline
* END OF SERVER CONNECTION SCRIPT EXAMPLE
```

Member ANSWIN95 in File QATOCPPSCR

```
*****
* SERVER CONNECTION SCRIPT EXAMPLE
* Windows 95 system with Microsoft Plus for Windows 95
* installed dialing into AS/400 *ANS session.
*****
(PROMPT)
& Userid:
(USERID)
& Password?
(PASSWORD)
& InternetLR/E>
(PROMPT)
& (IPGATE) IS AS/400 IP ADDRESS.
& (IPADDR) IS IP ADDRESS OF SYSTEM CALLING AS/400.
* END OF SERVER CONNECTION SCRIPT EXAMPLE
```

Creating SLIP Client (Dial-Out) Connection Scripts-Examples

This topic explains how to create connection scripts for dial-out connections.

The following is an example of a client connection script for a generic (non-IBM Global Network) ISP:

1. * Generic client connection script example
2. & (PROMPT)
3. Username:
4. & (USERID)
5. Password:
6. & (PASSWORD)
7. Protocol:
8. SLIP
9. The gateway address is (IPGATE)
10. Your IP address is (IPADDR)
11. * End of Generic client script example

After the modem dials and connects, AS/400 reads the connection script. Each line in the connection script causes AS/400 to send or receive data as follows:

1. Comment line.
2. AS/400 sends a blank followed by a carriage return to the remote system that AS/400 called.

3. AS/400 waits for the Username: prompt from the called system.
4. AS/400 sends the value specified in the configuration profile's Remote service access name field to the called system.
5. AS/400 waits for the Password: prompt from the called system.
6. AS/400 sends the value specified in the configuration profile's Remote service access password field to the called system.
7. AS/400 waits for the Protocol: prompt from the called system.
8. AS/400 sends the value slip to the called system to indicate that the SLIP protocol will be used.
9. AS/400 waits for the The gateway address is prompt from the called system. When this prompt is received, the next value received matches the (IPGATE) keyword in the script. AS/400 uses this value as the IP address of the called system for this point-to-point connection.
10. AS/400 waits for the Your IP address is prompt from the called system. When this prompt is received, the next value received matches the (IPADDR) keyword in the script. AS/400 uses this value as the local IP address of AS/400 for this point-to-point connection.
11. Comment line.

Creating SLIP Client (Dial-Out) Connection Scripts — Example: The following example describes in detail how to create a client connection script to an Internet Service Provider. The IBM Global Network is used as the ISP in the example.

When you open a new account with the IBM Global Network or with any ISP, the ISP should provide you with either:

- A connection dialog example
- An actual connection dialog script

The connection dialog for the IBM Global Network looks like the following example:

```
&
*****
Welcome to the IBM Global Network
*****
Enter dial script version ==>
1.1
Gateway: IBMT2YA0 Port: 22
Select one of the following services:
INTERNET
Enter service ==>
INTERNET
Enter account userID password [\new_password] ==>
usinet barrier password
129.37.3.150 is your IP address.
129.37.1.10 is the Gateway IP address.
Begin TCP/IP communication now.
```

Use the keywords and rules for creating AS/400 connection scripts (shown in “Rules for Creating and Changing Connection Scripts” on page 119) to convert the connection dialog for IBM Global Network into a client connection script.

* IBM Global Network Client Script Example

```
& &
version ==>
& 1.1
service ==>
& INTERNET
password
```

```
& (USERID) (PASSWORD)
(IPADDR) is your IP address.
(IPGATE) is the Gateway IP address.
ication now.
```

* End of IBM Global Network Client Script Example

Notice that this example differs from the supplied DIALIGN connection script in Table 9 on page 121. This is because a connection script needs to contain only sufficient text to allow AS/400 to locate and interpret the keywords and keyword responses between the two hosts. You must carefully ensure that enough text remains to maintain send/receive synchronization. The single word password and the last partial line ication now. demonstrate this concept.

If you cannot obtain a connection dialog for the server any other way, do the following:

1. Establish an interactive connection using a PC and copy down the dialog.
2. Use the dialog to create your client connection script.

Creating SLIP Server (Dial-In) Connection Scripts-Example

The following example is a server connection script for AS/400 to AS/400 SLIP connections:

1. * Server connection script example
2. (PROMPT)
3. & ENTER ACCOUNT/PASSWORD ==>
4. (USERID)/(PASSWORD)
5. & YOUR IP ADDRESS IS (IPADDR)
6. & (WAIT)'2'
7. & YOUR GATEWAY IP ADDRESS IS (IPGATE)
8. & BEGIN TCP/IP COMMUNICATION NOW
9. * End of Server connection script example

After the modem answers and connects, AS/400 reads the connection script. Each line in the connection script causes AS/400 to send or receive data as follows:

1. Comment line.
2. Waits for any input from the client before beginning dialog.
3. Sends the prompt for account and password.
4. Waits for the user ID and password from the client.
5. Sends the IP address to be used by the client.
6. Waits 2 seconds before proceeding.
7. Sends the IP address of the server host to the client.
8. Sends confirmation of the client-to-server connection.
9. Comment line.

Connection Script Considerations for PPP

For PPP on an AS/400, you can use connection dialog scripts to perform the following functions:

- Exchange sign-on and password information to authorize an AS/400 to connect to a remote host or ISP

- Exchange sign-on and password information to authorize a remote host to connect to an AS/400
- Select the type of service requested (for example SLIP or PPP) from an ISP

Normally, all of these functions are performed when a PPP connection is negotiated by AS/400. However, some ISPs may require user authorization or a type of service selection prior to negotiating the PPP connection. Additionally, you will need to take into account the following considerations:

- If a connection to or from an AS/400 is authenticated using a connection dialog script, then the user ID and password that will be used is the one that was specified in the connection profile for PAP validation. If the remote host does not support validation when PPP is negotiated, then you *must* deactivate this selection after you specify the PAP user ID and password.
- Connection validation with CHAP for PPP may also be requested after the initial authentication. This is accomplished using the connection dialog script with a different and more secure user ID and password.
- Any dynamic IP addresses that are exchanged during the connection dialog will not be used by AS/400.

NLS Considerations

Once a SLIP connection is established, the application, such as TELNET or FTP, that is running over the connection, must manage the ASCII to EBCDIC and EBCDIC to ASCII translation.

If you are not using a connection script to establish a SLIP connection to or from a remote system, then no other NLS considerations are necessary. If you are using a connection script then it may be important to understand how AS/400 translates connection scripts from EBCDIC to ASCII to be transmitted back and forth over the connection.

Connection scripts on AS/400 are stored in EBCDIC format. However, most systems that will connect to AS/400 are probably ASCII systems that will send their connection script data in ASCII format. Therefore, AS/400 must translate the ASCII data that is coming in to AS/400 over the connection from ASCII to EBCDIC. AS/400 can then compare the connection script to the EBCDIC connection script that is stored on AS/400.

Before AS/400 can send data to a remote system over the SLIP connection, AS/400 must translate the connection script data from EBCDIC to ASCII. The default EBCDIC and ASCII CCSID values that AS/400 uses to translate connection scripts are described below. The default values cover most needs, but you can change the default values if you need support for other characters.

The default connection scripts that are shipped with AS/400 are located in members in file QUSRSYS/QATOCPPSCR. See “Connection Dialog Scripts” on page 156 for details on using connection scripts and for the member names.

This default connection script file, QATOCPPSCR, uses a CCSID of 500, which means character set 697 and code page 500. Therefore, all character data in each connection script member have a hex code point that this CCSID supports. AS/400 uses the EBCDIC CCSID value to translate connection script data when AS/400 sends the connection script data to or receives the connection script data from a remote system.

AS/400 obtains the ASCII CCSID from the TCP/IP point-to-point configuration profile that is being used for the SLIP session. You can configure this value in the Script source information section of the profile, which is shown in “Use Connection Dialog Script” on page 145. The default ASCII CCSID value is 00819 (ISO 8859, part1 Latin Alphabet No.1). This ASCII CCSID covers most 8-bit languages that use character set 697. All of the code points that are defined for ASCII CCSID 00819 support translations to the EBCDIC CCSID 500. This default covers most translations. However, if you must use characters that are not supported in EBCDIC CCSID 500 and ASCII CCSID 00819, then you must do the following:

1. Change the ASCII CCSID defined in the TCP/IP point-to-point profile that you plan to use to a value that supports all of the characters that you require.
2. Create a connection script file with a CCSID that supports all of the characters that you require and that is compatible with the ASCII CCSID that you have selected.

Create the new connection script file as follows:

- Create the file with the desired EBCDIC CCSID:

```
CRSRCPF FILE(lib/file) MBR(*NONE)
        RCDLEN(128) AUT(*USE)
        CCSID(your_ccsid)
```

- Copy the default connection script file information into the new file:

```
CPYF FROMFILE(QUSRSYS/QATOCPPSCR)
      TOFILE(lib/file) FROMMBR(*ALL)
      TOMBR(*FROMMBR) MBROPT(*ADD)
      FMTOPT(*MAP)
```

Notes:

- a. If you use FMTOPT(*MAP), AS/400 database attempts to automatically translate the CCSID 500 code points in the default connection script file to the equivalent code points of the new CCSID in the new connection script file.
- b. If you do not want this automatic translation of data from CCSID 500 to your new CCSID, then use FMTOPT(*NOCHK).
- c. If you do not want to copy or use the default connection script members, then omit the step for CPYF. The SCRSRCPF step will suffice.

The ASCII and EBCDIC CCSID values that you select must be compatible CCSIDs. If the CCSIDs are not compatible, AS/400 issues message TCP8373 'Unable to convert data from CCSID &1' to CCSID &2' and ends the SLIP session. See *National Language Support*, SC41-5101 for information that can help you determine which CCSID to use and which ASCII and EBCDIC CCSIDs are compatible.

Using SLIP with an Asynchronous Line Description

This section discusses how to create a SLIP connection profile that uses an asynchronous line description.

Beginning in V4R2 support was added for the PPP line type. It is possible to create a SLIP connection profile using either a PPP line description or an asynchronous line description. While connection profiles using the older asynchronous line description type can still be used, we strongly recommend that you use the new PPP line type instead.

You configure a SLIP connection profile using a PPP line description with Operations Navigator. You configure SLIP over an asynchronous line description using the CL command line interface.

See “Configuring SLIP Connection Profiles” on page 115 for working with a SLIP connection profile that uses a PPP line description.

SLIP over an asynchronous line description uses the following four CL commands:

- CFGTCPPTP - Configure Point-to-Point TCP/IP
- ENDTCPPTP - End Point-to-Point TCP/IP
- STRTCPPTP - Start Point-to-Point TCP/IP
- WRKTCPPTP - Work with Point-to-Point TCP/IP

With these commands, you can start and end SLIP activity on AS/400. But more importantly, you can create and use connection profiles and connection dialog scripts.

Connection Dialog Scripts

Connection dialog scripts (or, simply, connection scripts) allow AS/400 and remote systems to exchange sign-on and password information before a remote client may connect to AS/400 system. Connection dialog scripts also allow you to authorize users for connection, so that the system is protected from unwanted users.

Another common use of connection scripts is to dynamically assign an IP address to the remote SLIP client. If you do not require these functions, you can bypass script processing.

Other point-to-point protocols such as X.25 and frame relay, do not have connection script support.

Configuring AS/400 Point-to-Point for SLIP

Before You Configure AS/400 for SLIP - Checklist

To create a SLIP connection profile that uses the older asynchronous line description, you need the following:

An asynchronous line description on AS/400.

See “Step 1 - Configure an Asynchronous Line Description” on page 129 to check for or create an asynchronous line description.

Your modem make and model, if you are using a modem. Also, make sure that you have the owner’s manual for your modem.

See “Step 2 - Configure AS/400 For Your Modem” on page 130 to check for or create a modem entry.

Information to complete your SLIP configuration profile.

See “Step 3 - Determine Configuration Profile Type” on page 132 to check for or create various SLIP connection profiles.

There are other options you may choose to configure. See “Dial-In (*ANS) Point-to-Point Profile Parameters” on page 140 and “Dial-Out Point-to-Point Profile Parameters” on page 147 for details about the connection profiles.

Hardware Requirements for the Asynchronous Line Description

To use the SLIP protocol, you must establish a serial asynchronous connection with the remote system. To establish this connection, the appropriate adapters and I/O Processor (IOP) that support serial (RS232) ports must be installed.

The following two families of adapters that can be used:

- You can use the I/O adapters listed below only with a SLIP connection that uses an asynchronous line description. You *cannot* use these IOAs with the new PPP line description.
 - 2609 -- Two-line EIA 232/V.24 adapter
 - 2612 -- One-line EIA 232/V.24 adapter
- You can use the I/O adapters listed below with either the PPP line description or the asynchronous line description. These IOAs *can* be used with either PPP or SLIP connections.
 - 2699 -- Two Line WAN IOA
 - 2720 -- PCI WAN/Twinaxial IOA
 - 2721 -- PCI Two-Line WAN IOA

To find out what adapters are installed on your system, do this on the command line:

```
==> go hardware
```

The Hardware Resources Display shown in Figure 83 is shown.

```
HARDWARE                      Hardware Resources                      System:  SYSNAM
Select one of the following:
    1. Work with communication resources
    2. Work with local workstation resources
    3. Work with storage resources
    4. Work with processor resources
    5. Work with token-ring LAN adapter resources
    6. Work with DDI LAN adapter resources
    7. Work with all LAN adapter resources
    8. Work with coupled system adapter
    70. Related commands
Selection or command
==> 1
F3=Exit  F4=Prompt  F9=Retrieve  F12=Cancel  F13=Information Assistant
F16=AS/400 Main menu
(C) COPYRIGHT IBM CORP. 1980, 1996.
```

Figure 83. Hardware Display

Select option 1., to get the display shown in Figure 84 on page 129. Three 2609 communications adapters are configured on the 2623 communications processor. Each 2609 adapter has two ports, to which you can attach modems.

```

Work with Communication Resources
System:  SYSNAM
Type options, press Enter.
  2=Edit  4=Remove  5=Work with configuration descriptions

Opt Resource      Type Text
  CC02          2623 Comm Processor
    LIN08       2609 Comm Adapter
      LIN081    2609 V.24 Port Enhanced
      LIN082    2609 V.24 Port
    LIN05       2609 Comm Adapter
      LIN051    2609 V.24 Port Enhanced
      LIN052    2609 V.24 Port
    LIN06       2609 Comm Adapter
      LIN061    2609 V.24 Port
      LIN062    2609 V.24 Port
    CMB01       2615 Combined function IOP
More...
F3=Exit  F5=Refresh  F6=Print  F11=Display resource addresses/statuses
F12=Cancel

```

Figure 84. Hardware Support for SLIP-Example

Step 1 - Configure an Asynchronous Line Description

You must use the line description name when you add the dial-in or dial-out profile for the line in “Step 3 - Determine Configuration Profile Type” on page 132.

To check for existing asynchronous line descriptions, enter the following:

```
WRKLIND LIND(*ASYNC)
```

If line descriptions exist, AS/400 displays them as shown in Figure 85 on page 130. If no line descriptions exist, or if you prefer to create a new one, do either of the following:

- Press F6=Create from the Work with Line Descriptions display.
- Enter the CRTLINASC on the command line and press F4, then F9.

Continue at “Asynchronous Line Description Parameters” on page 152. Then return to “Step 2 - Configure AS/400 For Your Modem” on page 130.

```

                                Work with Line Descriptions
                                System:  SYSNAM
Position to . . . . .           Starting characters

Type options, press Enter.
  2=Change  3=Copy  4=Delete  5=Display  6=Print  7=Rename
  8=Work with status  9=Retrieve source

Opt  Line          Type  Text
-   ASCDIAL7      *ASYNC  Switched line to modem. DIALCMD(*OTHER)
-   ASCNONSW4     *ASYNC
-   ASCSWIT6      *ASYNC  Switched line to modem. DIALCMD(*OTHER)
-   ASCSWIT7      *ASYNC  Switched line to modem. DIALCMD(*OTHER)
-   MODEMLIN      *ASYNC  Sample line to modem in this book
-

                                                                Bottom

Parameters or command
===>
F3=Exit  F4=Prompt  F5=Refresh  F6=Create  F9=Retrieve  F12=Cancel
F14=Work with status

```

Figure 85. Work with Asynchronous Line Descriptions

Step 2 - Configure AS/400 For Your Modem

Check to see if a sample configuration exists for your modem. Type CFGTCPPTP, then select option 11 to view the default modem entries on AS/400. If your modem is listed, go on to the next configuration step. If your modem is not listed, you can do either of the following:

- Try the \$generic Hayes modem entry.
- Add an entry for your modem.

To determine which option is best, check the user manual for your modem. Look for the following settings for command strings:

- Modem reset to factory defaults: This is often AT&F or AT&Z
- Modem initialization – Some of the options you should set are as follows:
 - Display Verbal Result Codes: Often this is Q0 and V1
 - Normal CD and DTR modes: Often this is &C1 and &D2
 - Echo mode off: Often this is E0
 - Data Set Ready (DSR) to follow Carrier Detect: Often, this is &S1
 - Enable hardware flow control (RTS/CST)
 - Enable error correction and optionally, compression
 - Ensure DTE-DCE line speed is enabled to run at fixed 19.2K
 - If the modem supports it, optionally enable the inactivity timer
- Modem answer mode:
 - Answer after n rings: Often this option is S0=n
 - Disconnect if no carrier (connection) after m seconds: Often this option is S7=m
- Modem dial type: This is usually ATDT, which selects tone dialing. To compare, ATDP selects pulse dialing.

If your modem and the \$generic hayes modem entry match for these settings, then you can most likely use the \$generic hayes modem entry. If your modem supports different command entries, then you should create a new modem entry.

Note: Although there is much consistency between AT-compatible modems, each one often has unique settings. Therefore, be aware that these are recommendations. Your modem user's guide contains the details that you need.

To add new modem information, enter option 1:

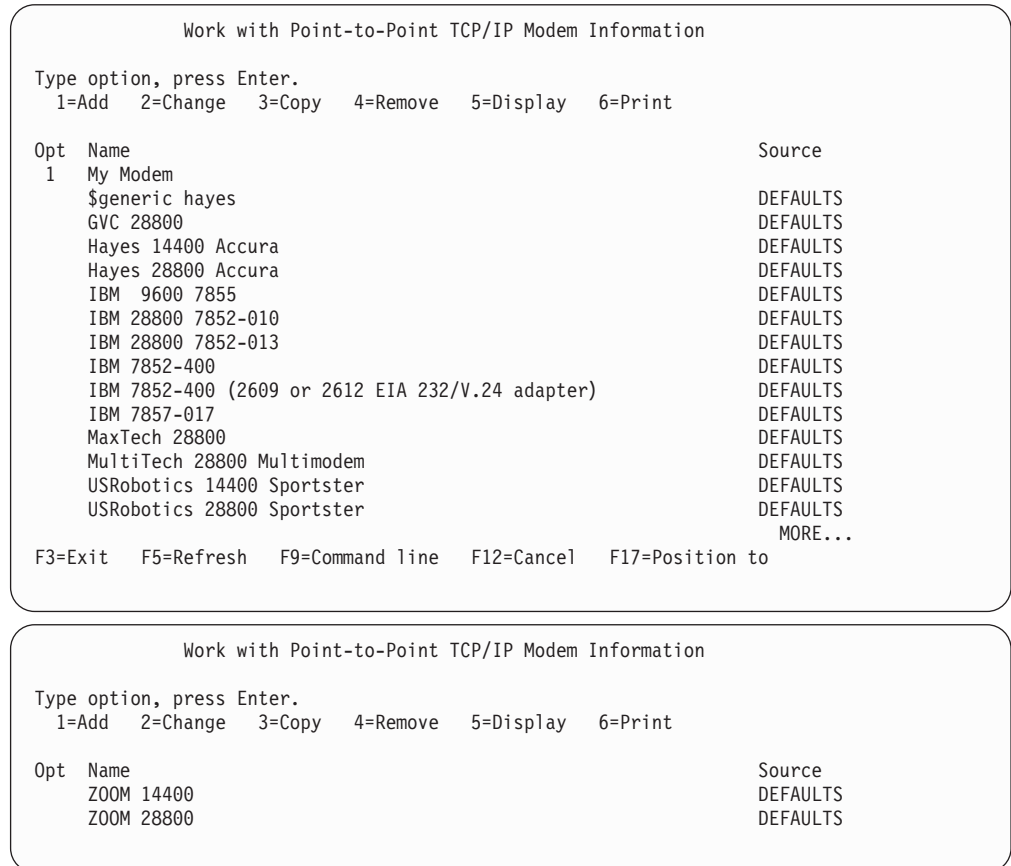


Figure 86. Adding a Modem Entry-Display 1

Figure 87 on page 132 shows the default values that you get if you add modem information for your modem. Change the defaults as needed.

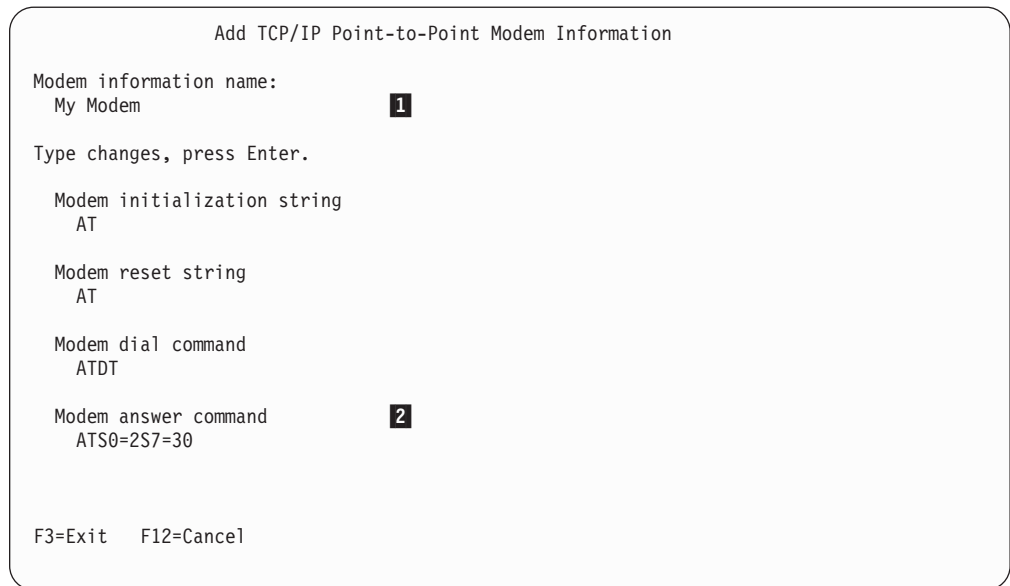


Figure 87. Adding a Modem Entry-Display 2

- 1** To add your own modem, you need the owner's manual from the modem to complete the prompts in this display.

Optionally, to add your own modem, copy the \$generic Hayes modem entry, and change the prompt information to match the information required for your modem.
- 2** The default modem answer command instructs the modem to answer an incoming call after one ring and then wait a maximum of 30 seconds for a connection. If you need more time to establish connections, you must change this command.

Step 3 - Determine Configuration Profile Type

There are two kinds of connection profiles, *ANS and *DIAL profiles. Use *ANS profiles for dial-in support. These profiles are called *ANS profiles because AS/400 uses them to answer incoming calls from remote systems.

Use *DIAL profiles for dial-out support. These profiles are called *DIAL profiles because AS/400 uses them to dial out to remote systems.

For each modem attached to AS/400, you need at least one configuration profile. For each line attached to the modem, you need an asynchronous line description.

If you are using the same modem to dial in and to dial out, you need two profiles for that modem. You need a *ANS profile for dial-in support, and a *DIAL profile for the dial-out support. If the modem has only one line attached, you need only one line description.

Only one caller at a time can use a given asynchronous line and its associated line description.

To add a dial-in profile, go to "Step 4 - Add a Dial-In (*ANS) Configuration Profile" on page 133.

To add a dial-out profile, go to “Step 5 - Add a Dial-Out (*DIAL) Configuration Profile”.

Step 4 - Add a Dial-In (*ANS) Configuration Profile

Before you add a profile, have the following information ready:

The IP address this profile uses for the remote system that is connecting to this AS/400 system.

The name of the *ASYNC line this connection will run over (from “Step 1 - Configure an Asynchronous Line Description” on page 129).

Modem information, if you are going to use a modem. (from “Step 2 - Configure AS/400 For Your Modem” on page 130).

The name of an authorization list, if you want to restrict use of the profile to a specific group of users. See “System Access Authorization List” on page 146 for details.

Other information is also required, depending on the choices you make when configuring the profile. Review the information about the dial-in profile, in “Dial-In (*ANS) Point-to-Point Profile Parameters” on page 140 to determine if you are ready to add the profile.

To add a profile:

1. Choose Option 1 - Work with point-to-point TCP/IP from the Configure Point-to-Point TCP/IP display.
2. Type 1 in the Opt column
3. Type a name for the profile
4. Type *ANS to add a dial-in profile

Step 5 - Add a Dial-Out (*DIAL) Configuration Profile

Dial-out connection profiles are called *DIAL profiles. You use them to dial out to remote systems. Before you start entering information on the following displays, you need to have some information ready. For example, write down the following:

The name of the *ASYNC line this connection will run over (from “Step 1 - Configure an Asynchronous Line Description” on page 129).

Modem information, if you are going to use a modem. (from “Step 2 - Configure AS/400 For Your Modem” on page 130).

The information for accessing the remote system, such as the modem telephone number, a user ID, and a password. See “Remote System Access Information” on page 152 for details.

Other information is also required, depending on the choices that you make when you configure the profile. To help determine if you need more information for your configuration profile, see the following:

- Figure 99 on page 147 contains a sample display for adding profile EEL.
- “Dial-Out Point-to-Point Profile Parameters” on page 147 contains information about each parameter.

To add the profile:

1. Type 1 in the Opt column

2. Type a name for the profile
3. Type *DIAL to create a dial-out profile

Step 6 - Start the Configuration Profile

Figure 88 shows a number of SLIP *DIAL profiles. Use the option 9=Start to start a profile.

```

Work with Point-to-Point TCP/IP

Type option, press Enter.
 1=Add      2=Change  3=Copy   4=Remove   5=Display details  6=Print
 9=Start    10=End    12=Work with line status  14=Work with session job

Opt  Name          Mode  Type  Status      Line  Line  Job
     Name          Mode  Type  Status      Description Type  Name

 9__ MOLEANS1      *ANS  *SLIP  INACTIVE    ASCSWIT6  *ASYNC
     NIMROD       *ANS  *SLIP  STRSSN      ASCSWIT4  *ASYNC  QTPPANS021
     PEBBLES      *ANS  *SLIP  RINGW       ASCSWIT8  *ASYNC  QTPPANS020
     AS8T07NSW   *DIAL *SLIP  INACTIVE    ASCNONSW3 *ASYNC
     BAMBAM      *DIAL *SLIP  INACTIVE    ASCNONSW3 *ASYNC
     DIALW31     *DIAL *SLIP  INACTIVE    ASCSWIT5  *ASYNC  QTPPDIAL90
     IBMIGNSP    *DIAL *SLIP  ACTIVE      ASCSWIT7  *ASYNC  QTPPDIAL93
     IBMIGNSP1   *DIAL *SLIP  INACTIVE    ASCSWIT7  *ASYNC  QTPPDIAL83
     IBMIGNSP2   *DIAL *SLIP  INACTIVE    ASCSWIT7  *ASYNC  QTPPDIAL89

More...

F8=Work with modems  F9=Command line  F10=Local interface status
F11=Display text    F12=Cancel      F14=Work with session jobs  F24=More keys

```

Figure 88. Work with Point-to-Point TCP/IP-Starting a SLIP Profile

Before you start a profile that is defined for a given line description, make sure that the line description is not already being used by a different profile. Only one profile at a time can use a given line.

When you start a profile by using option 9 from this display, you run the STRTCPPTP command with its default settings. Other settings are available for your use. For example, you may want to send an inquiry message that requires a response before AS/400 releases the job. For example, use the inquiry message if you are doing problem analysis and need to start the System Licensed Internal Code (SLIC) component trace, tracing the ASCII device and controller. If AS/400 is creating the device and controller names automatically at SLIP profile startup, you need a way to pause the SLIP session job so you can display the object names. Or you may want to keep the controller and device descriptions that AS/400 created automatically, so that the jobs can be reused.

If your configuration profile does not start, you can print the script dialog that occurs during start by specifying the *ERROR or *PRINT options on the STRTCPPTP command. For information about using the *ERROR and *PRINT parameters, see "Point-to-Point Jobs That Are Not Active" on page 139.

Monitoring Point-to-Point Activity

Use the Work with Point-to-Point TCP/IP (WRKTCPPTP) command to monitor and manage SLIP activity.

Options From WRKTCPPTP

Using Figure 89 as a reference, the following is a summary of selected management tasks you can do from this display. This information is a supplement to the information provided in the online help information for the various options. This information is not intended to be a complete summary of the online text. Be sure the profile that you want to change, copy, or remove is not active.

Work with Point-to-Point TCP/IP							
Type option, press Enter.							
1=Add	2=Change	3=Copy	4=Remove	5=Display details	6=Print		
9=Start	10=End	12=Work with line status	14=Work with session job				
Opt	Name	Mode	Type	Status	Line Description	Line Type	Job Name
___	MOLEANS1	*ANS	*SLIP	INACTIVE	ASCSWIT6	*ASYNC	
___	NIMROD	*ANS	*SLIP	STRSSN	ASCSWIT4	*ASYNC	QTPPANS021
___	PEBBLES	*ANS	*SLIP	RINGW	ASCSWIT8	*ASYNC	QTPPANS020
___	AS8T07NSW	*DIAL	*SLIP	INACTIVE	ASCNONSW3	*ASYNC	
___	BAMBAM	*DIAL	*SLIP	INACTIVE	ASCNONSW3	*ASYNC	
___	DIALW31	*DIAL	*SLIP	INACTIVE	ASCSWIT5	*ASYNC	QTPPDIAL90
_14	IBMIGNSP	*DIAL	*SLIP	ACTIVE	ASCSWIT7	*ASYNC	QTPPDIAL93
___	IBMIGNSP1	*DIAL	*SLIP	INACTIVE	ASCSWIT7	*ASYNC	QTPPDIAL83
___	IBMIGNSP2	*DIAL	*SLIP	INACTIVE	ASCSWIT7	*ASYNC	QTPPDIAL89

More...

F8=Work with modems F9=Command line F10=Local interface status
 F11=Display text F12=Cancel F14=Work with session jobs F24=More keys

Figure 89. Work with Point-to-Point TCP/IP-Working With a Specific Job

Option 1 (Add)

Use this option to add a new configuration profile. See “Step 3 - Determine Configuration Profile Type” on page 132 for information about the connection profiles.

Option 2 (Change)

Use this option to change the information associated with a configuration profile that is not active.

Option 3 (Copy)

Use this option to add a new profile that has the exact same characteristics as the one being copied.

Option 4 (Remove)

Use this option to remove a configuration profile that is not active.

Option 5 (Display)

Use this option to display a configuration profile.

Option 6 (Print)

Use this option to print the same information about the configuration profile that you see on the display when you select option 5.

Option 9 (Start)

Use this option to start a point-to-point session for a particular configuration profile. This option calls the Start Point-to-Point TCP/IP (STRTCPPTP) command with the name of the profile. If you enter the command and press F4=Prompt, you can specify:

- Which profile to start.
- Whether to capture any errors that occur while establishing the Point-to-Point session.
- Whether to capture the complete connection script dialog in a spooled file.

F8 (Work with modems)

Displays the Work with TCP/IP Point-to-Point Modem Information display (shown in Figure 86 on page 131).

F10 (Local interface status)

Displays the Work with TCP/IP Interface display. From this display you can display information about particular interfaces and associated routes, start or end a TCP/IP interface, or work with configuration status.

F14 (Work with session jobs)

Runs the Work with Active Jobs (WRKACTJOB) command for active jobs with job names that start with the characters QTPP. This displays a list of any active point-to-point TCP/IP jobs and allows you to work directly with these jobs. For more information about managing jobs, see “Working With Point-to-Point Jobs”.

Working With Point-to-Point Jobs

SLIP jobs are Point-to-Point jobs that run in the QSYSWRK subsystem. In general, you can work with the Point-to-Point jobs just as you work with any other jobs running on AS/400. The *Work Management* book contains detailed information about how to manage AS/400 jobs. You can work with individual SLIP jobs for profiles that have a job name associated with them as seen on the WRKTCPPTP display (Figure 89 on page 135). To work with a job, select 14=Work with session job. If you select option 14 against a profile that does not have a job associated with it, such as MOLEANS1, AS/400 displays an error message.

Point-to-Point Job Names: The Point-to-Point job names for SLIP start with the string QTPP, where dial-out jobs (*DIAL) are QTPPDIALnn, and Dial-In jobs (*ANS) are QTPPANSnnn. The number string nn or nnn is a sequential number from 1 to 99 for *DIAL jobs and 1 to 999 for *ANS jobs. The AS/400 system starts a new job for each SLIP profile that you start.

Point-to-Point Job Status Indicators: This topic summarizes the job status indicators that Point-to-Point jobs can have. Some of the status indicators show you what AS/400 is doing as a connection is starting, in use, or ending. Other status indicators show error conditions. The jobs displayed in Figure 89 on page 135 show some of the possible status indicators.

INACTIVE

No jobs are currently running for this profile. However, if a job is listed in under Job Name on the Work with Point-to-Point TCP/IP display, you can display the spooled file output for the job. To do this, select option 14 (Work with session job). Option 14 runs the WRKJOB command for the configuration profile.

SBMERR

An error occurred during the submission of the job. If a SBMJOB fails (SLIP status SBMERR), then no job name exists. The job name and job number are generated if the SBMJOB actually completes and the job is added to the job queue. If SBMERR occurs, look at the messages in the job log of the job that ran the STRTCPPTP command to determine what caused the error.

OUTQ

The connection has ended, and job information is available. Select option 14 to display the job information.

JOBQ

A request to start the connection was submitted. The job is waiting on the job queue.

STRSSN

A job is starting for this profile, but has not yet completed.

ENDSSN

The connection for this profile is ending.

JOBLOG

The connection for this profile has ended, and AS/400 is placing information in the job log for this job.

ADDTCPCFG

TCP/IP interface address information is being added for the connection.

RMVTCPCFG

TCP/IP interface information is being removed from the connection.

MSGW

An inquiry message was sent to either the QTCP or to the QSYSOPR message queue. Job connection processing is suspended until you respond to the inquiry message.

SSNERR

An error occurred while the connection was established. To find out more about the error, type option 14 next to the configuration profile name to run the WRKJOB command.

STRTCPMMN

Starting TCP/IP communications. AS/400 is starting the SLIP protocol jobs.

ENDTCPMMN

Ending TCP/IP communications.

DIAL

AS/400 is calling the remote system to establish the connection.

RINGW

For a *ANS session that is waiting for someone to call.

CNNDIALOG

Connection dialog information is being exchanged over this connection.

ACTIVE

Connection has been made, and the job is running.

ICFERR

An error occurred during intersystem communications function (ICF) communications. To find out more about the error, type option 14 next to the configuration profile name to run the WRKJOB command.

Active Point-to-Point Jobs: Figure 90 on page 138 shows some active point-to-point jobs that AS/400 displays when you do either of the following:

- Press F14 from the WRKTCPPTP display.
- Select option 2 from the Configure Point-to-Point TCP/IP display.

To display the jobs, AS/400 runs the WRKACTJOB command. If no SLIP profiles are started, AS/400 displays the following message:
No active jobs to display.

```

                                Work with Active Jobs

CPU %:  11.4    Elapsed time:  07:17:29    Active jobs:  59

Type options, press Enter.
  2=Change  3=Hold  4=End  5=Work with  6=Release  7=Display message
  8=Work with spooled files  13=Disconnect ...

Opt  Subsystem/Job  User      Type  CPU %  Function      Status
---  -
  ---  QTPPANS020  QTCP      BCH   .0    PGM-QTOCPPSM  ICFW
  ---  QTPPANS021  QTCP      BCH   .0    PGM-QTOCPPSM  RUN
  ---  QTPPDIAL93  QTCP      BCH   .0    PGM-QTOCPPSM  ICFW

                                                                Bottom

Parameters or command
===>
F3=Exit    F5=Refresh  F10=Restart statistics  F11=Display elapsed date
F12=Cancel F23=More options  F24=More keys

```

Figure 90. Work with Active Jobs-Displaying SLIP Jobs

Figure 91 shows the WRKJOB display for job QTPPDIAL93. For active point-to-point jobs, option 10 from the WRKJOB display shows you the current job log. You can use the job log to monitor the job while the job is running.

```

                                Work with Job

Job:  QTPPDIAL93    User:  QTCP          Number:  009919    System:  SYSNAM

Select one of the following:

  1. Display job status attributes
  2. Display job definition attributes
  3. Display job run attributes, if active
  4. Work with spooled files

 10. Display job log, if active or on job queue
 11. Display call stack, if active
 12. Work with locks, if active
 13. Display library list, if active
 14. Display open files, if active
 15. Display file overrides, if active
 16. Display commitment control status, if active

                                                                More...

Selection or command
===> 10

F3=Exit  F4=Prompt  F9=Retrieve  F12=Cancel

```

Figure 91. Working with a Job

Point-to-Point Jobs That Are Not Active: To work with point-to-point jobs that are not active, type option 14 next to the job in the Work with Point-to-Point TCP/IP display. Then select option 4 from the WRKJOB menu to work with the spooled file output from the job. The spooled file output includes a spooled file for the job log and possibly a spooled file for the connection script dialog that occurred between AS/400 and the remote system. AS/400 creates spooled file output for the connection script dialog for connections that:

- Use a connection script, and
- Run with the output parameter on the STRTCPPTP command set to either of the following:
 - *ERROR (the default), and an error occurs during the connection dialog between the local AS/400 system and the remote system.
 - *PRINT, which puts the complete dialog into a spooled file.

Figure 93 on page 140 contains an example of a spooled file that contains connection dialog.

Figure 92 shows the Work with Job Spooled Files (WRKSPLF) display. The display results when you use option 4 from the WRKJOB display for a SLIP job that is not active. In this case, job QTPPDIAL90, which is associated with SLIP profile DIALW31, has two spooled files. You can display the output by using option 5 (Display).

```

Work with Job Spooled Files

Job:  QTPPDIAL90   User:  QTCP           Number:  009524

Type options, press Enter.
  1=Send  2=Change  3=Hold  4=Delete  5=Display  6=Release  7=Messages
  8=Attributes  9=Work with printing status

  Opt  File          Device or      User Data      Status  Total  Current
     File          Queue         QTPPDIAL90    RDY     Pages  Page   Copies
  5  DIALW31       PRT01         QTPPDIAL90    RDY     2      1      1
  -  QPJOBLOG      QEZJOBLOG     QTPPDIAL90    RDY     6      6      1

                                                                 Bottom

Parameters for options 1, 2, 3 or command
===>
F3=Exit  F10=View 3  F11=View 2  F12=Cancel  F22=Printers  F24=More keys

```

Figure 92. Work with Job Spooled Files-Spooled Output for SLIP Jobs

The spooled file DIALW31 in Figure 92 is illustrated in Figure 93 on page 140.

- The leftmost column indicates the time a command, response, or informational message was recorded.
- The next column from the left indicates the type of information that is recorded.
 - === indicates that informational text follows
 - ==> indicates outbound text follows
 - <== indicates inbound text follows

- The remaining text on a line is one of the following:
 - Modem command or response
 - Informational message
 - Inbound or outbound text itself

Figure 93 shows an example of the information AS/400 places into the connection dialog script spooled file output.

```

                                Display Spooled File
File . . . . . : DIALW31                               Page/Line  1/6
Control . . . . .                               Columns   1 - 78
Find . . . . .
10:21:57 === Attempting modem reset.
10:21:57 ==> AT&FS0=0
10:21:57 === Reading modem response.
10:21:58 <==
10:21:58 <== OK
10:22:02 === Attempting modem initialization.
10:22:02 ==> AT&D2&C1X4V1Q0S7=70&#172;N6&K3%C1E0&S1
10:22:02 === Reading modem response.
10:22:02 <== AT&D2&C1X4V1Q0S7=70&#172;N6&K3%C1E0&S1
10:22:02 <== OK
10:22:06 === Attempting modem dial/answer.
10:22:06 ==> ATDT9,,752-4622
10:22:07 === Reading modem response.
10:22:12 === Reading modem response.
10:22:37 <==
10:22:37 <== CONNECT 19200/V42BIS
10:22:37 ==>
10:22:41 <== login:
10:22:41 ==> sliptest8
10:22:45 <== password:
10:22:45 ==> *****
10:22:45 === Establishing remote host connection.
10:27:08 === Remote host connection ended.

```

Figure 93. Work With Job Spooled Files-Spooled Output From a SLIP Job

You can use the connection script spooled file output to accomplish the following:

- Debug connection scripts.
- Ensure that the commands being sent to the modem are correct.
- Ensure that the expected modem responses are returned.

Dial-In (*ANS) Point-to-Point Profile Parameters

This topic provides explanations of the parameters available on the dial-in profile. Enter CFGTCPPTP on the command line to get the display shown in Figure 94 on page 141.


```

Work with Point-to-Point TCP/IP

Type option, press Enter.
1=Add    2=Change  3=Copy   4=Remove    5=Display details  6=Print
9=Start  10=End   12=Work with line status  14=Work with session job

Opt  Name      Mode  Type  Status      Line  Line  Job
     Name      Mode  Type  Status      Description  Type  Name
1    EDITH     *ANS

```

Figure 94. Add *ANS Configuration Profile for SLIP

Figure 95 shows the default entries you get when you first access the display.

```

Add TCP/IP Point-to-Point *ANS Profile
System:  SYSNAM

Name:  EDITH
Text
1

Type choices, press Enter.

TCP/IP information:
Protocol type . . . . . : *SLIP
Local interface address . . . . .
Remote IP address . . . . .
Maximum transmission unit . . . . . 576
Allow proxy ARP . . . . . N
Add default route . . . . . N

Physical line information:
Line description . . . . .
Line type . . . . . : *ASYNC
Autocreate controller and device Y
Remote location name . . . . .

F2=Change modem information  F3=Exit  F4=List  F9=Command line
F12=Cancel

```

Figure 95. Creating a *ANS Configuration Profile-Display 1

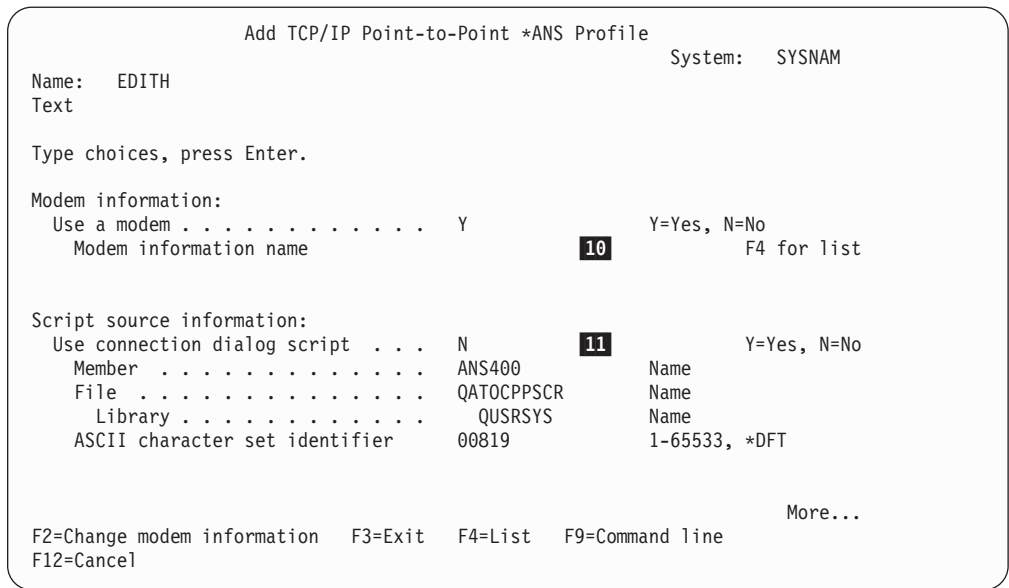


Figure 96. Creating a *ANS Configuration Profile-Display 2

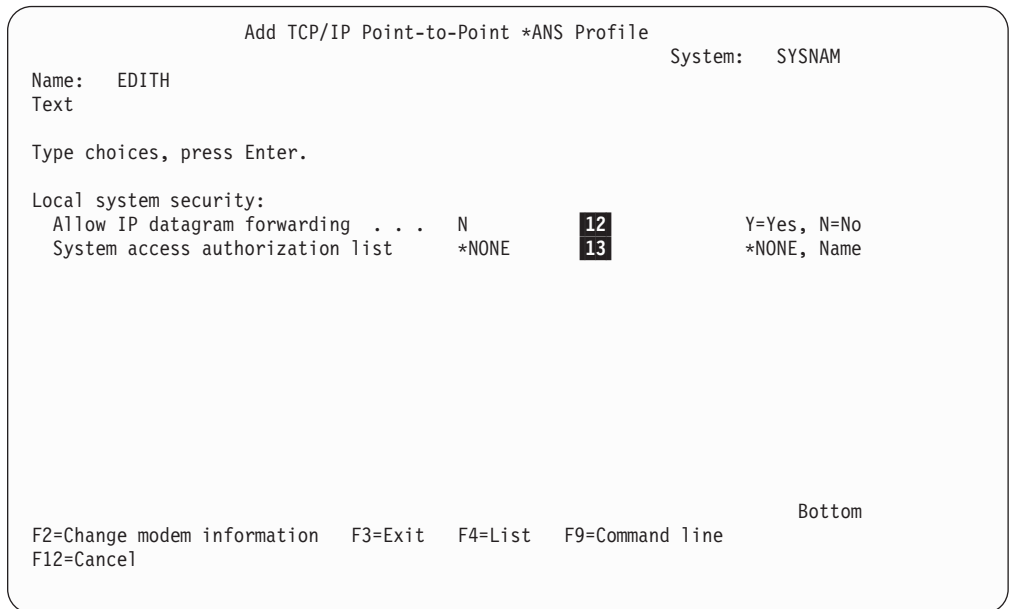


Figure 97. Creating a *ANS Configuration Profile-Display 3

Text:

- 1** Enter descriptive text for this configuration profile

Local Interface Address:

- 2** You can do one of the following:
 - Enter a new interface address.
 The new address can be:

- A new interface address, or
- An interface address on the same network as an existing interface.
- Choose an existing address.

To find these existing addresses, press F4 (List) when the cursor is in this field. Then select the interface you want to use.

Note: Using this option allows this AS/400 system to serve as a proxy for ARP requests for the remote IP address of the system dialing in.

For more information about proxy ARP, see “Allow Proxy ARP”.

Remote IP Address:

3 If you choose an existing interface address for the local interface address, then AS/400 uses the remote interface address for TCP/IP communications over the point-to-point connection. If you define a new local interface address, then AS/400 uses the remote interface address only for routing to the remote system.

If the remote system that is dialing in expects the IP address to be passed back to it, then the remote system uses the remote address defined here as its local interface address for the connection.

Note: If you choose to use proxy address resolution (ARP), then this remote IP address must be on the **same** subnet as the local interface address. See “Subnetworks and Subnet Masks” on page 6 for more information about subnets.

Maximum Transmission Unit:

4 This is the size in bytes of the largest packet that AS/400 can send over the physical line that this interface uses. Ensure that the size of the MTU is no larger than the maximum buffer size (MAXBUFFER) parameter that is specified for the *ASYNC line you select for this *SLIP interface.

Allow Proxy ARP:

5 Select Y if you plan to use the local interface for proxy ARP. The local interface must be an existing interface address.

Proxy ARP-Definition: Proxy ARP is a technique that allows one machine, the proxy agent, to answer ARP requests that are actually destined for a different machine. Proxy ARP is useful with SLIP because it allows a remote SLIP client to logically appear to be part of a local network. This provides automatic connectivity between the SLIP client and all hosts on a local network.

Proxy ARP-Example: Consider AS/400 with an IP address of 9.4.24.93, attached to the 9.4.24 subnet. A remote workstation dials in to AS/400. AS/400 assigns an IP address of 9.4.24.193. to the remote workstation. AS/400 sends data for 9.4.24.193 over the SLIP line. But to all other 9.4.24.x hosts on the local LAN, 9.4.24.193 appears to be attached to the local LAN. To send data to 9.4.24.193, the other hosts send an ARP request to 9.4.24.193. SLIP connections do not support ARP. Without proxy ARP, the remote workstation has no way to respond to the ARP request. If AS/400 is configured to allow proxy ARP, AS/400 answers the ARP request for 9.4.24.193. AS/400 receives the IP packet from the originating host, and forwards the packet on over the SLIP connection to the remote client.

If you want AS/400 to do this, set the Proxy ARP parameter (“Allow Proxy ARP” on page 143) to Y.

Notes:

1. Only interface types that support ARP can be used with Proxy ARP. Interface types that support ARP include token-ring, Ethernet, and FDDI. Point-to-point interface types such as X.25 and *SLIP do not support ARP.
2. Proxy ARP does not mean *proxy gateway*. PCs connected directly to the LAN cannot use AS/400 system as a gateway for Internet access through SLIP.

Be sure that the local interface is active before you start the configuration profile. The start fails if the local interface is not active when you start the point-to-point TCP/IP profile. To determine if the interface is active, type the following:

```
NETSTAT *IFC
```

Add Default Route:

- 6** Select Y to add a default route to the system. AS/400 adds the route when the SLIP session is started. The local interface address specified in this configuration profile is the next hop for the default route that is added.

Note: Only one default route can be active at a time. If you run the Start Point-to-Point TCP/IP (STRTCPPTP) command for a configuration profile that specifies Y to add a default route, and a default route is already active, the session cannot start.

Use this option for *ANS profiles if you want to do the following:

- Connect two networks together and use a SLIP connection to do so
- Use an automatic default route to the other network

Otherwise, do not automatically add routes for your *ANS profiles.

Line Description:

- 7** Type the name of the line description to be used for this connection.

See “Step 1 - Configure an Asynchronous Line Description” on page 129 for information about how to create or find the line description.

Autocreate Controller and Device:

- 8** Specify the default value of Y to have AS/400 automatically create the controller and device descriptions for the line description specified in 7 (“Line Description”). If you specify N, you must specify a remote location name. See “Remote Location Name” on page 145.

Notes:

1. The setting for this parameter does not affect and is not affected by any of the autoconfiguration system values (QAUTOCFG, QAUTORMT and QAUTOVRT).
2. AS/400 creates controller and device descriptions only if they do not already exist.

If Start Point-to-Point TCP/IP (STRTCPPTP) is issued with AUTODLTFCFG(*YES), then the controller and device description are deleted when the point-to-point connection ends. The next time you start a point-to-point connection using this profile the controller and device descriptions are created again.

However, if STRTCPPTP is issued with AUTODLTCFG(*NO), AS/400 does not delete the controller and device descriptions when the connection ends. The next time the profile is used the previously created controller and device descriptions are re-used.

Remote Location Name:

- 9** If Autocreate Controller and Device is N then this field must specify the remote location name defined in the device description of the controller and device description pair to use when Start Point-to-Point TCP/IP (STRTCPPTP) is issued for this configuration profile. If Autocreate Controller and Device is Y, then any name entered in this field is ignored.

Specify a value for this field only when using a specific controller and device description pair that you created.

Notes:

1. The value specified is the same value specified for the Create Asynchronous Device Description (CRTDEVASC) command RMTLOCNAME parameter on the device description you want to use when activating this point-to-point connection.
2. The controller and device description that you create must have status VARIED ON prior to starting the TCP/IP point-to-point connection.
3. The TCP/IP point-to-point connection cannot start if the device description is already in use.

Modem Information:

- 10** You must specify modem information if you are using a modem. For most modems, you can select one of the entries that appears when you press F4.

Note: You can change modem information by pressing F2 on the Add TCP/IP Point-to-Point *ANS Profile display. If no modem information is found, go to "Step 2 - Configure AS/400 For Your Modem" on page 130.

Use Connection Dialog Script:

- 11** The remote system that is dialing in must provide the information required by the server connection script. The remote system can do this by doing either of the following:

- Using a matching connection script on its system.
- Providing the information interactively.

Note: Not all systems support the option to provide the information interactively.

You determine whether a script is used by specifying Y or N on this parameter. The most common use of scripts is to exchange sign-on and password information before a remote client may connect to AS/400 system. Another common use is to dynamically assign an IP address to the remote SLIP client. See "NLS Considerations" on page 125 for information about the ASCII character set identifier used in scripts.

Allow IP Datagram Forwarding:

- 12** Use IP datagram forwarding to forward IP datagrams that come from a remote host, but are not meant for the local IP address.

The default is N (No). Datagrams from the remote system that are not destined for this address are discarded.

Note: You can disable datagram forwarding for all TCP/IP interfaces on this system by using the command CHGTCPA IPDTGFWD(*NO). When you do this, the value set in configuration profiles is ignored, and no IP datagrams are forwarded. Once a remote system is connected to AS/400 and the user signs on to AS/400 using TELNET or FTP, then the user can access the other systems in the network that is connected to this AS/400 system. The user has access because the interface address for the user is now from this AS/400 system and not from the original remote client.

To determine the current value for system datagram forwarding, enter CHGTCPA and press F4. The AS/400 prompter will show the current value for the parameter.

For information about security for AS/400 support of the SLIP protocol, see the *Tips and Tools for Securing Your AS/400* book.

PING-ing your local IP address: After establishing a point-to-point connection with a remote system, it is typical to try to PING both the remote and local IP addresses defined for the connection. This is done to ensure the connection to the remote system is actually operational. If the connection is complete, the PING to the remote IP address should complete successfully. However, a PING to your local IP address may or may not work, depending on whether the remote system forwards IP datagrams or not.

When a PING is done on a local IP address for point-to-point links such as SLIP, X.25, and so on, the PING ECHO request will actually leave the local system and travel to the remote system. The remote system will then look at the PING ECHO request and determine that the PING address is not its own. If the remote system is capable of forwarding IP datagrams, it will resend the PING ECHO request back out over the point-to-point link to the local system. When the local system receives the PING ECHO request, it determines that the PING address is its own and replies back with a PING ECHO reply completing the PING request. However, if the remote system does not do IP datagram forwarding, then a PING of your local IP address will not work since the PING ECHO request will be thrown out.

PING time metrics will normally show that a PING to the local IP address takes twice as long to complete as a PING to the remote IP address because all PING requests to the local address have to travel through the remote system first.

System Access Authorization List:

- 13** Enter the name of an AS/400 authorization list if you want to allow only the user profiles that are specified in the authorization list to connect to this AS/400 from a remote system over SLIP.

Note: If 'Use connection dialog script' (see "Use Connection Dialog Script" on page 145) is set to N (No), you must set the system access authorization list to *NONE. Authorization list checking is only done as part of a connection dialog script.

Use the Create Authorization List (CRTAUTL) command to create an authorization list. Use the Add Authorization List Entry (ADDAUTLE) command to add user profile entries to the list.

For more information about how and when to use authorization lists, see the *Tips and Tools for Securing Your AS/400* book.

Dial-Out Point-to-Point Profile Parameters: To create a dial-out (*DIAL) profile, enter WRKTCPPPTP. AS/400 displays the Work with Point-to-Point TCP/IP display shown in Figure 98. Enter the following:

1. Option 1 to add a profile.
2. A name for the profile.
3. *DIAL for a dial-out profile.

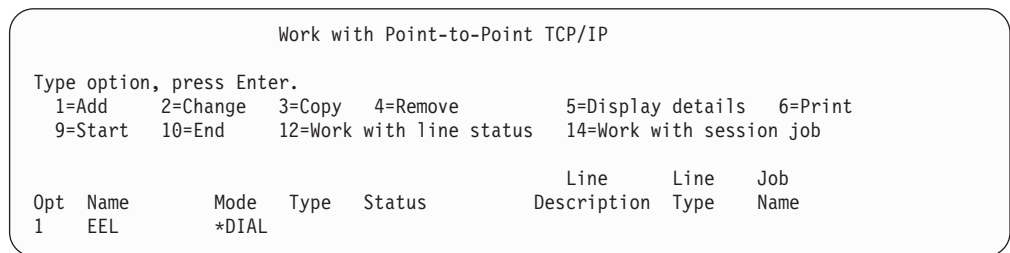


Figure 98. Add *DIAL Configuration Profile for SLIP

Figure 99, Figure 100, Figure 101, and Figure 102 show the default entries that AS/400 displays for adding a *DIAL profile.

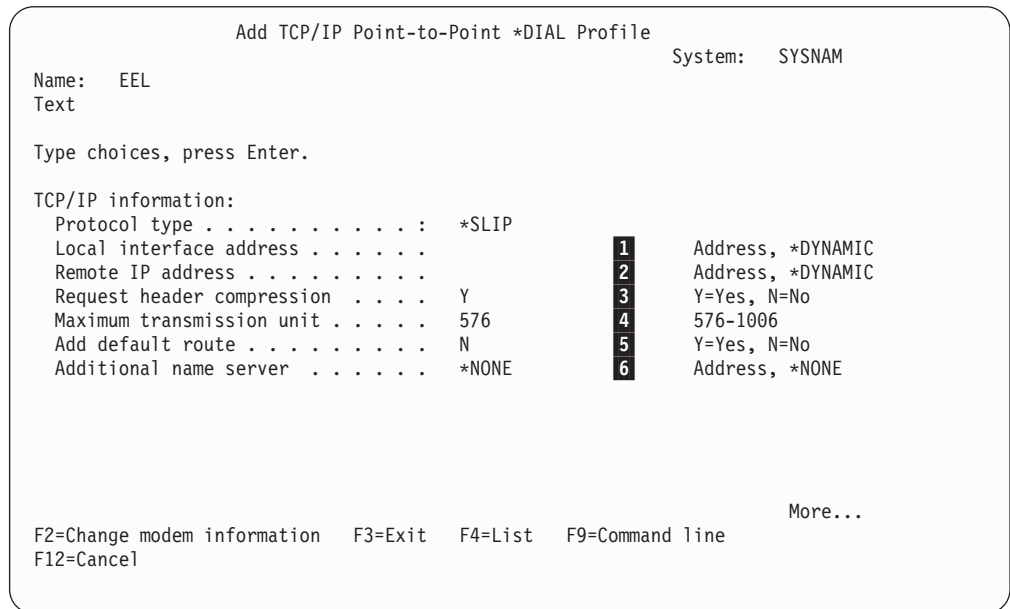


Figure 99. Creating a *DIAL Configuration Profile-Display 1

```

Add TCP/IP Point-to-Point *DIAL Profile
System:  SYSNAM

Name:  EEL
Text

Type choices, press Enter.

Physical line information:
Line description . . . . . 7      Name
Line type . . . . . : *ASYNC      8      Y=Yes, N=No
Autocreate controller and device Y    9      Name
Remote location name . . . . .

Modem information:
Use a modem . . . . . Y    10     Y=Yes, N=No
Modem information name      F4 for list

F2=Change modem information  F3=Exit  F4=List  F9=Command line
F12=Cancel

More...

```

Figure 100. Creating a *DIAL Configuration Profile-Display 2

```

Add TCP/IP Point-to-Point *DIAL Profile
System:  SYSNAM

Name:  EEL
Text

Type choices, press Enter.

Script source information:
Use connection dialog script . . . N    11     Y=Yes, N=No
Member . . . . . DIAL400      Name
File . . . . . QATOCPPSCR     Name
Library . . . . . QUSRSYS     Name
ASCII character set identifier 00819  1-65533, *DFT

F2=Change modem information  F3=Exit  F4=List  F9=Command line
F12=Cancel

More...

```

Figure 101. Creating a *DIAL Configuration Profile-Display 3

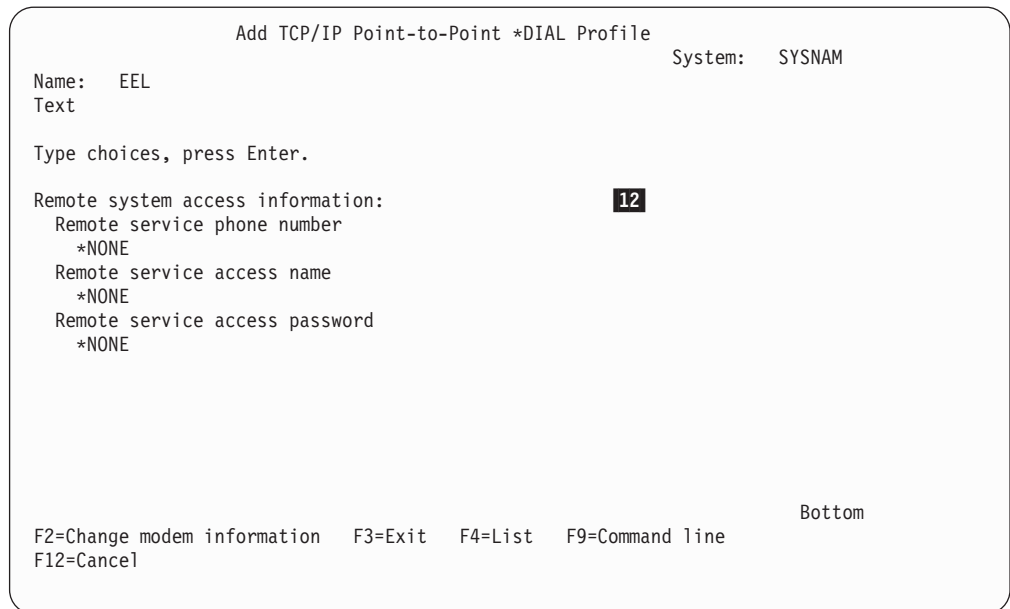


Figure 102. Creating a *DIAL Configuration Profile-Display 4

Local Interface Address:

1 This address is the local interface address to be used by this AS/400 for point-to-point communications.

You can do one of the following:

- Enter a new interface address.
- Enter a new interface address on the same network as an existing interface when you want to tie two AS/400 systems together.
- Choose *DYNAMIC.

Use the special value *DYNAMIC when the remote system specifies a local interface address for your system. If you specify *DYNAMIC, you must use a connection dialog script. Both the connection dialog script and the remote system must support IP address passing.

Remote IP Address:

2 Enter the IP address of the remote system for this profile. This address is the local interface address on the remote system.

Specify *DYNAMIC to allow the remote address to be determined dynamically. You need this special value when the remote system specifies a remote IP address for your system.

Note: *DYNAMIC requires use of a connection dialog script.

Request Header Compression:

3 The default, Y, indicates this system compresses header information when it establishes the point-to-point connection. If the remote system does not support the header compression, which is called Van Jacobson (VJ) header

compression, ¹ AS/400 establishes a session that does not use header compression. Therefore, you should not need to change this value.

Using header compression improves the performance of the serial line, which is usually slow anyway.

Maximum Transmission Unit:

- 4** The maximum transmission unit (MTU) is the size in bytes of the largest packet that is to be sent over the physical line that this interface uses.

Note: The MTU size cannot be larger than the maximum buffer size (MAXBUFFER) on the *ASYNC line description that this *SLIP interface uses.

Add Default Route:

- 5** If you change this value to Y (Yes), AS/400 adds a default route to the remote system when the session is started. The local interface address specified in this configuration profile becomes the next hop for the default route that is added. Turn this option off when you know to which host this AS/400 system will connect.

Note: There can be only one active default route. If the Start Point-to-Point TCP/IP (STRTCPPTP) command is issued for a configuration profile that specifies Y for *Add default route* field, and a default route is already active, the session cannot start.

Additional Name Server:

- 6** Enter the IP address of a name server that is accessed through the point-to-point connection to the remote system.

If you specify a name server address, AS/400 adds the name server address to the end of any existing list of name servers when this configuration profile is started. This field is valid only for *DIAL point-to-point TCP/IP profiles.

If you use an ISP to connect to the Internet, you may receive a name server address from the ISP. Then when you connect, you can use host names instead of IP addresses for remote sites.

If you add an additional name server for this connection, then you must fully qualify any host names on this connection. Otherwise, the domain of your local AS/400 is appended to the unqualified host name. If this occurs, the additional name server cannot find the host entry.

Note: There can only be one active point-to-point session with an additional name server address defined. If the Start Point-to-Point TCP/IP (STRTCPPTP) command is issued for a profile that specifies Y for the *Additional Nameserver* field, and a point-to-point profile is already active with an additional name server address, the session cannot start. The default value for this field is *NONE. The list of name servers is not changed.

This name server provides limited support when connecting to a remote network. Most sockets applications that were activated prior to establishing

1. SLIP support with header compression is also called CSLIP, or Compressed Serial Line Internet Protocol.

the *DIAL connection to a remote system will not see this name server unless they are ended and restarted. In particular, you will need to restart SMTP after establishing a remote connection in order to use a name server that has been dynamically added.

Line Description:

7 Type the name of the line description to be used for this connection.

See “Step 1 - Configure an Asynchronous Line Description” on page 129 for information about how to create or find the line description.

Autocreate Controller and Device:

8 Specify the default value of Y to have AS/400 automatically create the controller and device descriptions for the line description specified in 7 (“Line Description”). If you specify N, you must specify a remote location name (see “Remote Location Name”).

Notes:

1. The setting for this parameter does not affect and is not affected by any of the autoconfiguration system values (QAUTOCFG, QAUTORMT and QAUTOVRT).
2. AS/400 creates controller and device descriptions only if they do not already exist.

If Start Point-to-Point TCP/IP (STRTCPPTP) is issued with AUTODLTCFG(*YES), then when the point-to-point connection ends the controller and device description are deleted. The next time you start a point-to-point connection using this profile the controller and device descriptions are created again.

However, if STRTCPPTP is issued with AUTODLTCFG(*NO), AS/400 does not delete the controller and device descriptions when the connection ends. The next time the profile is used the previously created controller and device descriptions are re-used.

Remote Location Name:

9 If Autocreate Controller and Device is N then this field must specify the remote location name defined in the device description of the controller and device description pair to use when Start Point-to-Point TCP/IP (STRTCPPTP) is issued for this configuration profile. If Autocreate Controller and Device is Y, then any name entered in this field is ignored.

Specify a value for this field only when using a specific controller and device description pair that you created.

Notes:

1. The value specified is the same value specified for the Create Asynchronous Device Description (CRTDEVASC) command RMTLOCNAME parameter on the device description you want to use when activating this point-to-point connection.
2. The controller and device description that you create must have status VARIED ON prior to starting the TCP/IP point-to-point connection.
3. The TCP/IP point-to-point connection cannot start if the device description is already in use.

Modem Information:

- 10** You must specify modem information if you are using a modem. For most modems, you can select one of the entries that appears when you press F4.

If your modem does not appear in the list, go to “Step 2 - Configure AS/400 For Your Modem” on page 130.

Script Source Information:

- 11** You determine whether a connection script is to be used by specifying Y or N. If you specify Y, the remote system that is dialing in must provide the information required by the server connection script that you specify. The remote system can do this by either of the following:

- Using a matching connection script on its system.
- Providing the information interactively.

Note: Not all systems support the option to provide the information interactively.

The most common uses of scripts are for exchanging sign-on and password information before permitting the remote client to connect to AS/400 system. Another common use is to dynamically assign an IP address to the remote SLIP client. If you do not require these functions, you can bypass script processing.

For information about the ASCII character set identifier, see “NLS Considerations” on page 125.

For examples of scripts, see “Connection Dialog Scripts” on page 156.

Remote System Access Information:

- 12** The remote system access information consists of the following:

- The remote service telephone number.
- The remote service access name.
- The remote service access password.

The remote service telephone number is the number that the modem calls to contact the remote system. Use the remote service access name to provide a user ID when the remote system requires this information before allowing you to connect. If the remote system requires an account name in addition to the user ID, then enter both values into the Remote service access name field, separated by a blank as follows:

```
account userID
```

Use the remote service access password when the remote system also requires you to provide a valid password for the user ID or account name.

If the remote system requires a user ID and password information, then you must use a connection script and include the user ID and PASSWORD keywords.

For examples of scripts, see “Connection Dialog Scripts” on page 156.

Asynchronous Line Description Parameters: There are many parameters for AS/400 Create Line Description Asynchronous (CRTLINASC) command. This topic

describes the parameters that you must specify to use an asynchronous line description with AS/400 TCP/IP point-to-point configuration profiles. This topic also discusses parameters and parameter values that differ from the command defaults. For more information about the parameters options for the CRTLINASC command, see the *Communications Configuration* book.

Two examples of commands to create asynchronous line descriptions follow. The line MODEMLIN using resource LIN011 is an example for a line that is connected to a modem. The line description NOMODEMLIN using resource LIN022 is an example of a direct connection. That is, the physical line for NOMODEMLIN is directly connected to the remote system through a null modem adapter.

CRTLINASC Parameters When Using a Modem: The following topic discusses a command that was used to create an asynchronous line description that is named MODEMLIN. MODEMLIN uses resource LIN011. It was created to be used with AS/400 point-to-point TCP/IP. Figure 103 on page 155 shows the parameters that you should use for an asynchronous line that is attached to a modem.

```
CRTLINASC  LIND(MODEMLIN)  RSRcname(LIN011)
            LINESPEED(19200) MAXBUFFER(1500)
            CNN(*SWTPP)     DIALCMD(*OTHER)
            SWTCNN(*DIAL)   AUTOANS(*NO)
            AUTODIAL(*YES)  INACTTMR(*NOMAX)
```

LIND(MODEMLIN)

The name of the line description- you specify this name in any AS/400 TCP/IP point-to-point configuration profile that uses this line.

RSRCNAME(LIN011)

The unique name that is assigned by AS/400 to identify the physical communications port attached to your system. This example uses LIN011 as the name AS/400 has associated with the communications port. For information about how to determine the resource name you need, see "Hardware Requirements for the Asynchronous Line Description" on page 128.

LINESPEED(19200)

The line speed of the asynchronous line in bits per second (bps). If your modem supports data rate conversion, you should be able to use line speed 19200. Most modems that support error correction also support data rate conversion. If your modem does not support data rate conversion, then the asynchronous line speed must match the rate that the modem connects to the remote system.

Note: The 19,200 line speed is used for illustration here because this value can be specified on any AS/400 system using any adapter and IOP combination. If both your modem and AS/400 hardware support a higher line speed, use this value instead.

See "Step 2 - Configure AS/400 For Your Modem" on page 130 for more information on configuring your modem.

MAXBUFFER(1500)

The maximum size of any single data packet sent across the line. This value must always be at least as large as the value specified for the Maximum Transmission Unit (MTU) in any TCP/IP point-to-point profile that uses this line.

Note: The value MAXBUFFER(1500) is used in this example because the value 1500 is larger than any MTU value that could be specified in a TCP/IP point-to-point configuration profile.

CNN(*SWTPP)

Specifies that this is a switched point-to-point asynchronous line. The line description must be specified as a switched line when using a modem.

DIALCMD(*OTHER)

Specifies that this asynchronous line is attached to a modem that uses the Hayes AT command set. Only AT command modems are supported for use with AS/400 TCP/IP point-to-point connection profiles. You must use this value if SLIP is to use the asynchronous line.

SWTCNN(*DIAL)

You must specify this value when you specify DIALCMD(*OTHER).

Note: Do not confuse the value specified for SWTCNN with the operating mode of your configuration profile. It does not matter whether line description will be used with an operating mode *DIAL configuration profile or a *ANS profile. You must always specify SWTCNN(*DIAL) for the line.

AUTOANS(*NO)

Specifies that the support in AS/400 line description for automatic answering is not used. You must specify this value when you specify SWTCNN(*DIAL).

Note: AS/400 TCP/IP Point-to-Point uses the Auto Answer Mode capability of your modem to answer incoming calls to an *ANS Operating Mode configuration profile. AUTOANS(*NO) does not affect the ability of AS/400 SLIP to respond to an incoming call.

AUTODIAL(*YES)

You must specify this value when you specify DIALCMD(*OTHER).

INACTTMR(*NOMAX)

You must specify this value when you specify DIALCMD(*OTHER).

```

Create Line Desc (Async) (CRTLINASC)

Type choices, press Enter.

Line description . . . . . LIND > MODEMLIN
Resource name . . . . . RSRNAME > LIN011
Online at IPL . . . . . ONLINE *YES
Physical interface . . . . . INTERFACE *RS232V24
Connection type . . . . . CNN> *SWTPP
Switched network backup . . . . . SNBU *NO
Vary on wait . . . . . VRYWAIT *NOWAIT
Autocall unit . . . . . AUTOCALL *NO
Data bits per character . . . . . BITSCHAR 8
Type of parity . . . . . PARITY *NONE
Stop bits . . . . . STOPBITS 1
Duplex . . . . . DUPLEX *FULL
Echo support . . . . . ECHO *NONE
Line speed . . . . . LINESPEED > 19200
Modem type supported . . . . . MODEM *NORMAL
Switched connection type . . . . . SWTCNN> *DIAL

More...

F3=Exit F4=Prompt F5=Refresh F12=Cancel F13=How to use this display
F24=More keys

```

```

Create Line Desc (Async) (CRTLINASC)

Type choices, press Enter.

Autoanswer . . . . . AUTOANS> *NO
Autodial . . . . . AUTODIAL> *YES
Dial command type . . . . . DIALCMD> *OTHER
Autocall resource name . . . . . ACRSRNAME
Calling number . . . . . CALLNBR *NONE
Inactivity timer . . . . . INACTTMR> *NOMAX
Maximum buffer size . . . . . MAXBUFFER > 1500
Flow control . . . . . FLOWCNTL *NO
XON character . . . . . XONCHAR 11
XOFF character . . . . . XOFFCHAR 13
End-of-Record table: EORTBL
End-of-Record character . . . . . 00
Trailing characters . . . . . 0
+ for more values

```

Figure 103. Creating an Asynchronous Line Description for a Line With a Modem-Example

CRTLINASC Parameters For a Direct Connection: This topic provides a sample asynchronous line description for a direct connection. The line description name is NOMODEMLIN. NOMODEMLIN uses resource LIN022. It was created to be used with AS/400 point-to-point TCP/IP. This example assumes that asynchronous ports of AS/400 and the remote system are directly connected through a NULL modem adapter.

```

CRTLINASC LIND(NOMODEMLIN) RSRNAME(LIN022)
          LINESPEED(19200) MAXBUFFER(1500)
          CNN(*NONSWTPP)

```

LIND(NOMODEMLIN)

The name of the line description- you specify this name in any AS/400 TCP/IP point-to-point configuration profile that uses this line.

RSRNAME(LIN022)

The unique name that AS/400 has assigned to identify the physical communications port attached to your system. This example uses LIN022 is the

name AS/400 has assigned to the communications port. To determine the resource name you need, see "Hardware Requirements for the Asynchronous Line Description" on page 128.

LINESPEED(19200)

The line speed of the asynchronous line in bits per second (bps). The line speed you specify must be compatible with the modem attached to the line. In this example, the line speed is set to 19,200 bps.

Note: When using a direct connection, both systems **must** specify the **same** value for the line speed.

MAXBUFFER(1500)

The maximum size of any single data packet sent across the line. This value must always be at least as large as the value specified for the Maximum Transmission Unit (MTU) in any TCP/IP point-to-point profile that uses this line.

Note: The value MAXBUFFER(1500) is used in this example because the value 1500 is larger than any MTU value that could be specified in a TCP/IP point-to-point configuration profile.

CNN(*NONSWTPP)

Specifies that this is a non-switched point-to-point asynchronous line. The line description must be specified as a non-switched line when two systems are directly connected with a NULL modem adapter.

Connection Dialog Scripts: To control whether AS/400 uses a connection script, you configure the Use connection dialog script field on the point-to-point configuration profile. Valid values for this field are:

- N** No login sequence is required beyond the physical modem connection.
- Y** A connection dialog script is required. Each connection script contains a text template that outlines the exchange of authorization and connection parameters with a remote host.

Security: For information about security for AS/400 support of the SLIP protocol, see the *Tips and Tools for Securing Your AS/400* book.

PPP/SLIP over *PPP

You must use the Operations Navigator to configure and administer PPP and SLIP profiles using a *PPP linetype.

If you use the command line interface to control Dial-in and Dial-out operations, you can still use these functions:

- STRTCPPTP to START *PPP profiles
- ENDTCPPTP to END *PPP profiles
- WRKTCPPTP to Work with *PPP profiles, but only in a limited capacity:
 - 9 (Start)
 - 10 (End)
 - 12 (Work with line status)
 - 14 (Work with session job)

Note: You can perform these panel options **only** by using AS/400 Operations Navigator: 2 (Change), 3 (Copy), 4 (Remove), and 5 (Display details).

Basically, you can control the operational aspects of PPP from the command line interface, but for configuration tasks that are PPP specific, you need to use Operations Navigator.

```

Work with Point-to-Point TCP/IP

Type option, press Enter.
 1=Add      2=Change  3=Copy   4=Remove  5=Display details  9=Start  10=End
12=Work with line status  14=Work with session job

Opt  Name      Mode  Type  Status      Line  Line  Job
     Name      Mode  Type  Status      Description  Type  Name

-----
BAMBAM  *ANS  *SLIP  STRSSN      ANSWERIT4  *ASYN  QTPPANS002
BARNEY  *ANS  *SLIP  ACTIVE      ANSWERIT1  *ASYN  QTPPANS001
BETTY   *ANS  *PPP  INACTIVE    ANSWERIT2  *PPP   QTPPANS003
DINO    *ANS  *PPP  OUTQ        ANSWERIT5  *PPP   QTPPANS010
PEBBLES *ANS  *SLIP  CALLW      ANSWERIT3  *PPP   QTPPANS004
FRED    *DIAL *SLIP  INACTIVE    DIALOMATIC *ASYN  QTPPANS004
WILMA   *DIAL *PPP  ACTIVE      ALTDIAL    *PPP   QTPPDIAL01

Bottom

F3=Exit  F5=Refresh  F9=Command line  F10=Local interface status
F11=Display status  F12=Cancel  F14=Work with session jobs  F24=More keys

```

Figure 104. Sample Work with Point-to-point TCP/IP panel with *PPP profiles

Chapter 5. Telnet Client

Important note: A thorough and in-depth explanation of Telnet is beyond the scope and purpose of this document. The majority of material on the Telnet client application is covered in the **AS/400e Information Center** under the **TCP/IP** topic. For more information see “TCP/IP Topics in the Information Center” on page xv.

The topics that remain in this chapter include conceptual and reference information on the Telnet client 3270 and VTxxx full-screen modes.

5250 Full-Screen Mode Considerations

This material is covered in the **AS/400e Information Center** under the **TCP/IP** topic. For more information see “TCP/IP Topics in the Information Center” on page xv.

TN5250—Start TCP/IP Telnet Command

The following optional parameters on the STRTCPTELN or Telnet command are applicable during a 5250 full-screen mode session:

- Timeout wait for host (INZWAIT)
- Keyboard language type (KBDTYPE) (See “TN3270 or TN5250—Specifying Keyboard and Character Sets” on page 160)
- Port number of the remote host server application (PORT)

TN5250—Screen Size

Telnet 5250 full-screen mode supports the following screen sizes:

- 1920-character (24 x 80) on all 5250 display stations.
- 3564-character (27 x 132) on 3180 Model 2; 3197 Models D1, D2, W1, W2; and 3477 Models FA, FC, FD, FE, FG, FW.

3270 Full-Screen Mode Considerations

The 3270 full-screen mode is activated by negotiating 327x workstation support with the remote Telnet server application. A typical example of a Telnet client 3270 mode environment is shown in Figure 105.

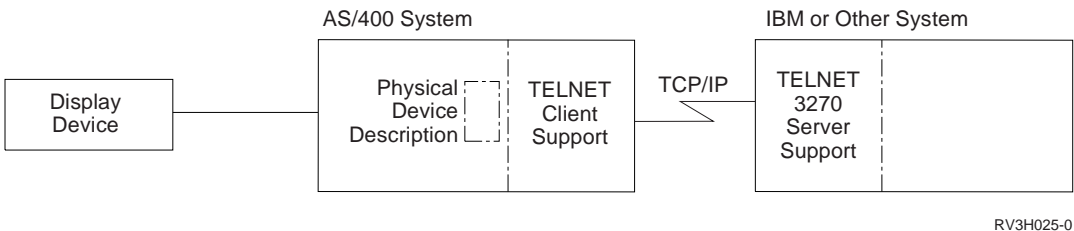


Figure 105. Typical Telnet 3270 Mode Environment (Telnet Client)

3270 full-screen support is negotiated with any Telnet server application that supports 3270 full-screen (rather than 5250) applications. An example of such a

system is the System/370 or System/390* system. All workstation types are negotiated to the 3278 Model 2 workstation when Telnet is in 3270 full-screen mode. The exceptions to this rule are:

- Remote 3277 workstations are negotiated as a 3277 Model 2
- Remote 3279 workstations are negotiated as a 3279 Model 2

When the Telnet session is started, your display station is controlled by the remote system application. You receive the same displays and enter data in the same way that you do for other 3270 devices attached to the remote system.

TN3270—Start TCP/IP Telnet Command

The following optional parameters on the STRTCPTELN or Telnet command are applicable during a 3270 full-screen mode session:

- Keyboard language type (KBDTYPE) (See “TN3270 or TN5250—Specifying Keyboard and Character Sets”)
- Page up (roll down) key (PAGEUP)
- Page down (roll up) key (PAGEDOWN)
- Cursor select key (CSRSLT)
- Outgoing 3270 translation table (TBL3270OUT)
- Incoming 3270 translation table (TBL3270IN)
- Timeout wait for host (INZWAIT)
- Numeric lock keyboard (NUMLCK)
- Change how nulls are handled (NULLS)

Using a Display Station during Telnet 3270 Full-Screen Mode

When using a display station during a Telnet 3270 full-screen session, you should be aware of keyboard and display differences. Other special considerations for Telnet 3270 mode include number of input fields, error messages, and ending a session.

TN3270 or TN5250—Specifying Keyboard and Character Sets

The keyboard language type you specify for your work station, using the keyboard language type parameter on the STRTCPTELN command, must be the same as the keyboard language type parameter of the remotely attached workstation. If you specify a keyboard language type that does not match, some of the characters are not displayed as expected.

For a description of different keyboard language types, see the *Local Device Configuration* book.

5250 and 3270 Keyboards

The placement and function of keys is different on the 5250 keyboard (3196G, 3180 Model 2, or 5291) than on the 3278 keyboard. Differences between these keyboards are shown in the *3270 Device Emulation Support* book.

Note: For the Telnet client operating in a 3270 full-screen mode, the 3270 Clear function defaults to the key sequence Shift-Cmd-Backspace.

Keyboard differences for the following keyboards are shown in the *System Operation for New Users* book.

- IBM-enhanced keyboard
- 122-key typewriter keyboard
- 5250 keyboard
- Personal computer or personal computer AT style keyboard
- Personal computer or personal computer AT 5250 style keyboard
- IBM-enhanced personal computer keyboard

Personal Computer Keyboards

If your personal computer uses the Client Access Workstation Function (WSF), you can display the layout of your 5250 keyboard using the Work Station Function Keys (WSFKEYS) command. You can alter the style using the Configure Work Station Function (CFGWSF) command. These commands are discussed in the *Client Access/400 for DOS with Extended Memory Setup* book. If your personal computer does not use the workstation function, refer to the appropriate documentation for your emulator (for example, OS/2 CM/2) to view or change the keyboard style.

TN3270—Minus Sign

If you specified the value *YES for the numeric lock keyboard parameter of the STRTCPTELN command, if you are using a data entry keyboard, and if the cursor is located in a numeric-only field, then do the following to display a minus sign.

To display a 5250 minus sign:

1. Press the Num (Numeric) key.
2. Press the minus sign (–) key.

To display a 3278 minus sign, press the minus sign key.

TN3270—Page Down and Page Up

If the 3270 application has a display that does not allow all the input data fields to be viewed, use the 5250 Page Down and Page Up keys to enter data when the maximum number of input fields on the display is exceeded.

You can also assign PF and PA functions to the page keys by specifying their use on the STRTCPTELN command.

The cursor always appears as an underline on both 5250 and 3270 displays.

TN3270—Screen Size

Telnet 3270 full-screen mode requirements:

- If the negotiated 3270 device type requires 1920 characters, the AS/400 Telnet client code will run with any 5250 device type as the client terminal.
- If the negotiated 3270 device type requires 3564 characters, the AS/400 Telnet client code requires either a 3180 Model 2, 3197 Model D1, D2, W1, W2, or 3477 Model FA, FC, FD, FE, FG, or FW 5250 device type as the client terminal.

TN3270—Cursor Select Key

The existing Cursor Select key is disabled if you choose to emulate the Cursor Select key. The Cursor Select key is emulated if you specify one of the following parameters for the STRTCPTELN command:

Parameter	Value
Page Up (Roll Down) key	*CSRSLT
Page Down (Roll Up) key	*CSRSLT
Cursor Select key	*F-key (specify a function key *F1 to *F24)

TN3270—Messages

Several types of error messages may be displayed when you are using Telnet 3270 full-screen mode.

- Key entry errors appear as flashing 4-digit numbers on the lower left corner of the display. Press the Help key or F1 (Help) to obtain more information about the message. See the *System Operation* book if you cannot correct the error.
- System messages include Telnet messages and are issued from the AS/400 system.
- For information on messages sent from the remote system, see the remote system documentation.

TN3270—Handling Null Characters

All null characters are removed when a data stream is sent from a 3270 display station. Specify one of the following values for the handle nulls (NULLS) parameter on the STRTCPTELN command:

***REMOVE**

Removes beginning and embedded null characters

***BLANK**

The default value; changes beginning and embedded null characters to blanks

Trailing null characters are always removed for both values. For example, assume the data consists of the following (0 indicates a null):

```
0x0yz000
```

The data stream sent from a 5250 display station running Telnet 3270 full-screen with the default *BLANK would contain the following:

```
bxbyz
```

The data stream sent from a 3270 display station or from a 5250 display station running a Telnet 3270 full-screen session when the value *REMOVE is specified would contain the following:

```
xyz
```

The value *REMOVE is valid for the following devices:

- Any locally attached display
- Displays attached to a remote 5394 controller
- Personal computer displays using the workstation function

VTxxx Full-Screen Mode Considerations

VT220 and VT100 terminals are ASCII full-screen terminals manufactured by Digital Equipment Corporation (DEC) and are the full-screen terminal types supported by AS/400. A typical example of a Telnet client VTxxx mode environment is shown in Figure 106.

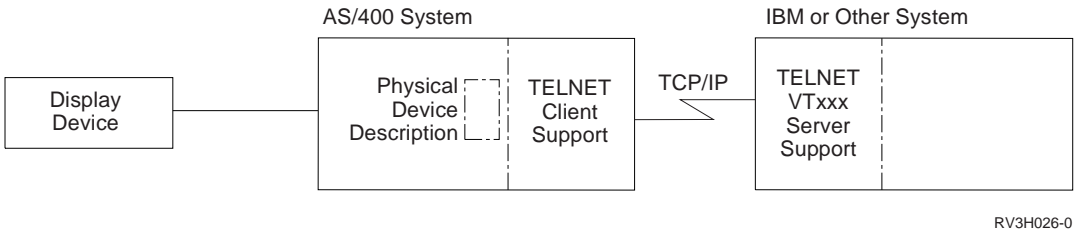


Figure 106. Typical Telnet VTxxx Mode Environment (Telnet Client)

When the VT220 terminal type is negotiated, there are several operating modes that are supported:

- **VT200 mode, 7-bit controls** is the default mode and uses the standard ANSI functions. This mode provides the full range of VT220 capabilities in an 8-bit communications environment with 7-bit controls. This mode supports the DEC** multinational character set or national replacement character (NRC) sets, depending on the character set mode selected.
- **VT200 mode, 8-bit controls** uses the standard ANSI functions and provides the full range of VT220 capabilities in an 8-bit communications environment with 8-bit controls. This mode supports the DEC multinational character set or national replacement character (NRC) sets, depending on the character set mode selected.
- **VT100 mode** uses standard ANSI functions. This mode restricts the use of the keyboard to VT100 keys. All data is restricted to 7 bits, and only ASCII, national replacement characters (NRC), or special graphics characters are generated.
- **VT52 mode** uses DEC private functions (not ANSI). This mode restricts the use of the keyboard to VT52 keys.

If VT220 mode is negotiated, then an initial operating mode for Telnet client is selected using the ASCII operating mode (ASCOPRMOD) parameter of the STRTCPTELN or Telnet command.

Telnet VTxxx support allows AS/400 users to sign on to non-AS/400 systems as if they were on a VTxxx terminal locally attached to the system and to access full-screen VTxxx applications. VTxxx client support allows an AS/400 user to sign on to any remote system in a TCP/IP network that supports the VTxxx terminal data stream.

Operational Differences

As an AS/400 Telnet user, you should be aware of physical and operational differences between VTxxx and 5250 terminals.

The 5250 is a block mode terminal. Data typed on a 5250 is accumulated in a buffer and only sent to the AS/400 system when an AID (attention identifier) key is

pressed. An AID key on a 5250 keyboard is a key that initiates a function. The following are the AID keys on a 5250 keyboard:

- Clear
- Command Function 1 through 24
- Enter/Rec Adv
- Help
- Print
- Record Backspace Function
- Roll Down (Page Up)
- Roll Up (Page Down)

VTxxx terminals operate in a character mode. Characters are sent immediately to the host when a key is pressed.

Another difference is the way the data arrives on the display. Data is written to a VTxxx terminal one character at a time, and you see the data arrive as streams of characters. With the 5250, data is written in blocks, and all or part of the display changes at once.

Keyboard Issues

You should avoid using the 5250 cursor movement keys. Instead, you should use the function keys associated with the *CSRUP, *CSRDOWN, *CSRRIGHT, and *CSRLEFT keywords. By default these are keys F13, F14, F15 and F16 respectively. If you use the 5250 cursor movement keys, the VTxxx application you are using may not function as expected because the results of using these keys are not transmitted to the remote system until an attention identifier (AID) key is pressed.

For example, using Telnet to the RISC System/6000 and obtaining VT220 emulation, the SMIT command provides a menu driven interface to AIX. Here the function keys associated with *CSRxx keywords perform as you would expect the cursor movement keys to do. However, the 5250 cursor movement keys, while physically moving the cursor down the screen and correctly selecting the SMIT option, do not cause the selected option to be highlighted. The highlighting in reverse video remains with the first option on the SMIT menu, regardless of the key position.

Typing a control character on an AS/400 keyboard is different than typing a control character on an actual VTxxx terminal. On a VTxxx terminal, the control key is pressed and held down while the character associated with the control function is pressed. For example, the VTxxx Control-C function is entered by pressing the following key sequence:



Figure 107. VTxxx Control-C Sequence

When using the AS/400 Telnet support, the equivalent is achieved by typing a two-character control indicator followed by pressing the function key associated with the *SENDWOCR (Send without Carriage Return) default function (the F11 key).

For example, if the default keyboard map and the default STRTCPTELN command parameters are in effect, the VTxxx Control-C function can be entered by typing &C followed by pressing the F11 key. (It can also be entered simply by <F12> using the default keyboard map, but in case you are using an application where <F12> is remapped, this example is included, and illustrates the principle of the *SENDWOCR key.

The character used to indicate a control character can be selected on the CTLCHAR parameter of the STRTCPTELN command. The default is &. The &C characters must be the last characters typed before pressing the *SENDWOCR function key or the &C is not interpreted as a control character. A control character is only sent when the *SENDWOCR function key is pressed. You can assign frequently used VTxxx control characters to a function key. An example of the Ctrl-C command can be described as follows. When using Telnet client to connect to an RS/6000 system, VT220 emulation will typically be negotiated. The Ctrl-C sequence is an important one in AIX to end long running commands, such as PING. It is, therefore, important that you know how to do this before issuing any RS/6000 commands. By default the sequence is &C <F11>. Note that these keys have to be entered quickly, and it may take several attempts before the RS/6000 task accepts the input.

If you do not want the characters that are being typed to be displayed, the function key associated with the *HIDE function should be pressed (F6 on the default keyboard map). This function should be used when typing a password.

If you want the characters that have been typed to be sent to the remote system for processing without pressing the Enter key, you should press the function key associated with the *SENDWOCR function (F11 on the default keyboard map).

It is often useful to be able to recall previously entered commands. On the AS/400 F9 often provides this function. On AIX, this can be activated by typing the command set -o vi and pressing Enter. After this, you can start retrieving commands with the sequence EscK. To perform this sequence using the default keyboard map while in VTxxx emulation, you should use the sequence <F5>k<F11>. The Esc character starts the command retrieval. The k can then be used to retrieve further commands. While operating in this mode, the commands H for right, L for left, X for delete, I for insert, and R for replace apply. The sequence <F5>i<F11> switches this facility off.

Screen Issues

The character in the position just before the cursor position will always be blank. The actual character is saved internally and is displayed when the display is refreshed with the cursor in a different position.

A VTxxx application that uses row 1, column 1 of the display does not work the same when using AS/400 Telnet client support. Most 5250-type display stations do not allow input to row 1, column 1. If the VTxxx application positions the cursor at row 1, column 1, the AS/400 system puts the cursor at row 1, column 2, automatically.

Due to architectural differences, certain VTxxx commands or sequences are not supported and are ignored. An example is downstream loadable character sets.

VTxxx—Screen Size

Telnet VTxxx full-screen mode supports the following screen sizes:

- On 3180 display stations:
 - 24 x 80 VTxxx screens should display as 24 x 80.
 - 24 x 132 VTxxx screens should display as 24 x 132.
- On 5250 display stations:
 - 24 x 80 VTxxx screens should display as 24 x 80.
 - 24 x 132 screens require the function key assigned to *SHIFTDSP (F10 on the default keyboard map) to move the information on the screen right or left.

VTxxx—Character Attributes

A VTxxx terminal supports the following attributes:

- Blink
- Bold
- Reverse video
- Underline
- Any combination of the above

The 5250 data stream supports the previous attributes so that all of the VTxxx attributes can be represented on a 5250 display station. However, there are some limitations.

- The 5250 data stream can only support three of the character attributes at the same time. If all VTxxx attributes are selected at the same time by the remote system, the underline, blink, and reverse video attributes are displayed. Also, the combination of underline, bold, and reverse image cannot be displayed on a 5250 display station. When a VTxxx application selects this combination, underline and reverse image are displayed.
- The attribute byte takes up a space on the 5250 display stations that do not support extended attributes. Attributes do not take up space on a VTxxx terminal. This means that you do not see all of the data shown on the 5250 display if character attributes are selected. When VTxxx data is received that is to be displayed with character attributes, the position before the data is overlaid with the 5250 attribute byte. The character that was displayed there is lost. If a character is to be displayed in row 1, column 1 with the attributes set, that character is not displayed. You can choose not to have the character attributes displayed by specifying DSPCHRATTR(*NO) on the STRTCPTELN command. This allows you to see all of the data on the display without attributes.

Note: This restriction is not applicable for displays that support extended attributes such as the 3477 display.

VT100—Keyboard Indicator

A VT100 terminal has an L1 indicator that can be programmed for different applications. This indicator is not emulated by the AS/400 Telnet support.

VTxxx—Start TCP/IP Telnet Command

The following optional parameters on the STRTCPTELN or Telnet command are applicable during a VTxxx full-screen mode session:

- Control characters (CTLCHAR)
- Incoming ASCII translation table (TBLVTIN)
- Outgoing ASCII translation table (TBLVTOUT)
- Special table out (TBLVTDRWO)
- Special table in (TBLVTDRWI)
- Options selected (VTOPT)
- Display character attributes (DSPCHRATTR)
- Page scroll feature (PAGE_SCROLL)
- Answerback feature (ANSWERBACK)
- Tab stops (TABSTOP)
- Timeout wait for host (INZWAIT)
- Coded character set identifier (CCSID)
- ASCII operating mode (ASCOPRMODE). This parameter applies to initializing a VT220 session only and has no effect on negotiations. See “VTxxx Full-Screen Mode Considerations” on page 163 for a description of the operating modes that are supported.
- Port number of the remote host server application (PORT)

If unexpected characters appear, the remote server system may not be configured correctly. Some server systems use a workstation-type system value to define the capabilities of the client workstation. If you are connected to such a server system, verify that the workstation-type value is set to an appropriate value for a VTxxx full-screen mode workstation.

You can also use the *set term* command to change the full-screen mode of the connection. For example, if you use Telnet to sign on to a RISC System/6000 (RS/6000) system, you should type:

```
set term=vt100
```

after the connection has been established. This should correctly map the unexpected characters.

Changing the VTxxx Keyboard Map

The client session support for both the VT100 and VT220 modes provides a primary and alternate keyboard mapping. This method of providing the keyboard mapping is done to accommodate the additional keypad capabilities of the VT220 mode. All changes to these keyboard mappings can be saved for later sessions using the F6 key from the Change VTxxx ... Keyboard Map display. The data is saved in the user profile, and once saved will automatically apply the next time Telnet VTxxx emulation is activated.

The keyboard option that you select from the Send Telnet Control Functions menu determines which keyboard mapping you use. The following displays show the VTxxx functions that correspond to the 5250 AID key.

- Option 6 (Change VT100 Primary Keyboard Map), shown in Figure 108 on page 168 and Figure 109 on page 169.
- Option 7 (Change VT100 Alternate Keyboard Map), shown in Figure 110 on page 169 and Figure 111 on page 170.
- Option 8 (Change VT220 Primary Keyboard Map), shown in Figure 112 on page 170 and Figure 113 on page 171.

- Option 9 (Change VT220 Alternate Keyboard Map), shown in Figure 114 on page 171 and Figure 115 on page 172.

The level of support negotiated between the AS/400 system and the server system determines which options are displayed on the Send Telnet Control Functions menu. If the VT100 full-screen mode support is negotiated initially, options 6 and 7 are displayed. If the VT220 full-screen mode support is negotiated initially, options 8 and 9 are displayed.

If you have previously installed TCP/IP at Version 2 Release 2, and upgrade to a later version, and if you previously used VT100 emulation, then if the server system is capable of negotiating the VT220 level of support, then the VT100 keyboard map previously used on the AS/400 system is no longer used. Instead the VT220 keyboard map would be used.

Note: There are no differences in the default values of the VT100 primary and alternate keyboard mappings.

The following figures show the default keyboard mappings. You can change any of the values. If you press the Enter key, your changes are saved for the current session only. If you press F6 (Save), your changes are permanently saved and are in effect the next time you start a VTxxx Telnet session.

```

Change VT100 Primary Keyboard Map
Type changes, press Enter:
5250 key          VT100 function
Function Key 1 . . . *PF1
Function Key 2 . . . *PF2
Function Key 3 . . . *PF3
Function Key 4 . . . *PF4
Function Key 5 . . . *ESC
Function Key 6 . . . *HIDE
Function Key 7 . . . *TAB
Function Key 8 . . . *CTLA
Function Key 9 . . . *CTLB
Function Key 10 . . *SHIFTDSP
Function Key 11 . . *SENDWOCR
Function Key 12 . . *CTLC
Function Key 13 . . *CSRUP
Function Key 14 . . *CSRDOWN
Function Key 15 . . *CSRRIGHT
Function Key 16 . . *CSRLEFT

F3=Exit  F6=Save  F12=Cancel

More...
```

Figure 108. Change VT100 Primary Keyboard Map (Display 1)

```
Change VT100 Primary Keyboard Map
Type changes, press Enter:
5250 key          VT100 function
Function Key 17 . . *CTLD
Function Key 18 . . *CTLE
Function Key 19 . . *CTLF
Function Key 20 . . *CTLG
Function Key 21 . . *CTLH
Function Key 22 . . *CTLI
Function Key 23 . . *CTLJ
Function Key 24 . . *CTLK
Rollup key . . . . *CTLL
Rolldown key . . . . *CTLM

F3=Exit  F6=Save  F12=Cancel

Bottom
```

Figure 109. Change VT100 Primary Keyboard Map (Display 2)

```
Change VT100 Alternate Keyboard Map
Type changes, press Enter:
5250 key          VT100 function
Function Key 1 . . . *PF1
Function Key 2 . . . *PF2
Function Key 3 . . . *PF3
Function Key 4 . . . *PF4
Function Key 5 . . . *ESC
Function Key 6 . . . *HIDE
Function Key 7 . . . *TAB
Function Key 8 . . . *CTLA
Function Key 9 . . . *CTLB
Function Key 10 . . *SHIFDSP
Function Key 11 . . *SENDWOCR
Function Key 12 . . *CTLC
Function Key 13 . . *CSRUP
Function Key 14 . . *CSRDOWN
Function Key 15 . . *CSRRIGHT
Function Key 16 . . *CSRLEFT

F3=Exit  F6=Save  F12=Cancel

More...
```

Figure 110. Change VT100 Alternate Keyboard Map (Display 1)

```

Change VT100 Alternate Keyboard Map
Type changes, press Enter:
5250 key          VT100 function
Function Key 17 . . *CTLD
Function Key 18 . . *CTLE
Function Key 19 . . *CTLF
Function Key 20 . . *CTLG
Function Key 21 . . *CTLH
Function Key 22 . . *CTLI
Function Key 23 . . *CTLJ
Function Key 24 . . *CTLK
Rollup key . . . . *CTLL
Rolldown key . . . . *CTLM

F3=Exit  F6=Save  F12=Cancel

Bottom

```

Figure 111. Change VT100 Alternate Keyboard Map (Display 2)

You can switch between the primary and alternate keyboard mappings during a VTxxx session using the function key assigned to the *KEYPRI and *KEYALT keywords. You can assign these keywords to any of the available 5250 function keys. It is recommended that you assign *KEYPRI to the Page Up 5250 function key and *KEYALT to the Page Down 5250 function key for both the primary and alternate keyboard mappings.

```

Change VT220 Primary Keyboard Map
Type changes, press Enter:
5250 key          VT220 function
Function Key 1 . . . *PF1
Function Key 2 . . . *PF2
Function Key 3 . . . *PF3
Function Key 4 . . . *PF4
Function Key 5 . . . *ESC
Function Key 6 . . . *HIDE
Function Key 7 . . . *TAB
Function Key 8 . . . *CTLA
Function Key 9 . . . *CTLB
Function Key 10 . . *SHIFDSP
Function Key 11 . . *SENDWOCR
Function Key 12 . . *CTLC
Function Key 13 . . *CSRUP
Function Key 14 . . *CSRDOWN
Function Key 15 . . *CSRRIGHT
Function Key 16 . . *CSRLEFT

F3=Exit  F6=Save  F12=Cancel

More...

```

Figure 112. Change VT220 Primary Keyboard Map (Display 1)

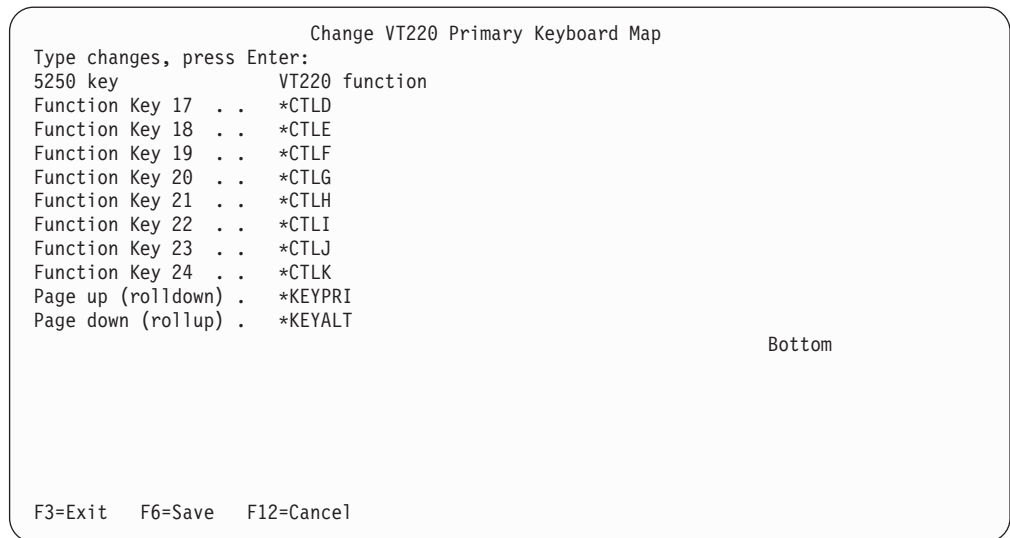


Figure 113. Change VT220 Primary Keyboard Map (Display 2)

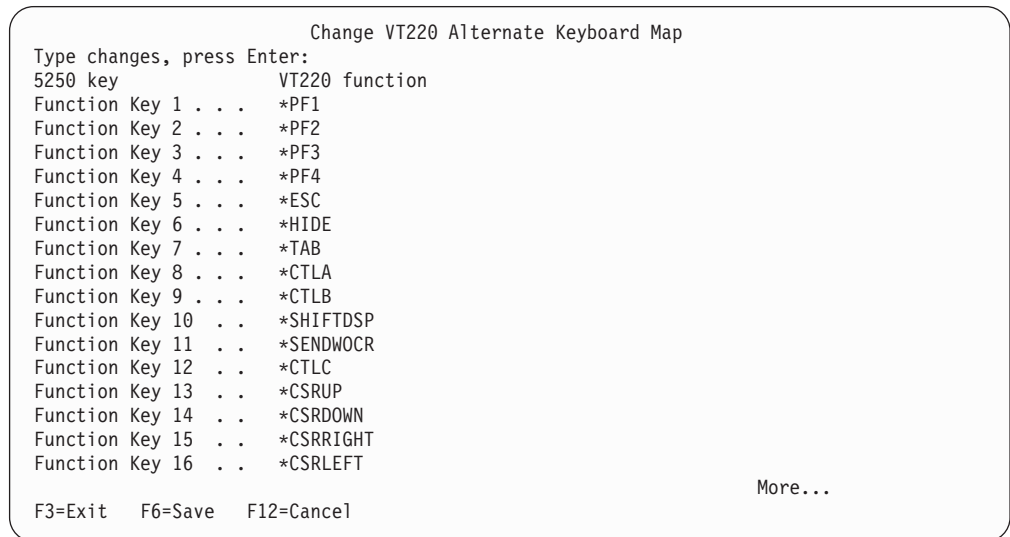


Figure 114. Change VT220 Alternate Keyboard Map (Display 1)

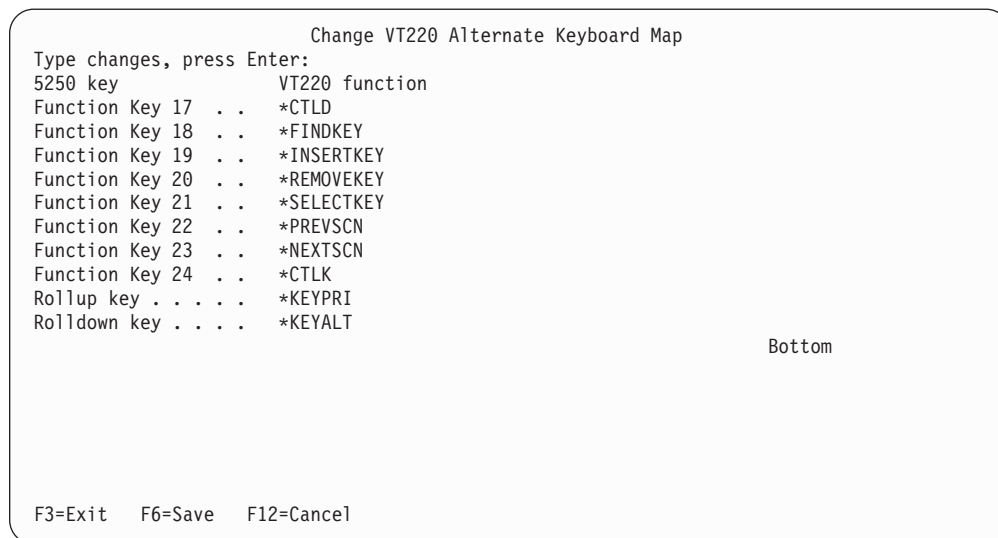


Figure 115. Change VT220 Alternate Keyboard Map (Display 2)

There are several types of VTxxx information that you can enter.

- **Character data.** You can assign a character string to a function key. For example, you are on the AS/400 system and are using Telnet to establish a connection with an RS/6000 system. To assign the character string set term=vt100 to the following function key:

```
Function Key 24 . . *CTLK
```

from the AS/400 system you would type:

```
Function Key 24 . . 'set term=vt100'
```

This allows you to press a function key rather than always having to type in that character string.

When you press the function key during a VTxxx session, the character string assigned to the function key is sent to the remote system with the carriage return, line feed characters added. If you type data before pressing the function key, the character string is added to the data that you type. This allows you to assign a frequently used command string to a function key. The character data typed is mapped from EBCDIC to ASCII before being transmitted to the remote system.

- **Control key keywords.** You can assign a VTxxx control keystroke to a function key using a defined keyword. For example, if you wanted to assign a different VTxxx control keystroke to the following function key:

```
Function Key 24 . . *CTLK
```

you would type:

```
Function Key 24 . . *CTLZ
```

When you press the function key, the new control character assigned to the function key is sent to the remote system. If you type data before pressing the function key, the control character is added to the typed data and sent to the remote system.

- **Hexadecimal data.** You can assign a hexadecimal string to a function key. When you press the function key, the hexadecimal data is sent to the remote system.

The carriage return, line feed characters are not added to hexadecimal data. If you type data before pressing the function key, the hexadecimal data is added to the typed data and sent to the remote system. This allows you to type a character that is not on the 5250 keyboard (for example, square brackets). Hexadecimal data is entered by typing X followed by a quoted string of hexadecimal characters, for example, X'1A1A'. The hexadecimal data is not mapped before being transmitted to the remote system.

- **Local AS/400 control functions.** You can assign a keyword to be handled locally within the AS/400 Telnet client session. These assignments or mappings may not result in ASCII data stream traffic being sent to the remote Telnet server session. These local control functions are *HIDE, *SHIFTDSP, *KEYPRI, and *KEYALT. The *SENDWOCR function is a local function, but ASCII data streams are sent to the remote Telnet server session.

Figure 116 shows the VT100 keyboard. Figure 117 on page 174 shows the VT220 keyboard. Table 10 on page 174 shows the valid control codes that are sent. The CTRL key is used in conjunction with other keys on the keyboard to generate control codes.

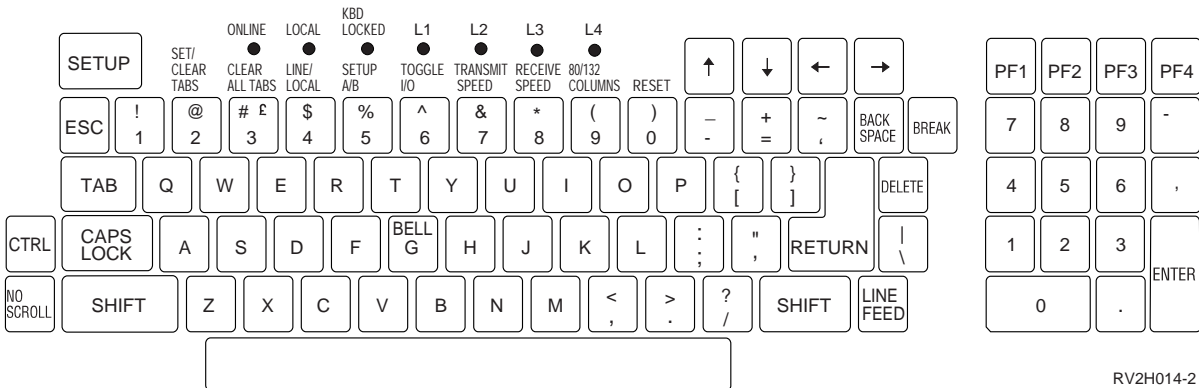
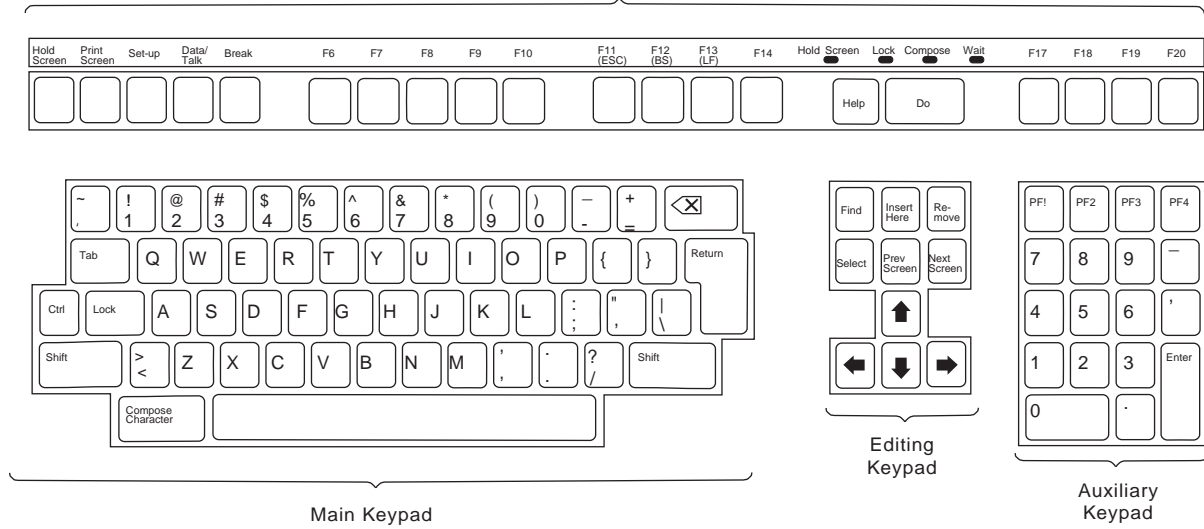


Figure 116. VT100 Keyboard

RV2H014-2

Function Keys



RV2N452

Figure 117. VT220 Keyboard

Table 10. VT100 and VT220 Control Character Keywords

Control Character Description	Key Pressed with CTRL Key Down	Keyword	Hex Character Transmitted
Null	Spacebar	*NUL	X'00'
Start of heading	A	*SOH,*CTLA	X'01'
Start of text	B	*STX,*CTLB	X'02'
End of text	C	*ETX,*CTLC	X'03'
End of transmission	D	*EOT,*CTLD	X'04'
Enquire	E	*ENQ,*CTLE	X'05'
Acknowledge	F	*ACK,*CTLF	X'06'
Bell	G	*BEL,*CTLG	X'07'
Back Space	H	*BS,*CTLH	X'08'
Horizontal tabulation	I	*HT,*CTLI	X'09'
Line feed	J	*LF,*CTLJ	X'0A'
Vertical tab	K	*VT,*CTLK	X'0B'
Form feed	L	*FF,*CTLL	X'0C'
Carriage return	M	*CR,*CTLM	X'0D'
Shift out	N	*SO,*CTLN	X'0E'
Shift in	O	*SI,*CTLO	X'0F'
Data link escape	P	*DLE,*CTLP	X'10'
Device control 1	Q	*DC1,*CTLQ	X'11'
Device control 2	R	*DC2,*CTLR	X'12'
Device control 3	S	*DC3,*CTLS	X'13'
Device control 4	T	*DC4,*CTLT	X'14'
Negative acknowledgement	U	*NAK,*CTLU	X'15'

Table 10. VT100 and VT220 Control Character Keywords (continued)

Control Character Description	Key Pressed with CTRL Key Down	Keyword	Hex Character Transmitted
Synchronous idle	V	*SYN,*CTLV	X'16'
End of transmission block	W	*ETB,*CTLW	X'17'
Cancel previous word or character	X	*CAN,*CTLX	X'18'
End of medium	Y	*EM,*CTLY	X'19'
Substitute	Z	*SUB,*CTLZ	X'1A'
Escape	[*ESC	X'1B'
File separator	\	*FS	X'1C'
Group separator]	*GS	X'1D'
Record separator	~	*RS	X'1E'
Unit separator	?	*US	X'1F'
Delete		*DEL	X'7F'

Table 11 shows the keys on the auxiliary keypad that normally transmit the codes for the numerals, decimal point, minus sign, and comma.

Table 11. Numeric Keypads

Keyword	Hex Character Transmitted	Control Character Description
*NUM0	X'30' or X'1B3F70 ¹	Numeric keypad 0 key (VT52 mode)
	X'30' or X'1B4F70 ¹	Numeric keypad 0 key (VT100 or VT220 7-bit mode)
	X'30' or X'8F70 ²	Numeric keypad 0 key (VT220 8-bit mode)
*NUM1	X'31' or X'1B3F71 ¹	Numeric keypad 1 key (VT52 mode)
	X'31' or X'1B4F71 ¹	Numeric keypad 1 key (VT100 or VT220 7-bit mode)
	X'31' or X'8F71 ²	Numeric keypad 1 key (VT220 8-bit mode)
*NUM2	X'32' or X'1B3F72 ¹	Numeric keypad 2 key (VT52 mode)
	X'32' or X'1B4F72 ¹	Numeric keypad 2 key (VT100 or VT220 7-bit mode)
	X'32' or X'8F72 ²	Numeric keypad 2 key (VT220 8-bit mode)
*NUM3	X'33' or X'1B3F73 ¹	Numeric keypad 3 key (VT52 mode)
	X'33' or X'1B4F73 ¹	Numeric keypad 3 key (VT100 or VT220 7-bit mode)
	X'33' or X'8F73 ²	Numeric keypad 3 key (VT220 8-bit mode)
*NUM4	X'34' or X'1B3F74 ¹	Numeric keypad 4 key (VT52 mode)
	X'34' or X'1B4F74 ¹	Numeric keypad 4 key (VT100 or VT220 7-bit mode)
	X'34' or X'8F74 ²	Numeric keypad 4 key (VT220 8-bit mode)
*NUM5	X'35' or X'1B3F75 ¹	Numeric keypad 5 key (VT52 mode)
	X'35' or X'1B4F75 ¹	Numeric keypad 5 key (VT100 or VT220 7-bit mode)
	X'35' or X'8F75 ²	Numeric keypad 5 key (VT220 8-bit mode)
*NUM6	X'36' or X'1B3F76 ¹	Numeric keypad 6 key (VT52 mode)
	X'36' or X'1B4F76 ¹	Numeric keypad 6 key (VT100 or VT220 7-bit mode)
	X'36' or X'8F76 ²	Numeric keypad 6 key (VT220 8-bit mode)

Table 11. Numeric Keypads (continued)

Keyword	Hex Character Transmitted	Control Character Description
*NUM7	X'37' or X'1B3F77' ¹	Numeric keypad 7 key (VT52 mode)
	X'37' or X'1B4F77' ¹	Numeric keypad 7 key (VT100 or VT220 7-bit mode)
	X'37' or X'8F77' ²	Numeric keypad 7 key (VT220 8-bit mode)
*NUM8	X'38' or X'1B3F78' ¹	Numeric keypad 8 key (VT52 mode)
	X'38' or X'1B4F78' ¹	Numeric keypad 8 key (VT100 or VT220 7-bit mode)
	X'38' or X'8F78' ²	Numeric keypad 8 key (VT220 8-bit mode)
*NUM9	X'39' or X'1B3F79' ¹	Numeric keypad 9 key (VT52 mode)
	X'39' or X'1B4F79' ¹	Numeric keypad 9 key (VT100 or VT220 7-bit mode)
	X'39' or X'8F79' ²	Numeric keypad 9 key (VT220 8-bit mode)
*NUMMINUS	X'2D' or X'1B3F6D' ¹	Numeric keypad minus key (VT52 mode)
	X'2D' or X'1B4F6D' ¹	Numeric keypad minus key (VT100 or VT220 7-bit mode)
	X'2D' or X'8F6D' ²	Numeric keypad minus key (VT220 8-bit mode)
*NUMCOMMA	X'2C' or X'1B3F6C' ¹	Numeric keypad comma key (VT52 mode)
	X'2C' or X'1B4F6C' ¹	Numeric keypad comma key (VT100 or VT220 7-bit mode)
	X'2C' or X'8F6C' ²	Numeric keypad comma key (VT220 8-bit mode)
*NUMPERIOD	X'2E' or X'1B3F6E' ¹	Numeric keypad period key (VT52 mode)
	X'2E' or X'1B4F6E' ¹	Numeric keypad period key (VT100 or VT220 7-bit mode)
	X'2E' or X'8F6E' ²	Numeric keypad period key (VT220 8-bit mode)
*PF1	X'1B50'	Numeric keypad PF1 key (VT52 mode)
	X'1B4F50'	Numeric keypad PF1 key (VT100 or VT220 7-bit mode)
	X'8F50' ²	Numeric keypad PF1 key (VT220 8-bit mode)
*PF2	X'1B51'	Numeric keypad PF2 key (VT52 mode)
	X'1B4F51'	Numeric keypad PF2 key (VT100 or VT220 7-bit mode)
	X'8F51' ²	Numeric keypad PF2 key (VT220 8-bit mode)
*PF3	X'1B52'	Numeric keypad PF3 key (VT52 mode)
	X'1B4F52'	Numeric keypad PF3 key (VT100 or VT220 7-bit mode)
	X'8F52' ²	Numeric keypad PF3 key (VT220 8-bit mode)
*PF4	X'1B53'	Numeric keypad PF4 key (VT52 mode)
	X'1B4F53'	Numeric keypad PF4 key (VT100 or VT220 7-bit mode)
	X'8F53' ²	Numeric keypad PF4 key (VT220 8-bit mode)
Notes:		
1. A single-character is transmitted when in keypad numeric mode; a 3-character sequence is sent when in keypad application mode.		
2. This sequence is a shortened version of the 7-bit sequence. It is either presented when operating in 8-bit mode, which can be called by the remote VT220 host or server, or it may be specified in the ASCOPRMOD parameter of the STRTCPTLN CL command.		

Table 12 on page 177 shows the keys that transmit the codes for the function keys on the top row of the VT220 keyboard.

Table 12. Top Row Function Keys

Keyword	Hex Character Transmitted	Control Character Description
*F6	X'1B5B31377E'	Top row F6 function key (VT220 7-bit mode)
	X'9B31377E' ¹	Top row F6 function key (VT220 8-bit mode)
*F7	X'1B5B31387E'	Top row F7 function key (VT220 7-bit mode)
	X'9B31387E' ¹	Top row F7 function key (VT220 8-bit mode)
*F8	X'1B5B31397E'	Top row F8 function key (VT220 7-bit mode)
	X'9B31397E' ¹	Top row F8 function key (VT220 8-bit mode)
*F9	X'1B5B32307E'	Top row F9 function key (VT220 7-bit mode)
	X'9B32307E' ¹	Top row F9 function key (VT220 8-bit mode)
*F10	X'1B5B32317E'	Top row F10 function key (VT220 7-bit mode)
	X'9B32317E' ¹	Top row F10 function key (VT220 8-bit mode)
*F11	X'1B5B32337E'	Top row F11 function key (VT220 7-bit mode)
	X'9B32337E' ¹	Top row F11 function key (VT220 8-bit mode)
*F12	X'1B5B32347E'	Top row F12 function key (VT220 7-bit mode)
	X'9B32347E' ¹	Top row F12 function key (VT220 8-bit mode)
*F13	X'1B5B32357E'	Top row F13 function key (VT220 7-bit mode)
	X'9B32357E' ¹	Top row F13 function key (VT220 8-bit mode)
*F14	X'1B5B32367E'	Top row F14 function key (VT220 7-bit mode)
	X'9B32367E' ¹	Top row F14 function key (VT220 8-bit mode)
*F15 or *HELP	X'1B5B32387E'	Top row F15 function key (also HELP key) (VT220 7-bit mode)
	X'9B32387E' ¹	Top row F15 function key (also HELP key) (VT220 8-bit mode)
*F16 or *DO	X'1B5B32397E'	Top row F16 function key (also Do key) (VT220 7-bit mode)
	X'9B32397E' ¹	Top row F16 function key (also Do key) (VT220 8-bit mode)
*F17	X'1B5B33317E'	Top row F17 function key (VT220 7-bit mode)
	X'9B33317E' ¹	Top row F17 function key (VT220 8-bit mode)
*F18	X'1B5B33327E'	Top row F18 function key (VT220 7-bit mode)
	X'9B33327E' ¹	Top row F18 function key (VT220 8-bit mode)
*F19	X'1B5B33337E'	Top row F19 function key (VT220 7-bit mode)
	X'9B33337E' ¹	Top row F19 function key (VT220 8-bit mode)
*F20	X'1B5B33347E'	Top row F20 function key (VT220 7-bit mode)
	X'9B33347E' ¹	Top row F20 function key (VT220 8-bit mode)
Notes:		
1. This sequence is a shortened version of the 7-bit sequence. It is only presented when operating in 8-bit mode, which can be called by the remote VT220 host or server, or it may be specified in the ASCOPRMOD parameter of the STRTCPTLN CL command.		

Table 13 on page 178 shows the keys that transmit codes for the editing keypad keys.

Table 13. Editing Keypad

Keyword	Hex Character Transmitted	Control Character Description
*CSRUP	X'1B41'	Cursor-up key (VT52 mode)
	X'1B5B41'	Cursor-up key (VT100 or VT220 7-bit Cursor Key Mode Reset)
	X'9B41'	Cursor-up key (VT220 8-bit Cursor Key Mode Reset)
	X'1B4F41'	Cursor-up key (VT100 or VT220 7-bit Cursor Key Mode Set)
	X'8F41'	Cursor-up key (VT220 8-bit Cursor Key Mode Set)
*CSRDOWN	X'1B42'	Cursor-down key (VT52 mode)
	X'1B5B42'	Cursor-down key (VT100 or VT220 7-bit Cursor Key Mode Reset)
	X'9B42'	Cursor-down key (VT220 8-bit mode Cursor Key Mode Reset)
	X'1B4F42'	Cursor-down key (VT100 or VT220 7-bit Cursor Key Mode Set)
	X'8F42'	Cursor-down key (VT220 8-bit mode Cursor Key Mode Set)
*CSRRIGHT	X'1B43'	Cursor-right key (VT52 mode)
	X'1B5B43'	Cursor-right key (VT100 or VT220 7-bit Cursor Key Mode Reset)
	X'9B43'	Cursor-right key (VT220 8-bit Cursor Key Mode Reset)
	X'1B4F43'	Cursor-right key (VT100 or VT220 7-bit Cursor Key Mode Set)
	X'8F43'	Cursor-right Key (VT220 8-bit Cursor Key Mode Set)
*CSRLEFT	X'1B44'	Cursor-left key (VT52 mode)
	X'1B5B44'	Cursor-left key (VT100 or VT220 7-bit Cursor Key Mode Reset)
	X'9B44'	Cursor-left key (VT220 8-bit Cursor Key Mode Reset)
	X'1B4F44'	Cursor-left key (VT100 or VT220 7-bit Cursor Key Mode Set)
	X'8F44'	Cursor-left key (VT220 8-bit Cursor Key Mode Set)
*FINDKEY	X'1B5B317E'	Editing keypad Find key (VT220 7-bit mode)
	X'9B317E' ¹	Editing keypad Find key (VT220 8-bit mode)
*INSERTKEY	X'1B5B327E'	Editing keypad Insert Here key (VT220 7-bit mode)
	X'9B327E' ¹	Editing keypad Insert Here key (VT220 8-bit mode)
*REMOVEKEY	X'1B5B337E'	Editing keypad Remove key (VT220 7-bit mode)
	X'9B337E' ¹	Editing keypad Remove key (VT220 8-bit mode)
*SELECTKEY	X'1B5B347E'	Editing keypad Select key (VT220 7-bit mode)
	X'9B347E' ¹	Editing keypad Select key (VT220 8-bit mode)
*PREVSCN	X'1B5B357E'	Editing keypad Prev Screen key (VT220 7-bit mode)
	X'9B357E' ¹	Editing keypad Prev Screen key (VT220 8-bit mode)
*NEXTSCN	X'1B5B367E'	Editing keypad Next Screen key (VT220 7-bit mode)
	X'9B367E' ¹	Editing keypad Next Screen key (VT220 8-bit mode)

Table 13. Editing Keypad (continued)

Keyword	Hex Character Transmitted	Control Character Description
Notes:		
1. This sequence is a shortened version of the 7-bit sequence. It is only presented when operating in 8-bit mode, which can be called by the remote VT220 host or server, or it may be specified in the ASCOPRMOD parameter of the STRTCPTELN CL command.		

Table 14 shows the keywords that are handled locally within the AS/400 Telnet client session.

Table 14. Local AS/400 Function Keys

Keyword	Hex Character Transmitted	Control Character Description
*SENDWOCR	None ¹	VT100 and VT220 control key
*SHIFTDSP	None ²	Shift display
*HIDE	None ³	Hide input
*KEYPRI	None ⁴	Call primary keyboard mapping of 5250 function keys
*KEYALT	None ⁴	Call alternate keyboard mapping of 5250 function keys
Notes:		
1. The data that has been typed is sent to the remote system without appending the carriage return and line feed characters.		
2. The *SHIFTDSP keyword is used when the remote system has selected 132-column mode and the 5250 terminal only has 80 columns. When the function key that has the *SHIFTDSP function assigned to it is pressed, the rightmost 80 columns or the leftmost 80 columns are shown depending on what is currently on the display.		
3. The *HIDE keyword is used when the user does not want typed characters shown on the display, for example, when typing a password.		
4. The *KEYPRI and *KEYALT keywords signal the TELNET client session to dynamically call the respective full-screen ASCII keyboard map. The *KEYPRI mapping calls the primary keyboard map, and *KEYALT calls the alternate keyboard map. By default, they are assigned to the Page Down (Roll Up) and the Page Up (Roll Down) 5250 function keys. When the requested keyboard map is already in effect, no action is taken.		

VTxxx—National Language Support

There are alternative methods of selecting character mapping between the client and server systems with VTxxx emulation. These are:

- Coded character set identifier (CCSID)
- Multinational mode
- National mode

If none of these modes is suitable, you may set up and specify your own user-defined mapping tables.

Note: VTxxx support is limited to a subset of single-byte character set (SBCS) languages. A list of the supported languages is found later in this section. Any of these supported single-byte language translation tables can be modified to map any single-byte language that is preferred, then identified in the appropriate parameter for starting Client Telnet.

Mode selection is done with the CCSID parameter of the Start TCP/IP Telnet (STRTCPTELN) command. The incoming ASCII/EBCDIC table (TBLVTIN) and outgoing EBCDIC/ASCII table (TBLVTOUT) parameters of this command allow the

specification of user-defined mapping tables. If these are not required, the default value of *CCSID allows for character mapping by using the mode specified in the CCSID parameter.

VTxxx—Multinational Mode

The **multinational mode** supports the DEC multinational character set, which is an 8-bit character set that contains most characters used in the major European languages. The ASCII character set is included in the DEC multinational character set. The DEC multinational character set is used by default.

VTxxx—National Mode

The **national mode** supports the national replacement character set, which is a group of 7-bit character sets. Only one national character set is available for use at any one time. VT220 also supports the standard 7-bit ASCII character set as part of the national mode. The VT220 terminal supports the following 7-bit ASCII national language character sets:

- British
- Dutch
- Finnish
- French
- French/Canadian
- German
- Italian
- Norwegian Danish
- Spanish
- Swedish
- Swiss
- US English

To use a national mode, mapping tables are required to map incoming ASCII data into EBCDIC and outgoing EBCDIC data into ASCII when operating in VTxxx full-screen mode.

A national mode (NLS mapping table) may be selected with the CCSID parameter on the Telnet command (see “VTxxx—Start TCP/IP Telnet Command” on page 166).

A numeric value representing a registered CCSID value in the range 1-65553 may be entered to identify the appropriate mapping table. Details of registered CCSIDs are found in the *International Application Development* book.

The NLS mapping tables are built dynamically to a remote system the first time Telnet is used, and are based on DEC national replacement character sets. Because the character sets are based on 7 bits, they can contain only the unique characters from one country. Because the DEC multinational character set is based on 8 bits, it has sufficient bits to allow the unique characters from a group of countries to be included.

Identifying Table Objects

You can identify the table objects (*TBL) using the Work with Object command:

```
WRKOBJ OBJ(QUSRSYS/Q*) OBJTYPE(*TBL)
```

All of the system table objects are in QUSRSYS library.

The table objects are named Qxxxxyyyzzz where xxx is the FROM code page, yyy is the TO character set and zzz is the TO code page.

For the outgoing (EBCDIC-to-ASCII) table:

- The FROM code page ID is taken from the code page ID in QCHRID of message description CPX8416 (use WRKMSGD CPX8416 to display), 037 in Figure 118 from a US English based system.
- The TO character set and code page are derived from the CCSID parameter used with the Telnet command. See Table 15 for the IDs used.

For the incoming (ASCII-to-EBCDIC) table:

- The FROM code page ID is derived from the CCSID parameter used with the Telnet command. See Table 15 for the IDs used.
- The TO character set and code page are taken from the character set ID and code page ID in QCHRID of message description CPX8416 (use WRKMSGD CPX8416 to display), 697 and 037 in Figure 118 from a US English based system.

```

System:  SYSNAM01
Message ID . . . . . : CPX8416
Message file . . . . . : QCPFMSG
Library . . . . . : QSYS

Message . . . . . :
QCHRID 697 37          QCURSYM $ QDATFMT MDY QDATSEP /
QDECFMT QLEAPADJ 0 QCCSID 37 QTIMSEP : QLANGID  ENU
QCNTYID US QIGCCDEFNT *NONE

```

Figure 118. Example CPX8416 Message

Table 15. ASCII/EBCDIC Translation Table Naming

CCSID	Character Set		Code Page	
	Actual ID	Table ID	Actual ID	Table ID
MULTINAT	1290	A05	1100	A5U
BRITISH	1291	A06	1101	A5V
1292	A07	1102	A5W	
1293	A08	1103	A5X	
289	289	1104	A5Y	
1192	A8E	1020	A3M	
265	265	1011	A3D	
293	293	1012	A3E	
1297	BAB	1107	A52	
1195	A8H	1023	A3P	
1296	BAA	1106	A51	
1193	A8F	1021	A3N	

For example, on a British system with a QCHRID of 697 285 (character set 697 code page 285) in message CPX8416 that uses Telnet with CCSID(*BRITISH), the tables would have the following names:

- Outgoing (EBCDIC-to-ASCII) Q285A06A5V
- Incoming (ASCII-to-EBCDIC) QA5V697285

User-Defined Mapping Tables (ASCII Mode)

Where the multinational or NLS mapping tables do not meet the requirements of a user, user-defined character mapping tables can be created and used.

You also have the ability to specify user-defined mapping tables using the outgoing ASCII-to-EBCDIC table (TBLVTOUT) and incoming ASCII-to-EBCDIC table (TBLVTIN) parameters of the STRTCPTLN command. You can specify a user-defined mapping table for either the outgoing mapping table or the incoming mapping table and then use the system default value for the other.

For details on how to create user-defined mapping tables, see Appendix C. Mapping Tables Associated with TCP/IP Function.

System Functions Available during a Telnet Client Session

This material is covered in the **AS/400e Information Center** under the **TCP/IP** topic. For more information see "TCP/IP Topics in the Information Center" on page xv.

Print

Press the 5250-mode Print key to start the printing function and produce a printed copy of the display. Press the Reset key to return to the Telnet session. The Print key is always processed on the client system, so the spooled file created by the Print key is placed on the job output queue on the client system. Refer to the *System Operation* book for more information about working with and printing spooled files.

Chapter 6. Telnet Server

Important note: A thorough and in-depth explanation of Telnet is beyond the scope and purpose of this document. The majority of material on the Telnet server is covered in the **AS/400e Information Center** under the **TCP/IP** topic. For more information see "TCP/IP Topics in the Information Center" on page xv.

The topics that remain in this chapter include conceptual and reference information on the Telnet server 3270 and VTxxx full-screen modes, and ASCII line and Printer pass-through modes. Also in this chapter, you will find the following topics:

- Telnet scenarios for establishing cascaded sessions
- workstation type negotiations and mappings
- system API enhancement, including a discussion on dynamic application printing with TCP/IP

Setting Up the Telnet Server

This material is covered in the **AS/400e Information Center** under the **TCP/IP** topic. For more information see "TCP/IP Topics in the Information Center" on page xv.

Determining Which Emulation Is Negotiated

To determine which type of emulation an autoselected Telnet client session negotiates, the type of virtual device that is created should be examined by the `WRKDEVD QPADEV*` command. For both VT220 and VT100, the virtual device type that is created is V100. For virtual printer sessions, the device type that is created is 3812 for single byte and 5553 for double byte.

5250 Full-Screen Mode

This material is covered in the **AS/400e Information Center** under the **TCP/IP** topic. For more information see "TCP/IP Topics in the Information Center" on page xv.

Examples of 5250 Server to 5250 Full-Screen Telnet Client

This topic describes some practical experiences of a 5250 full-screen Telnet server with different 5250 clients.

OS/2

Apple Macintosh

OS/2 5250 Full-Screen Telnet Client

IBM TCP/IP Version 2.0 for OS/2 provides a 5250 full-screen Telnet client (TN5250). TN5250 can be started either by clicking on the TN5250 icon or from an OS/2 command line. For example:

```
[C:\]tn5250 sysnam123
```

The session is ended by selecting *Exit* from the menu bar.

Keyboard Mapping: The default keyboard map can be changed by creating a file named TN5250.KEY by using a text editor. For example, the following would map the PS/2 Enter key (Enter key on numeric keypad) to the AS/400 Enter key function:

```
enter enter
```

TN5250.KEY is searched for (and used if found) when TN5250 is called. OS/2 looks for this file first in the current directory and then in the ETC directory. There is a list of valid PS/2 keys and AS/400 functions plus a listing of the default keyboard map in the *IBM TCP/IP Version 2.0 for OS/2 User's Guide*.

Character Mapping: The PS/2 is an ASCII-based device. 5250 Telnet data streams are in EBCDIC format. The PS/2 must, therefore, translate all incoming data from EBCDIC to ASCII and all outgoing data from ASCII to EBCDIC. A default mapping table is provided to do this. User-defined mapping tables can also be created. A sample table (5250XLT.SAM) is provided in the ETC directory. The table includes both ASCII-to-EBCDIC and EBCDIC-to-ASCII translation. A user-defined mapping table is selected using the `-tx` option when starting a session, for example:

```
[C:\]tn5250 sysnam123 -tx 5250x1t.sam
```

Apple Macintosh 5250 Full-Screen Telnet Client

Apple SNA•ps 5250 provides 5250 connectivity for Apple Macintosh computers. Version 1.2 adds support for AppleTalk and TCP/IP. The TCP/IP support is used in conjunction with TCP/IP Connection for Macintosh (M8113Z). This support provides a TN5250 client that allows Apple Macintosh computers to connect directly (the gateway support is SNA/APPC only) to an AS/400. Token-ring and Ethernet are supported. LocalTalk/Ethernet gateways are available from third parties.

Once installed, the Macintosh TCP/IP support is configured using three displays in two steps:

1. The first two displays are selected with the MacTCP control panel icon. Having selected the type of network adapter to be used (token-ring or Ethernet) from the first display, click on *more* for the second display. The following is configured with the second display:
 - The gateway internet address
 - Your domain name
 - The name server internet address
 - Your internet address

The internet address of the gateway in our network example was 9.4.73.193. This was entered in the *Gateway Address* field under *Routing Information*. Our domain name of RCHLAND.IBM.COM was entered in the *Domain* field under *Domain Name Server Information*. The name server internet address (9.4.191.76 in our network example) was entered in the *IP Address* field, also under *Domain Name Server Information*.

The *IP Address* section is used to enter your own internet address. Enter your network class (A in our network example). The slider will now move to show the bits allocated to the network ID (Net), 8-bits in our class A network example (see Figure 119 on page 185). The slider bar must now be moved to correctly divide the remaining section into Subnetwork ID (Subnet) and Host ID (Node). The subnet mask in our network example was 255.255.255.192. As you move the slider bar, the subnet mask ID is dynamically updated. You should now move the bar until the subnet mask ID matches yours. At this point, the number

of bits in the subnet and in the node in your network are shown. In the example here, this equates to 18-bits allocated to the Subnet and 6-bits allocated to the Node. For information on subaddressing, see "Subnetworks and Subnet Masks" on page 6

Now enter your local internet address by entering information in the network ID (Net), Subnetwork ID (Subnet), and Host ID (Node) fields. The information is entered in these fields in **decimal format**. In our example the local IP address in the more normal dotted decimal format was 9.4.73.214. This information was entered as follows: 9 in the *Net* field, 5399 in the *Subnet* field and 22 in the *Node* field. The example below shows how these numbers are derived from the dotted decimal IP address once the internet address has been divided into Net, Subnet and Node:

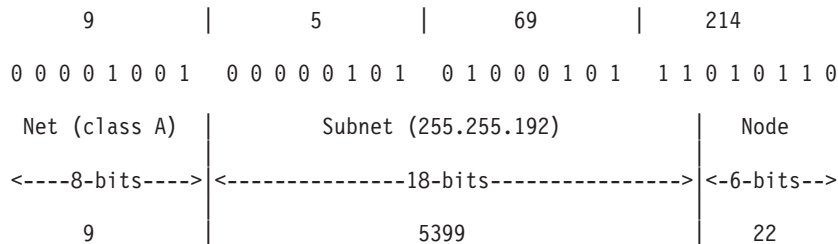


Figure 119. Class A Network Example

Note:

The arithmetic involved in converting a dotted decimal address to the required decimals for the subnet and node can be tedious. If you are unsure of the value to put here, enter what you think it should be and click on OK. An IP address will be displayed (corresponding to the numbers you have entered) in the more normal dotted decimal format on the previous panel. By now correcting the IP Address on this panel, and selecting *more* again (do not press the Enter key after correcting the IP address), you will see that the Macintosh has calculated the correct Subnet and Node values on the next panel.

Once you are satisfied with this, click on OK and you are returned to the first panel where your IP address is shown in dotted decimal format.

Having completed the above, close the MacTCP control panel and start the Macintosh again.

2. The Macintosh host table information is configured with the third panel. Select the SNA•ps TCP/5250 option from the Apple Menu control panels option. Add host names and addresses as required. Click on ADD to add a name entered to the table.

To start a TN5250 session:

1. Select TCP/IP with SNA•ps 5250 from the SNA•ps folder.
2. Select *Session* from the menu bar
3. Select *Connection* from the menu bar
4. Select a *Connection Type* of 5250 Access/TCP.
5. Select the required host name from the list of configured *Hosts*.

6. Click on *tn5250* on the right side of the screen.
7. Click on *Connect*.
8. You should then be presented with an AS/400 sign-on display.

When the AS/400 sign-on display appears, it is inside a Macintosh Window. Along the top of the window is a menu bar, where you see various options. Most of these need not concern us. However, you should take note of the following:

Note: To end the session, select *Session* and then *Disconnect* from the menu bar.

Keyboard Mapping: A function key can be selected either with the mouse or with the keyboard. To select a function key with the mouse, use the *Keypad* option from the menu bar. A default keypad is provided (Standard Keypad). Others can be created as required that might include Autokeys for example. To display the current keyboard map, select *Preferences* and then *Keyboard Map* from the menu bar. The keyboard map can also be changed from the panel presented using a 'drag and drop' process.

Character Mapping: The Apple Macintosh is an ASCII-based device. Because the TN5250 data stream is EBCDIC-based, the Macintosh must do ASCII-to-EBCDIC character translation. The translation table used is changed by selecting an appropriate host language. To do this, select *Session* and then *Host Language* from the menu bar. A list of the languages that are supported is presented from which a language can be selected.

For more detailed information about configuring an Apple Macintosh to connect to the AS/400 with both TCP/IP and SNA, see *Using Apple Macintosh with the AS/400*, GG24-4071.

3270 Full-Screen Mode

3270 full-screen support allows Telnet client users to sign on and run AS/400 5250 full-screen applications even though 3270 full-screen support is negotiated. 3270 full-screen support is negotiated with any Telnet client application that supports 3270 full-screen applications rather than 5250 full-screen applications. An example of a system that negotiates 3270 full-screen support is the System/390 family.

TN5250 delivers the data stream between the two systems as EBCDIC. Because the 3270 data streams are translated into 5250 data streams, the workstation devices operate as a remote 5251 display to the AS/400 system and application programs.

Figure 120 on page 187 shows a network configuration using the AS/400 3270 Telnet server support.

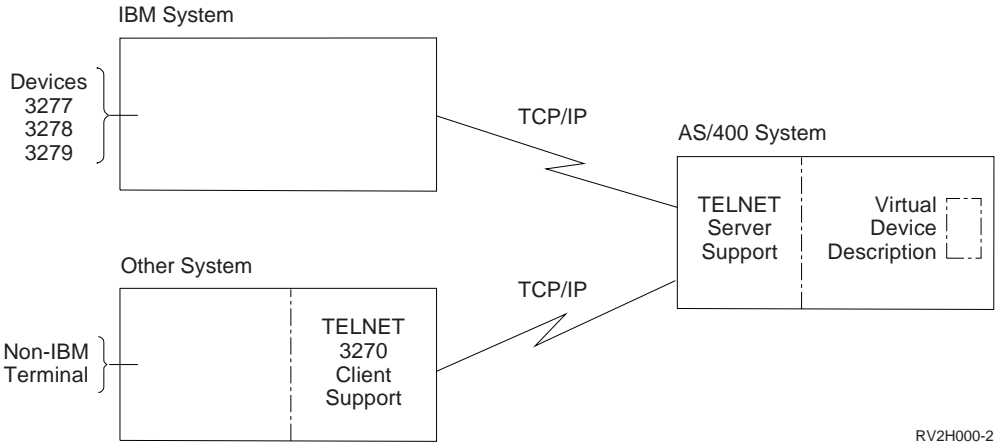


Figure 120. Configuration Example of 3270 Telnet Server Support

Setting up for 3270 Full-Screen Mode

You can use the CFGTCPTELN command to set up your 3270 full-screen mode session.

```

                                Configure TCP/IP TELNET
                                System:  SYSNAM01

Select one of the following:

    1. Change TELNET attributes

    6. Display 3270 keyboard map
    7. Change 3270 keyboard map
    8. Set 3270 keyboard map

Work with associated system values:
    10. Autoconfigure virtual devices
    11. Limit security officer device access
    12. Inactive job time-out
    13. Inactive job message queue
    14. Limit device sessions
    15. Action to take for failed sign-on attempts
    16. Maximum sign-on attempts allowed

Selection or command
===>
F3=Exit  F4=Prompt  F9=Retrieve  F12=Cancel
More...

```

Figure 121. CFGTCPTELN in 3270 Full-Screen Session

Step 1—3270—Starting the Telnet Server Job

The server job for a TCP/IP application must be started in the QSYSWRK subsystem. The Start TCP/IP Server (STRTCPSVR) command starts the servers that are shipped with the TCP/IP Utilities licensed program.

Even though the Change Telnet Attributes (CHGTELNA) command has an AUTOSTART parameter, that parameter is overridden or ignored by the STRTCPSVR command.

Step 2—3270—Setting the Number of Virtual Devices

The server system uses virtual devices to direct output to devices on your system. AS/400 Telnet server support automatically selects (and creates, if necessary) these devices for you. You may also choose to create your own virtual device under the QVIRCDnnnn virtual controller. Note that virtual devices that you create under QVIRCDnnnn will not be auto-selected. This would require a user exit program or client subnegotiation to select this device.

The option is available for you to allow the Telnet server support on the AS/400 system to automatically configure virtual controllers and devices. The QAUTOVRT system value specifies the maximum number of devices that are automatically configured by the system. Use the Change System Value (CHGSYSVAL) command to change the value of the QAUTOVRT system value. For example, entering the following command string changes the number of virtual devices that can be allocated on a system to 50:

```
CHGSYSVAL SYSVAL(QAUTOVRT) VALUE(50)
```

Note: QAUTOVRT has been modified for Version 4 Release 2 to support numeric values of 0 through 32500, and a special value of *NOMAX.

To determine and set the maximum number of users you want signed on to the AS/400 system at any time, do the following:

1. Set the QAUTOVRT value to 32500, the maximum value allowed, or use the *NOMAX value.
2. Let your users use pass-through, Telnet, the virtual terminal application program interface and Telnet Printer pass-through until you decide that the number of virtual devices created is sufficient for normal system operation.
3. Change the QAUTOVRT value from 32500 to the number of virtual devices you require for normal system operation.

If you have never allowed automatic configuration of virtual devices on your system, the QAUTOVRT value is 0. A Telnet connection attempt with a dependence on automatic creation of the virtual device then fails because the Telnet server does not create more than the specified QAUTOVRT devices (zero). If you try to connect, you receive a message (TCP2504) indicating that the Telnet client session has ended and the connection is closed. In addition, the QTGTELNETS job in the QSYSWRK subsystem on the AS/400 Telnet server sends a message (CPF8940) indicating that a virtual device cannot be automatically selected.

If you change the QAUTOVRT value to 10, the next Telnet connection attempt causes the Telnet server to create a virtual device. This virtual device is created because the number of virtual devices on the controller (0) is less than the number specified in the QAUTOVRT (10). Even if you change the specified number to 0 again, the next user attempting a Telnet connection succeeds. When a Telnet connection attempt fails, the CPF87D7 message is sent to the system operator message queue on the Telnet server system. The CPF87D7 message indicates that the AS/400 server is not able to create a virtual device.

The Telnet server uses the following conventions for naming virtual controllers and devices:

- Virtual controllers are named QPACTLnn.

- Virtual device descriptions can be a name selected by the user. If a valid device name is communicated to the Telnet server either via user exit or a Telnet client, a device by that name will be created, if necessary, under virtual controller QVIRCDnnnn.

The virtual controller descriptions (QPACTLnn) have the 5250 data stream optimization switch (OPTDTASTR) set to *YES by default. There is no reason to change this for use by 3270 Telnet.

Security Considerations for 3270 Full-Screen Mode: The number of sign-on attempts allowed increases if virtual devices are automatically configured. The number of sign-on attempts is equal to the number of system sign-on attempts allowed multiplied by the number of virtual devices that can be created. The number of system sign-on attempts allowed is defined by the QMAXSIGN system value. The number of virtual devices that can be created is defined by the QAUTOVRT system value.

In Version 4 Release 2, the following level of support has been added with regard to security of virtual devices:

- With a user-supplied exit program, you can audit the number of sign-on attempts
- You have the ability to deny connections
- You have the ability to allow bypassing of the sign-on screen

For more information on Telnet exit points and how to use them, see “TELNET Exit Points” on page 541 in Appendix E. TCP/IP Application Exit Points and Programs.

Telnet and SNA 5250 Pass-Through Considerations for 3270 Full-Screen Mode: The AS/400 system supports 5250 pass-through. 5250 pass-through is similar to Telnet but runs on an SNA (Systems Network Architecture) protocol network rather than a TCP/IP network. 5250 pass-through uses virtual displays to direct output to the physical devices just as Telnet does. In 5250 pass-through, the AS/400 system automatically creates virtual devices in the same way that it does for Telnet. Therefore, the QAUTOVRT system value controls the number of automatically configured virtual devices for both 5250 pass-through and Telnet. For more information about 5250 pass-through, see the *Remote Work Station Support* book.

Step 3—3270—Setting the QLMTSECOFR Value

The OS/400 licensed program supports the limit security officer (QLMTSECOFR) system value, which limits the devices the security officer can sign on to. If the QLMTSECOFR value is greater than zero, the security officer must be authorized to use the virtual device descriptions. However, when this value is 0, the system does not limit the devices users with *ALLOBJ or *SERVICE special authority can sign on to.

On AS/400 systems with a QSECURITY value of 30 or greater, a user with security officer authority (*ALLOBJ) must be authorized to use devices before the system allows the user to use those devices. For example, each display device that a security officer wants to sign on to (local, remote, or virtual), must have had the following authority specified with the Grant Object Authority (GRTOBJAUT) command:

```
GRTOBJAUT OBJ(display_name) OBJTYPE(*DEVD)
          AUT(*CHANGE) USER(QSECOFR)
```

This procedure is very important because Telnet automatically configures virtual devices. If the QLMTSECOFR value is set to 0, all devices automatically configured by Telnet can be used by the security officer. If you set the QLMTSECOFR value to 1, your security officer is not able to use the virtual devices created by Telnet unless you grant object authority to the security officer for that virtual device. The automatic configuration support can delete and re-create the virtual device. If this occurs, authority must be granted to the security officer each time the virtual device is created.

Step 4—3270—Working with Associated System Values

In addition to the QAUTOVRT and QLMTSECOFR, the following system values are available for you to work with from the Configure TCP/IP Telnet (CFGTCPTELN) menu:

- QINACTIVT: Inactive job time-out
- QINACTMSGQ: Inactive job message queue
- QLMTDEVSSN: Limit device sessions
- QMAXSGNACN: Action to take for failed sign-on attempts
- QMAXSIGN: Maximum sign-on attempts allowed
- QRMTSIGN: Remote sign-on control
- QDEVRCYACN: Device I/O error action
- QDSCJOBITV: Time interval before disconnected jobs end

Figure 122 on page 192 shows the Configure TCP/IP Telnet (CFGTCPTELN) menu.

Setting the Telnet Timemark Timeout Value: You should also take into consideration the TIMMRKTIMO parameter.

The Telnet timemark timeout (TIMMRKTIMO) parameter specifies the number of seconds between TIMEMARK commands sent by the Telnet server. If Telnet is unable to send the TIMEMARK command, it closes the connection.

Step 5—3270—Creating Virtual Controllers and Devices

You can create virtual controllers and devices. If you create your own virtual devices, by allowing the system to automatically select the device name, you must be aware of the following:

- The virtual controller must be named QPACTLnn, where nn is a decimal number 01 or greater.
- The virtual device should be named QPADEVxxxx, where xxxx is an alphanumeric character from 0001 to ZZZZ.

Note: Starting with Version 4 Release 2, the xxxx are no longer only numeric characters, but also alphanumeric characters from 0001 to ZZZZ, allowing a maximum of 1,679,615 unique names (devices).

If you want to use more than 32500 devices, which is the maximum value for the QAUTOVRT system, you can set the QAUTOVRT system value to *NOMAX to allow additional devices to be created.

- The Telnet server reuses available existing virtual devices that were auto-created by selecting virtual devices of the same device type and model. When there are no more device type and model matches, but there are still available virtual devices, then the device type and model will be changed to match the client

device and model negotiated. This is true only for auto-created (QPADEVnnn) virtual devices. Typically, the auto-created virtual device will use the AS/400 system values for keyboard type, character set, and code page. Optionally, these display device attributes may be more specifically defined through the exit program or device specified client subnegotiation. Devices can also be selected via the exit program interface as opposed to being negotiated.

Step 6—3270—Defining Workstations to Subsystems

When you use Telnet to sign-on to an AS/400 server, the sign-on screen may not be displayed on your workstation. Before a user can sign on to the AS/400 server, the workstation must be defined to the subsystem. If the workstation has not been defined to the subsystem, you need to add a workstation entry to the subsystem description under which you want your job to run on the AS/400 server. The workstation in this case is the virtual display device automatically created by the Telnet server (QPADEVxxxx). The workstation name or the workstation type must be specified in the subsystem description on the AS/400 server. Use the Display Subsystem Description (DSPSBSD) command to see the workstation entries defined to a subsystem. (This only applies to display devices. Printer devices typically run in the QSPL subsystem.) The following command can be used to add all workstation types to a subsystem named QINTER:

```
ADDWSE SBSB(QINTER) WRKSTNTYPE(*ALL)
```

Note: The Add Work Station Entry (ADDWSE) command can be done when the subsystem is active. However, the changes may or may not take effect immediately. You may need to end and restart the subsystem.

Step 7—3270—Activating the QSYSWRK Subsystem

The QSYSWRK subsystem must be active. Use the Work with Subsystem (WRKSBS) command to display the status of the subsystem.

The Telnet server must also be started. The interactive subsystem, QINTER, which is used in previous examples in this chapter, needs to be started to run interactive jobs for Telnet sessions. The spooling subsystem (QSPL) needs to be active to run printer pass-through sessions.

Step 8—3270—Creating User Profiles for Telnet Users

At the server system, create one or more user profiles for Telnet users from other systems. The default user profile is *SYS. The following example shows a sample user profile:

```
CRTUSRPRF  USRPRF(CLERK1)
           PASSWORD(unique-password)
           JOB(CLERKLIB/CLERK1)
           TEXT('User profile Clerks Group 1')
```

Step 9—3270—Checking the QKBDTYPE System Value

When the AS/400 Telnet server automatically creates virtual display devices, it uses the QKBDTYPE system value to determine the keyboard type for the virtual device.

If the initial creation of the virtual device fails using the QKBDTYPE system value, the Telnet server attempts to create the device again, using a keyboard type value of USB. If the second attempt to create the virtual display device fails using the value of USB, then a message (CPF87D7) indicating that the virtual device cannot be automatically selected is sent to the system operator message queue.

Step 10—3270—Setting the Default Keyboard Mapping

A 3270 display station connected to an AS/400 system using Telnet appears to be a 5251 display station to an AS/400 system. The 3270 display station keyboard has a 5251-equivalent keyboard map associated with it which allows it to complete 5251-equivalent functions on the AS/400 system.

When a Telnet client system user first signs on in 3270 full-screen mode, the AS/400 system automatically assigns the *default* keyboard map to the user's 3277, 3278, or 3279 keyboard (unless a user-defined keyboard map has been set up to be automatically included in the user's profile sign-on procedure). This supplies the mapping needed for the 3270 keyboards to do most of the same functions as their 5250-equivalent keyboards do.

Note: If you are satisfied with the default keyboard mapping supplied when you sign on, you may skip the remainder of this topic and go to the next topic, "Break Messages in 3270 Full-Screen Mode" on page 195.

Configure TCP/IP TELNET

System: SYSNAM11

Select one of the following:

- 1. Change TELNET attributes
- 6. Display 3270 keyboard map
- 7. Change 3270 keyboard map
- 8. Set 3270 keyboard map

Work with associated system values:

- 10. Autoconfigure virtual devices
- 11. Limit security officer device access
- 12. Inactive job time-out
- 13. Inactive job message queue
- 14. Limit device sessions
- 15. Action to take for failed sign-on attempts
- 16. Maximum sign-on attempts allowed

More...

Selection or command
===>

F3=Exit F4=Prompt F9=Retrieve F12=Cancel

Figure 122. Configure TCP/IP TELNET Menu—TN3270

Displaying a Keyboard Map: Table 16 shows the default PF key assignments to perform the various 5250 functions. You can use the Display Keyboard Map (DSPKBDMAP) command to see the current keyboard mapping or use option 6 (Display 3270 keyboard map) on the Configure TCP/IP Telnet menu, while your terminal is in 3270 emulation mode.

Table 16. Default Keyboard Mapping

5250 Key Function	Default 3270 Keys to Select Function
Help	PF1
3270 Help	PF2
Clear	PF3
Print	PF4
Display Embedded Attributes	PF5
Test Request	PF6

Table 16. Default Keyboard Mapping (continued)

5250 Key Function	Default 3270 Keys to Select Function
Roll Down	PF7
Roll Up	PF8
Error Reset	PF10 (System/38), PF10 or Enter (System/36 and AS/400 system)
Sys Req	PF11
Record Backspace	PF12
F1 through F12	Press PA1, then one of the following: PF1 through PF12 ¹
F13 through F24	Press PA2, then one of the following: PF1 through PF12, or PF13 through F24 (if present)
Field Exit	Erase EOF, then Field Tab
Attention	For 3277 use Test Request, then PA1. For 3278/3279 use Attn key

Notes:

1. For example, to start F3, press PA1, wait for the system to respond, and then press PF3.

Changing a Keyboard Map: If you want to make either minor changes to the default keyboard map or to set a new keyboard map, use the Change Keyboard Map (CHGKBDMAP) or the Set Keyboard Map (SETKBDMAP) command. These commands are available from the Configure TCP/IP Telnet menu as option 7 (Change 3270 keyboard map) and option 8 (Set 3270 keyboard map), while your terminal is in 3270 emulation mode. The key assignments you specify are in effect until you use these commands again to specify new key assignments or until you sign off.

Note

The difference between CHGKBDMAP and SETKBDMAP is that with SETKBDMAP the system defaults are taken and then the changes in the SETKBDMAP are applied. With CHGKBDMAP, the system defaults plus any changes you have previously made during this session are taken and then the changes in the CHGKBDMAP are applied.

The following example of a CL program sets the keyboard mapping for a 327x-type terminal that is using Telnet to go to an AS/400 system. This program maps the AS/400 function keys to their equivalent function keys on the 327x terminal. The CPF8701 message is received if you attempt a CHGKBDMAP command from a terminal not in 3270 emulation mode. By monitoring for it, the rest of the program is not used in these circumstances.

```
PGM
MONMSG      MSGID(CPF8701 CPF0000)
CHGKBDMAP  PF1(*F1) PF2(*F2) PF3(*F3) PF4(*F4) PF5(*F5)
           PF6(*F6) PF7(*DOWN) PF8(*UP) PF9(*F9)
           PF10(*F10) PF11(*F11) PF12(*F12)
           PA1PF1(*HELP) PA1PF2(*HLP3270)
           PA1PF3(*CLEAR) PA1PF4(*PRINT)
           PA1PF5(*DSPATR) PA1PF6(*TEST) PA1PF7(*F7)
           PA1PF8(*F8) PA1PF9(*ATTN) PA1PF10(*RESET)
           PA1PF11(*SYSREQ) PA1PF12(*BCKSPC)

ENDPGM
```

By storing this CL source as part of the QCLSRC file in library TCPLIB as member CHGKBD, you can create the CL program CHGKBD into the TCPLIB library by using the following CL command:

```
CRTCLPGM PGM(TCPLIB/CHGKBD) SRCFILE(TCPLIB/QCLSRC)
        TEXT('Change the keyboard mapping for 327x terminals')
```

The CHGKBD program can then be called by anyone using Telnet to an AS/400 system. It can also be called automatically at sign-on time by specifying the CHGKBD program for the Initial program parameter on the CHGUSRPRF command or the CHGKBD program can be called by the profile's initial program.

PA1 and PA2 Keys on a PC Keyboard: The PA1 and PA2 keys do not appear on a PC keyboard. The function of these 3270 keys on a PC keyboard is provided by a keyboard mapping in your 3270 emulator.

These keys are used by the default 3270 Telnet keyboard mapping, so it is important that you know where these keys are on the keyboard before starting a 3270 Telnet session. This is especially important if you are planning to start a session without changing the keyboard mapping. You should refer to your emulator documentation for the keys or keystrokes required to provide these functions.

There are some 5250 key sequences for which there is no supported 3270 key sequence and, therefore, it is not possible to set using the keyboard mapping commands. These key sequences are:

- Field Plus
- Field Minus
- Erase all input fields

The 5250 Field Exit Key function is performed on a 3270 keyboard using the Erase EOF key and then the tab key.

Note: When using Telnet 3270 full-screen mode from the 3270 terminal and before the default mapping for the terminal is changed, the keys PF1 to PF12 might be emulated by the key sequence PA1 PFx. Prior to creating a new keyboard map, as in the previous example, instructions like Press PF3 or Press PF4 should read: Press PA1 PF3 and Press PA1 PF4. In this case, depending on the installation of the Telnet client for the host (VM Telnet client for example), when pressing PA1 the user might get the instruction TELNET command: at the bottom line of the display. If this instruction is displayed, then type: PA1, press the Enter key, move the cursor to the command line and press the desired PF key. In this case PF1 to PF12 might be emulated by:

1. Press PA1, get the Telnet instruction TELNET command:
2. Type PA1, press the Enter key.
3. Move the cursor to the command line.
4. Press the desired PF key.

For additional keyboard mapping information, see Appendix D. TELNET 3270 Keyboard Mappings.

Note: The **Host Command Facility (HCF)** is a feature available on System/370, 43xx, and 30xx host systems that enables a user on the host system to use applications on an AS/400 system or other systems as if they were using remotely attached 5250-type display stations. If you use HCF to connect to an AS/400 system and then use Telnet to sign on to another AS/400 system from that AS/400 system, you are in a 3270 full-screen mode session. The keyboard is mapped twice, once for the initial HCF session and once for the Telnet session. To use your PF keys the way you normally would, you must

change the keyboard mapping on both AS/400 systems, making sure that you use the same keyboard mapping on each AS/400 system.

Break Messages in 3270 Full-Screen Mode

When your workstation message queue is in break mode (you must specify *BREAK on the CHGMSGQ command), messages appear on the 3270 device exactly as they appear on the 5250 display. When your workstation is not in break mode, the following message is displayed: A message has arrived on a message queue. To continue, press the function key assigned to the help function or the function key assigned the error reset function. Then use the Display Message (DSPMSG) command or the function key assigned to the system request function followed by option 4 (Display message) to view the waiting message. Set the workstation message queue to break mode to see the messages as they arrive.

Input-Inhibited Light

When using an AS/400 system from a 5250-type terminal, pressing certain keys in certain situations causes input to be inhibited and the input-inhibited light to be displayed on the 5250 terminal. When using Telnet server 3270 full-screen mode, the input-inhibited light is indicated by two asterisks shown in the lower right corner of the display (see Figure 123).

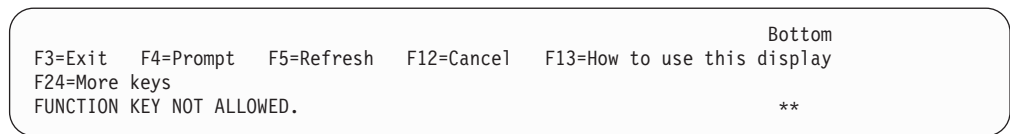


Figure 123. Input-Inhibited Light

When the keyboard is inhibited, any keys mapped to the AS/400 function keys are ignored. The keyboard must be reset by pressing the Enter key or by pressing the key mapped to the AS/400 Reset key.

Defining Capabilities for 3270 Devices

Table 17 lists the capabilities of the 3270 devices supported by Telnet.

Make sure that your Telnet client 3270 is negotiating one of the supported 3270 terminal types. The supported terminal types are shown in Table 19 on page 230.

Table 17. 3270 Device Capabilities

Device Type	Device Capabilities
3277	This display station supports generic 3270 data streams. Extended attributes, such as underlining, blinking, reverse image, or color are not supported.

Table 17. 3270 Device Capabilities (continued)

Device Type	Device Capabilities
3278	<p>This display station supports extended attributes, such as blinking, reverse image, and underlining if requested by the OS/400 DDS (data description specifications) keywords.</p> <p>Notes:</p> <ol style="list-style-type: none">1. Extended attributes are not supported by some client implementations of TELNET 3270 full-screen mode (TN3270).2. DBCS terminals that negotiate a 3278-2-E terminal type are supported.
3279	<p>This display station supports color attributes and the extended data stream attributes sent for a 3278 device. The color attributes are determined (in the same manner as a 5292 Full Color Display) by interpreting the DDS attributes as blinking, high intensity, or the DDS color keywords.</p>

VTxxx Full-Screen Mode

VTxxx server support allows Telnet client users to log on and run AS/400 5250 full-screen applications even though VTxxx full-screen support is negotiated. The Telnet client application must be able to negotiate VTxxx terminal support. When VTxxx full-screen mode is negotiated, the AS/400 Telnet server is responsible for mapping 5250 functions to VTxxx keys and vice versa.

Although the AS/400 Telnet server supports VTxxx clients, this is not the preferred mode to use because the AS/400 system is a block mode system, and the VTxxx terminal is a character mode device. Most Telnet implementations support a TN3270 or TN5250 client that should be used when connecting to an AS/400 Telnet server.

In general, when a key on a VTxxx terminal is pressed, the hexadecimal code associated with that key is immediately transmitted to the Telnet server. The Telnet server must process that keystroke and then echo that character back to the VTxxx terminal to be displayed. This results in a large amount of overhead associated with each keystroke. In contrast, the 5250 and 3270 block mode devices buffer all keystrokes at the client system until an Attention Identifier (AID) key is pressed. When an AID key is pressed, the client sends the buffered input to the server for processing. The block mode devices result in less overhead per keystroke and generally provide better performance than a character-mode device, such as the VTxxx terminal.

VTxxx delivers the data between the two systems as ASCII.

Setting up for VTxxx Full-Screen Mode

You can use the CFGTCPTLN command to set up your VTxxx full-screen mode session.


```

                                Configure TCP/IP TELNET
                                System:  SYSNAM01

Select one of the following:

    1. Change TELNET attributes
    2. Set VT mapping tables
    3. Display VT keyboard map
    4. Change VT keyboard map
    5. Set VT keyboard map

Work with associated system values:
    10. Autoconfigure virtual devices
    11. Limit security officer device access
    12. Inactive job time-out
    13. Inactive job message queue
    14. Limit device sessions
    15. Action to take for failed sign-on attempts
    16. Maximum sign-on attempts allowed

More...

Selection or command
====>

F3=Exit  F4=Prompt  F9=Retrieve  F12=Cancel

```

Figure 124. CFGTCPTELN in VTxxx Full-Screen Session

Step 1—VTxxx—Starting the Telnet Server Job

The server job for a TCP/IP application must be started in the QSYSWRK subsystem. The Start TCP/IP Server (STRTCPSVR) command starts the servers that are shipped with the TCP/IP Utilities licensed program.

Even though the Change Telnet Attributes (CHGTELNA) command has an AUTOSTART parameter, that parameter is overridden or ignored by the STRTCPSVR command.

Step 2—VTxxx—Setting the Number of Virtual Devices

Virtual devices are used by the server system to direct output to devices on your system. AS/400 Telnet server support automatically selects (and creates, if necessary) these devices for you. You may also choose to create your own virtual device under the QVIRCDnnnn virtual controller.

The option is available for you to allow the Telnet server support on the AS/400 system to automatically configure virtual controllers and devices. The QAUTOVRT system value specifies the maximum number of devices that are automatically configured by the system. Use the Change System Value (CHGSYSVAL) command to change the value of the QAUTOVRT system value. For example, entering the following command string changes the number of virtual devices that can be allocated on a system to 50:

```
CHGSYSVAL SYSVAL(QAUTOVRT) VALUE(50)
```

Note: QAUTOVRT has been modified for Version 4 Release 2 to support numeric values of 0 through 32500, and a special value of *NOMAX.

To determine and set the maximum number of users you want signed on to the AS/400 system at any time, do the following:

1. Set the QAUTOVRT value to 32500, the maximum value allowed, or use the *NOMAX value.
2. Let your users use pass-through, Telnet, and the virtual terminal application program interface until you decide that the number of virtual devices created is sufficient for normal system operation.
3. Change the QAUTOVRT value from 32500 to the number of virtual devices you require for normal system operation.

If you have never allowed automatic configuration of virtual devices on your system, the QAUTOVRT value is 0. A Telnet connection attempt with a dependence on automatic creation of the virtual device then fails because the Telnet server does not create more than the specified QAUTOVRT devices (zero). If you try to connect, you receive a message (TCP2504) indicating that the Telnet client session has ended and the connection is closed. In addition, the QTGTELNETS job in the QSYSWRK subsystem on the AS/400 Telnet server sends a message (CPF8940) indicating that a virtual device cannot be automatically selected.

If you change the QAUTOVRT value to 10, the next Telnet connection attempt causes the Telnet server to create a virtual device. This virtual device is created because the number of virtual devices on the controller (0) is less than the number specified in the QAUTOVRT (10). Even if you change the specified number to 0 again, the next user attempting a Telnet connection succeeds. When a Telnet connection attempt fails, the CPF87D7 message is sent to the system operator message queue on the Telnet server system. The CPF87D7 message indicates that the AS/400 server is not able to create a virtual device.

Security Considerations for VTxxx Full-Screen Mode: The number of sign-on attempts allowed increases if virtual devices are automatically configured. The number of sign-on attempts is equal to the number of system sign-on attempts allowed multiplied by the number of virtual devices that can be created. The number of system sign-on attempts allowed is defined by the QMAXSIGN system value. The number of virtual devices that can be created is defined by the QAUTOVRT system value.

In Version 4 Release 2, the following level of support has been added with regard to security of virtual devices:

- With a user-supplied exit program, you can audit the number of sign-on attempts
- You have the ability to deny connections
- You have the ability to allow bypassing of the sign-on screen

For more information on Telnet exit points and how to use them, see “TELNET Exit Points” on page 541 in Appendix E. TCP/IP Application Exit Points and Programs.

Telnet and SNA 5250 Pass-Through Considerations for VTxxx Full-Screen Mode: The AS/400 system supports 5250 pass-through. 5250 pass-through is similar to Telnet but runs on an SNA (Systems Network Architecture) protocol network rather than a TCP/IP network. 5250 pass-through uses virtual displays to direct output to the physical devices just as Telnet does. In 5250 pass-through, the AS/400 system automatically creates virtual devices in the same way that it does for Telnet. Therefore, the QAUTOVRT system value controls the number of automatically configured virtual devices for both 5250 pass-through and Telnet. For more information about 5250 pass-through, see the *Remote Work Station Support*, SC41-5402-00 book.

Step 3—VTxxx—Setting the QLMTSECOFR Value

The OS/400 licensed program supports the limit security officer (QLMTSECOFR) system value, which limits the devices the security officer can sign on to. If the QLMTSECOFR value is greater than zero, the security officer must be authorized to use the virtual device descriptions. However, when this value is 0, the system does not limit the devices users with *ALLOBJ or *SERVICE special authority can sign on to.

On AS/400 systems with a QSECURITY value of 30 or greater, a user with security officer authority (*ALLOBJ) must be authorized to use devices before the system allows the user to use those devices. For example, each display device that a security officer wants to sign on to (local, remote, or virtual), must have had the following authority specified with the Grant Object Authority (GRTOBJAUT) command:

```
GRTOBJAUT OBJ(display_name) OBJTYPE(*DEVD)
          AUT(*CHANGE) USER(QSECOFR)
```

This procedure is very important because Telnet automatically configures virtual devices. If the QLMTSECOFR value is set to 0, all devices automatically configured by Telnet can be used by the security officer. If you set the QLMTSECOFR value to 1, your security officer is not able to use the virtual devices created by Telnet unless you grant object authority to the security officer for that virtual device. The automatic configuration support can delete and re-create the virtual device. If this occurs, authority must be granted to the security officer each time the virtual device is created.

Step 4—VTxxx—Working with Associated System Values

In addition to the QAUTOVRT and QLMTSECOFR, the following system values are available for you to work with from the Configure TCP/IP Telnet (CFGTCPTELN) menu:

- QINACTITV: Inactive job time-out
- QINACTMSGQ: Inactive job message queue
- QLMTDEVSSN: Limit device sessions
- QMAXSGNACN: Action to take for failed sign-on attempts
- QMAXSIGN: Maximum sign-on attempts allowed
- QRMTSIGN: Remote sign-on control
- QDEVRCYACN: Device I/O error action
- QDSCJOBITV: Time interval before disconnected jobs end

Figure 124 on page 197 shows the Configure TCP/IP Telnet (CFGTCPTELN) menu.

Setting the Telnet Timemark Timeout Value: You should also take into consideration the TIMMRKTIMO parameter.

The Telnet timemark timeout (TIMMRKTIMO) parameter specifies the number of seconds between TIMEMARK commands sent by the Telnet server. If Telnet is unable to send the TIMEMARK command, it closes the connection.

Step 5—VTxxx—Creating Virtual Controllers and Devices

You can create virtual controllers and devices. If you create your own virtual devices, by allowing the system to automatically select the device name, you must be aware of the following:

- The virtual controller must be named QPACTLnn, where nn is a decimal number 01 or greater.
- The virtual device should be named QPADEVxxxx, where xxxx is an alphanumeric character from 0001 to ZZZZ.

Note: Starting with Version 4 Release 2, the xxxx are no longer only numeric characters, but also alphanumeric characters from 0001 to ZZZZ, allowing a maximum of 1,679,615 unique names (devices).

If you want to use more than 32500 devices, which is the maximum value for the QAUTOVRT system, you can set the QAUTOVRT system value to *NOMAX to allow additional devices to be created.

- The Telnet server reuses available existing virtual devices that were auto-created by selecting virtual devices of the same device type and model. When there are no more device type and model matches, but there are still available virtual devices, then the device type and model will be changed to match the client device and model negotiated. This is true only for auto-created (QPADEVnnn) virtual devices. Typically, the auto-created virtual device will use the AS/400 system values for keyboard type, character set, and code page. Optionally, these display device attributes may be more specifically defined through the exit program or device specified client subnegotiation. Devices can also be selected via the exit program interface as opposed to being negotiated.
- The Device Type for a virtual device with VTxxx emulation is V100.

Step 6—VTxxx—Defining Workstations to Subsystems

When you use Telnet to sign-on to an AS/400 server, the sign-on screen may not be displayed on your workstation. Before a user can sign on to the AS/400 server, the workstation must be defined to the subsystem. If the workstation has not been defined to the subsystem, you need to add a workstation entry to the subsystem description under which you want your job to run on the AS/400 server. The workstation in this case is the virtual display device automatically created by the Telnet server (QPADEVxxxx). The workstation name or the workstation type must be specified in the subsystem description on the AS/400 server. Use the Display Subsystem Description (DSPSBSD) command to see the workstation entries defined to a subsystem. (This only applies to display devices. Printer devices typically run in the QSPL subsystem.)

Note: The Add Work Station Entry (ADDWSE) command can be done when the subsystem is active. However, the changes may or may not take effect immediately. You may need to end and restart the subsystem.

Step 7—VTxxx—Activating the QSYSWRK Subsystem

The QSYSWRK subsystem must be active. Use the Work with Subsystem (WRKSBS) command to display the status of the subsystem.

The Telnet server must also be started. The interactive subsystem, QINTER, which is used in previous examples in this chapter, needs to be started to run interactive jobs for Telnet sessions. The spooling subsystem (QSPL) needs to be active to run printer pass-through sessions.

Step 8—VTxxx—Creating User Profiles for Telnet Users

At the server system, create one or more user profiles for Telnet users from other systems. The default user profile is *SYS. The following example shows a sample user profile:

```
CRTUSRPRF  USRPRF(CLERK1)
           PASSWORD(unique-password)
           JOBD(CLERKLIB/CLERKL1)
           TEXT('User profile for Clerks Group 1')
```

Step 9—VTxxx—Checking the QKBDTYPE System Value

When the AS/400 Telnet server automatically creates virtual display devices, it uses the QKBDTYPE system value to determine the keyboard type for the virtual device.

If the initial creation of the virtual device fails using the QKBDTYPE system value, the Telnet server attempts to create the device again, using a keyboard type value of USB. If the second attempt to create the keyboard type fails, then a message (CPF87D7) is sent to the QTCPIP job log, indicating that the virtual device cannot be automatically selected. This message is also sent to the system operator message queue.

Step 10—VTxxx—Setting the Default Keyboard Mapping

When a Telnet session is negotiated in VTxxx full-screen mode, a default keyboard map is used. To *display* the default keyboard map for VTxxx, use the Display VT Keyboard Map command (DSPVTMAP) (see “Displaying a VTxxx Keyboard Map” on page 205). To *change* the VTxxx keyboard map, use the Change VT Keyboard Map (CHGVMTMAP) command (see “Changing a VTxxx Keyboard Map” on page 206) or the Set VT Keyboard Map (SETVTMAP) command (see “Setting a VTxxx Keyboard Map” on page 206).

Because the VTxxx keyboard does not have the same keys as a 5250 keyboard, a keyboard mapping must exist between the VTxxx keys and the AS/400 functions. The AS/400 server assigns a default keyboard mapping when a VTxxx session is first established. In some cases there can be more than one key or key sequence that maps to a particular AS/400 function. In these cases, you can use any of the defined keys to call the desired AS/400 function. Table 18 on page 202 shows the 5250 functions along with the default VTxxx key or key sequences that are mapped to these functions.

Notes:

1. Each control character is a one-byte value that is generated from a VTxxx keyboard by holding down the CTRL key while pressing one of the alphabetic keys. Both shifted and unshifted control characters generate the same hexadecimal values.
2. The escape sequences are multiple byte codes that are generated by pressing the Esc key followed by the characters that make up the desired sequence.
3. The AS/400 server ignores the case of all alphabetic characters in an escape sequence. You can type alphabetic characters in escape sequences in either uppercase or lowercase.

4. The AS/400 F1-F12 functions are mapped to the Esc key followed by one of the keys in the top row of a VTxxx keyboard. The F13-F24 functions are mapped to the Esc key followed by a shifted key in the top row of a VTxxx keyboard.
5. Some Telnet VTxxx client systems use Ctrl-S and Ctrl-Q for flow control purposes. This is generally referred to as XON/XOFF flow control. If you are using a client system that has XON/XOFF enabled, you should not use the values *CTLS and *CTLQ in your keyboard mapping.

Table 18. Special Values for VTxxx Keys

Default 5250 Function	Special Value	VTxxx Keys	Hexadecimal Value ¹
Attention	*CTLA	<CTRL-A>	X'01'
	*ESCA	<ESC><A>	X'1B41'
Backspace	*BACKSPC	<Backspace or CTRL-H>	X'08'
Clear Screen	*ESCC	<ESC><C>	X'1B43'
Cursor Down	*CSRDOWN	<Down Arrow>	X'1B5B42'
Cursor Left	*CSRLEFT	<Left Arrow>	X'1B5B44'
Cursor Right	*CSRRIGHT	<Right Arrow>	X'1B5B43'
Cursor Up	*CSRUP	<Up Arrow>	X'1B5B41'
Delete	*DLT	<Delete>	X'7F'
	*RMV	<Remove>	X'1B5B337E' ² X'9B337E' ³
Duplicate	*ESCD	<ESC><D>	X'1B44'
Enter	*RETURN	<Return or CTRL-M>	X'0D'
Erase Input	*CTLE	<CTRL-E>	X'05'
Error Reset	*CTLR	<CTRL-R>	X'12'
	*ESCR	<ESC><R>	X'1B52'
Field Advance	*TAB	<TAB or CTRL-I>	X'09'
Field Backspace	*ESCTAB	<ESC><Tab or CTRL-I>	X'1B09'
Field Exit	*CTLK	<CTRL-K>	X'0B'
	*CTLX	<CTRL-X>	X'18'
	*ESCX	<ESC><X>	X'1B58'
Field Minus	*ESCM	<ESC><M>	X'1B4D'
Help	*CTLQST	<CTRL-Question Mark>	X'1F'
	*ESCH	<ESC><H>	X'1B48'
Home	*CTLO	<CTRL-O>	X'0F'
Insert	*ESCI	<ESC><I>	X'1B49'
	*ESCDLT	<ESC><Delete>	X'1B7F'
	*INS	<Insert Here>	X'1B5B327E' ² X'9B327E' ³
New Line	*ESCLF	<ESC> <Line Feed or CTRL-J>	X'1B0A'

Table 18. Special Values for VTxxx Keys (continued)

Default 5250 Function	Special Value	VTxxx Keys	Hexadecimal Value ¹
Page Down (Roll Up)	*CTLD	<CTRL-D>	X'04'
	*CTLF	<CTRL-F>	X'06'
	*NXTSCR	<Next Screen>	X'1B5B367E ² X'9B367E ³
Page Up (Roll Down)	*CTLB	<CTRL-B>	X'02'
	*CTLU	<CTRL-U>	X'15'
	*PRVSCR	<Prev Screen>	X'1B5B357E ² X'9B357E ³
Print	*CTLP	<CTRL-P>	X'10'
	*ESCP	ESC	X'1B50'
Redraw Screen	*CTLL	<CTRL-L>	X'0C'
	*ESCL	<ESC><L>	X'1B4C'
System Request	*CTLC	<CTRL-C>	X'03'
	*ESCS	<ESC><S>	X'1B53'
Test Request	*CTLT	<CTRL-T>	X'14'
Toggle Indicator Lights	*ESCT	<ESC><T>	X'1B54'
F1	*ESC1	<ESC><1>	X'1B31'
	*F1	<F1> ⁵	X'1B5B31317E ² X'9B31317E ³
	*PF1	<PF1>	X'1B4F50 ² X'8F50 ³
F2	*ESC2	<ESC><2>	X'1B32'
	*F2	<F2> ⁵	X'1B5B31327E ² X'9B31327E ³
	*PF2	<PF2>	X'1B4F51 ² X'8F51 ³
F3	*ESC3	<ESC><3>	X'1B33'
	*F3	<F3> ⁵	X'1B5B31337E ² X'9B31337E ³
	*PF3	<PF3>	X'1B4F52 ² X'8F52 ³
F4	*ESC4	<ESC><4>	X'1B34'
	*F4	<F4> ⁵	X'1B5B31347E ² X'9B31347E ³
	*PF4	<PF4>	X'1B4F53 ² X'8F53 ³
F5	*ESC5	<ESC><5>	X'1B35'
	*F5	<F5> ⁵	X'1B5B31357E ² X'9B31357E ³

Table 18. Special Values for VTxxx Keys (continued)

Default 5250 Function	Special Value	VTxxx Keys	Hexadecimal Value ¹
F6	*ESC6	<ESC><6>	X'1B36'
	*F6	<F6>	X'1B5B31377E' ² X'9B31377E' ³
F7	*ESC7	<ESC><7>	X'1B37'
	*F7	<F7>	X'1B5B31387E' ² X'9B31387E' ³
F8	*ESC8	<ESC><8>	X'1B38'
	*F8	<F8>	X'1B5B31397E' ² X'9B31397E' ³
F9	*ESC9	<ESC><9>	X'1B39'
	*F9	<F9>	X'1B5B32307E' ² X'9B32307E' ³
F10	*ESC0	<ESC><0>	X'1B30'
	*F10	<F10>	X'1B5B32317E' ² X'9B32317E' ³
F11	*ESCMINUS	<ESC><Minus>	X'1B2D'
	*F11	<F11>	X'1B5B32337E' ² X'9B32337E' ³
F12	*ESCEQ	<ESC><Equal>	X'1B3D'
	*F12	<F12>	X'1B5B32347E' ² X'9B32347E' ³
F13	*ESCEXCL	<ESC><Exclamation>	X'1B21'
	*F13	<F13>	X'1B5B32357E' ² X'9B32357E' ³
F14	*ESCAT	<ESC><At sign>	X'1B40'
	*F14	<F14>	X'1B5B32367E' ² X'9B32367E' ³
F15	*ESCPOUND	<ESC><Pound>	X'1B23'
	*F15	<F15>	X'1B5B32387E' ² X'9B32387E' ³
F16	*ESCDOLLAR	<ESC><Dollar>	X'1B24'
	*F16	<F16>	X'1B5B32397E' ² X'9B32397E' ³
F17	*ESCPCT	<ESC><Percent>	X'1B25'
	*F17	<F17>	X'1B5B33317E' ² X'9B33317E' ³
F18	*ESCCFX	<ESC><Circumflex Accent>	X'1B5E' ¹
	*F18	<F18>	X'1B5B33327E' ² X'9B33327E' ³

Table 18. Special Values for VTxxx Keys (continued)

Default 5250 Function	Special Value	VTxxx Keys	Hexadecimal Value ¹
F19	*ESCAMP	<ESC><Ampersand>	X'1B26'
	*F19	<F19>	X'1B5B33337E' ²
			X'9B33337E' ³
F20	*ESCAST	<ESC><Asterisk>	X'1B2A'
	*F20	<F20>	X'1B5B33347E' ²
			X'9B33347E' ³
F21	*ESCLPAR	<ESC><Left Parenthesis>	X'1B50'
F22	*ESCRPAR	<ESC><Right Parenthesis>	X'1B51'
F23	*ESCUS	<ESC><Underscore>	X'1B5F'
F24	*ESCPLUS	<ESC><Plus>	X'1B2B'
See note 4	*FIND	<Find>	X'1B5B317E'
			X'9B317E'
See note 4	*SELECT	<Select>	X'1B5B347E'
			X'9B347E'

Notes:

1. Unless otherwise identified, the hexadecimal value is in the VT100 mode.
2. VT220 7-bit control mode.
3. VT220 8-bit control mode.
4. There is no 5250 function key that maps to this VT key.
5. The keys F1 through F5 are not available on a VT220 terminal. However, many VT220 emulators send these hexadecimal values when the F1 through F5 keys are pressed.

Displaying a VTxxx Keyboard Map: You can display the current keyboard mapping with the Display VT Keyboard Map (DSPVTMAP) command. There are no parameters for the DSPVTMAP command. You are shown the VTxxx keys that are mapped to the AS/400 functions.

The DSPVTMAP command is only valid when called from within an AS/400 Telnet server session operating in VTxxx full-screen mode.

Type DSPVTMAP to see the following display, and then press the Page Down key to see the additional displays. You can display the VT keyboard map using option 3 from the Configure TCP/IP Telnet menu.

5250 Function	Display VT Keyboard Map VT100/VT220 key(s)		
5250 Attention	*CTLA	*ESCA	
5250 Help	*CTLQST	*ESCH	
Page Down (Roll Up)	*CTLD	*CTLF	*NXTSCR
Page Up (Roll Down)	*CTLB	*CTLU	*PRVSCR
System Request	*CTLC	*ESCS	
Insert	*ESCI	*ESCDLT	
Delete	*DLT		
Enter	*RETURN		
Backspace	*BACKSPC		
Duplicate	*ESCD		
Erase Input	*CTLE		
Error Reset	*CTLR	*ESCR	
Field Exit	*CTLK	*CTLX	*ESCX
Field Minus	*ESCM		
Home	*CTLO		
New Line	*ESCLF		

More

F1=Help F3=Exit F12=Cancel

Setting a VTxxx Keyboard Map: You can change the default keyboard mapping using the Set VT Keyboard Map (SETVTMAP) command or using option 5 (Set VT keyboard map) from the Configure TCP/IP Telnet menu while your terminal is in VTxxx emulation mode. When the command is used without any user-specified parameters, the default keyboard mapping specified in Table 18 on page 202 is restored. You can specify up to four of the defined special values for each parameter. A special value cannot be used to specify more than one AS/400 function.

Type SETVTMAP to show the keyboard mapping displays. You are shown the following initial display. Use the Page Down key to see the remainder of the displays.

Set VT Keyboard Map (SETVTMAP)		
Type choices, press Enter.		
5250 Attention	*CTLA	*CTLA, *CTLB, *CTLC...
+ for more values	*ESCA	
5250 Help	*CTLQST	*CTLA, *CTLB, *CTLC...
+ for more values	*ESCH	
Page Down (Roll Up)	*CTLD	*CTLA, *CTLB, *CTLC...
	*CTLF	
+ for more values	*NXTSCR	
Page Up (Roll Down)	*CTLB	*CTLA, *CTLB, *CTLC...
	*CTLU	
+ for more values	*PRVSCR	
System Request	*ESCS	*CTLA, *CTLB, *CTLC...
+ for more values	*CTLC	
Insert	*ESCI	*CTLA, *CTLB, *CTLC...
	*ESCDLT	
+ for more values	*INS	

More...

F3=Exit F4=Prompt F5=Refresh F12=Cancel F13=How to use this display
F24=More keys

You can press F4 (Prompt) to display the parameter values.

Changing a VTxxx Keyboard Map: Like the SETVTMAP command, the Change VT Keyboard Map (CHGVVTMAP) command allows you to customize the keyboard mapping when connected to an AS/400 Telnet server operating in VTxxx mode. The difference between these two commands is that the parameters for the SETVTMAP command default to the shipped values while the parameters for the CHGVVTMAP command default to the currently set values. Except for this distinction, the two commands are identical.

The CHGV TMAP command is only valid when called from within an AS/400 Telnet server session operating in VTxxx full-screen mode. You can change the VT keyboard map using option 4 from the Configure TCP/IP Telnet menu.

Step 11—VTxxx—Setting the DFTNVTTYPE Value

The DFTNVTTYPE (default network virtual terminal type) parameter specifies the mode to be used when the Telnet server is not able to negotiate one of the supported terminal types. Use the CHGTELNA command to set the value for this parameter to either *VT100 for VT100 or VT220 mode or *NVT for ASCII line mode.

Step 12—VTxxx—Setting the ASCII/EBCDIC Mapping Tables

The AS/400 Telnet server uses default ASCII-to-EBCDIC and EBCDIC-to-ASCII mapping tables based on the CCSID parameter in the TCP/IP Telnet attributes. The default is to use the DEC multinational character set (*MULTINAT). Other 7-bit and 8-bit ASCII CCSIDs can also be used as well as any of the 7-bit DEC national replacement character sets. These are discussed in “VTxxx—National Mode” on page 180.

Note: For VT220 8-bit mode, the mapping tables are not available. The DEC replacement character sets must be used. For the VT220 7-bit mode, you can use either the mapping tables or the DEC replacement character sets.

You can change the default by changing the CCSID parameter, or by specifying different values for the VTxxx outgoing (TBLV TOUT) and incoming tables (TBLV TIN) using the CHGTELNA command, or by changing the default for the current session using the Set VT Mapping Tables (SETV TTB L) command. This command can also be accessed with option 2 on the CHGTCPTELN command when in VTxxx emulation.

VTxxx Automatic Wrap

The AS/400 VTxxx server requires the VTxxx client to have the automatic wrap (autowrap) option turned on. When automatic wrap is on, a character written to column 80 of the VTxxx causes the cursor to move to the first column of the next line. Please refer to your VTxxx client documentation for details of how to set on this option.

System Request Processing for VTxxx Sessions

The system request processing for the VTxxx sessions is slightly different than that for a normal 5250 workstation. When the System Request key is pressed on a 5250 workstation, a system request command line appears at the bottom of the display. If you press the Enter key, the System Request menu appears. For VTxxx sessions, the system request command line is not displayed when the system request function is called. Instead, the System Request menu is displayed immediately.

Error Conditions on 5250 Keyboard

Certain error conditions cause a 5250 keyboard to lock and an error code to be displayed on the operator error line, for example, typing when the cursor is not in an input field. For VTxxx sessions, these errors cause a bell to sound on the VTxxx terminal and the keyboard to remain unlocked.

Certain AS/400 applications also lock the 5250 keyboard and turn on the 5250 input-inhibited light. The user is then required to press the Error Reset key before the keyboard is unlocked. For VTxxx sessions, the locking of the 5250 keyboard causes a bell to sound on the VTxxx terminal whenever a key is pressed. To unlock the keyboard, the VTxxx key that is mapped to the Error Reset key must be pressed. In the default VTxxx keyboard map, the key CTL-R is mapped to the Error Reset key.

Display Screens and VTxxx Support

When VTxxx support is negotiated, the Telnet server transmits screens that are a maximum of 24 rows by 80 columns. The VTxxx client system sees these screens in much the same way as they appear on a 5251 Model 11 workstation. However, there are some differences.

- A 5251 display has indicator lights on the right side that indicate:
 - System Available
 - Message Waiting
 - Keyboard Shift
 - Insert Mode
 - Input-Inhibited

The VTxxx server support emulates the System Available, Message Waiting, Insert Mode, and Input-Inhibited lights by putting an asterisk in column 80 of rows 9, 11, 13, or 15, respectively. When an asterisk is displayed, the asterisk overwrites the character that was previously displayed at that screen location. By default, the VTxxx server does not display the indicator lights. You can enable or disable these indicators by typing the key sequence that is mapped to the toggle indicator lights function. The default key sequence for this function is ESC-T.

Note: When using a VTxxx client to attach to the AS/400 Telnet, note that the Insert Mode and the Input-Inhibited lights may not always display as described above. 5250 supports the attachment as a local function while the VTxxx has no such facility. The System Available and Message Waiting indicators, however, will display correctly.

- A 5251 display supports a screen attribute known as a column separator. The column separator is a vertical line displayed between characters. This line does not take up a character space. The VTxxx does not support such an attribute. If an AS/400 application generates a screen that uses the column separator attribute, that screen is displayed on the VTxxx client system with the column separator mapped to the VTxxx underline attribute.

VT220 Control Characters

When VT220 8-bit emulation is negotiated, the range of characters X'80' - X'9F' are protected as C1 control characters as architecturally defined in DEC *VT220 Programmer Reference Manual*. This may result in the succeeding characters in a data stream being interpreted as data in relation to these characters. If VT220 7-bit or VT100 is negotiated, then the full range of characters from X'80' - X'FF' is available for character translation. X'80 - X'9F' *must* be interpreted as C1 control characters in VT220 8-bit control mode only.

This has particular relevance to NLS support, as several non-English languages use these values for language specific characters, and so in these cases, the VT220 8-bit emulation may not function as anticipated.

Some Practical Examples

This topic discusses using TCP/IP Telnet VTxxx emulation with the following clients:

- DEC MicroVAX
- Sun Sparc Classic
- HP705 Apollo

DEC MicroVAX VT100 Full-Screen Telnet Client

The VT100 Telnet client provided by the Wollongong MicroVAX TCP/IP package was tested to the AS/400 VTxxx Telnet server. Although this testing was by no means thorough, the connection appeared to work satisfactorily providing the Telnet session was initiated in the following way:

```
$ TELNET
TELNET> xon
Data flow control characters processed locally.
TELNET> promptmode
Will show 'TELNET>' prompt when connected.
TELNET> open 9.4.73.251
Trying 9.4.73.251.....
Connected to 9.4.73.251
Escape character is '^]'.
```

An AS/400 sign-on display was then received from SYSNAM123.

Note: Without xon turned on, the AS/400 display was corrupted.

The Telnet session from the MicroVAX could either be ended by using SIGNOFF ENDCNN(*YES) or by Entering QUIT or CLOSE at the *TELNET* prompt. To obtain the Telnet prompt, Enter the 'escape character' (Ctrl-] by default). 'Promptmode' must be enabled to obtain the Telnet prompt.

The connection was initiated from a VT320 terminal attached to the MicroVAX. The results may be different for different terminal types and the previous information is provided for guidance only. During testing, a requirement for a change to the default keyboard map became apparent because the F6 function key on the VT320 terminal being used appeared to perform a local function (Interrupt). The SETVTMAP command or the CHGVMTMAP command maps this function to another key if it were not possible to override this key being a local function.

Sun Sparc Classic VT100 Full-Screen Telnet Client

The VT100 Telnet client provided by the Sun UNIX TCP/IP package was tested to the AS/400 VTxxx Telnet server. Although this testing was by no means thorough, the connection appeared to work satisfactorily using the default Telnet command parameters:

```
# TELNET
TELNET> open 9.4.73.251
Trying...
Connected to 9.4.73.251
Escape character is '^]'.
```

An AS/400 sign-on display was then received from SYSNAM123.

The Telnet session from the Sun system could either be ended by using SIGNOFF ENDCNN(*YES) or by entering QUIT or CLOSE at the *Telnet* prompt. To obtain the Telnet prompt, Enter the 'escape character' (Ctrl-] by default).

The connection was initiated from a console attached to the Sun system. The results may be different for different terminal types and the above information is provided for guidance only. During testing a requirement for a change to the default keyboard map became apparent. The F11 and F12 function keys did not appear to perform the normal AS/400 functions. (F11 resulted in a print screen function and F12 did nothing.) The SETVTMAP command or the CHGVTMAP command maps these functions to other keys.

HP 705 Apollo VT100 Full-Screen Telnet Client

The VT100 Telnet client provided by the HP UNIX TCP/IP package was tested to the AS/400 VTxxx Telnet server. Although this testing was by no means thorough, the connection appeared to work using the default Telnet command parameters from an *xterm* session on the HP X Window display:

```
# TELNET
TELNET> open 9.4.73.251
Trying.....
Connected to 9.4.73.251
Escape character is '^]'
```

An AS/400 sign-on display was then received from SYSNAM123.

Note: An *xterm* session is initiated by typing the *xterm* command from an *hpterm* session.

The Telnet session from the HP system could be ended by using SIGNOFF ENDCNN(*YES) or by entering QUIT or CLOSE at the *TELNET* prompt. To obtain the Telnet prompt, Enter the 'escape character' (Ctrl-] by default).

The connection was initiated from an HP X-station attached to the HP system. The results may be different for different terminal types and the previous information is provided for guidance only. During the testing a requirement for a change to the default keyboard map became apparent because the HP X-station appeared to have function keys F1 to F8 only. The SETVTMAP command or the CHGVTMAP command maps these functions to other keys.

ASCII Line Mode

If 5250 full-screen mode or 3270 full-screen mode cannot be negotiated, and if the AS/400 Telnet server is not configured to default to VT mode, ASCII line mode is used. ASCII line mode is the standard Telnet network virtual terminal (NVT) support. The AS/400 Telnet server supports interaction with Telnet clients in ASCII line mode. A **network virtual terminal (NVT)** is a type of AS/400 virtual display station that represents an ASCII line-mode type of physical display station when the AS/400 system is the server in a TCP/IP Telnet connection.

Because the AS/400 system operates in full-screen mode and has screens with many input fields, it is difficult to map these screens to a line mode device. Therefore, consider the following when using ASCII line mode support for the AS/400 Telnet server:

- You cannot sign on to the AS/400 system from a non-IBM system and use workstation functions unless the non-IBM system provides Telnet 3270, Telnet 5250, VT100, or VT220 support.
- A sign-on screen for the AS/400 system is not automatically displayed when ASCII line mode is negotiated.
- You must create a controller description for virtual workstations and device descriptions for NVT workstations.
- You must have an application program running that opens a display file to the NVT workstation devices.
- The programming language chosen for the application program must support display files.
- Your application program must use a **user-defined data stream (UDDS)** to interact with the device. UDDS is a data stream in which the user has defined and embedded all device characters and allows AS/400 users to send their own workstation data stream to a display device. This interface is described in the *Application Display Programming* book.
- Your application program must format all data and provide security. Your application code must control access to the NVT device (provide password control, for example).
- If your application program does not acquire a virtual device before the Telnet inactivity timer expires, the AS/400 server system ends the connection. A **virtual device** is a device description that does not have hardware associated with it. It is used to form a connection between a user and a physical workstation attached to a remote system.
- Data is passed to the application without being mapped from ASCII to EBCDIC.

Setting up for ASCII Line Mode

To set up the Telnet server for ASCII line mode, you need to set the number of automatically created virtual devices.

To run a workstation application in ASCII line mode, you must create and vary on a controller description for a virtual workstation and a device description for a network virtual terminal (NVT) display. The Telnet server support selects an NVT device description that is varied on and not being used by another user. Your Telnet server application must open a display file that uses the selected NVT device.

You should already have read “Setting Up the Telnet Server” on page 183. In addition to the steps discussed in that topic, you need to perform the following:

1. Create a controller description for a virtual workstation.
2. Create a network virtual device.
3. Create an NVT application program.

Step 1—ASCII—Starting the Telnet Server Job

The server job for a TCP/IP application must be started in the QSYSWRK subsystem. The Start TCP/IP Server (STRTCPSVR) command starts the servers that are shipped with the TCP/IP Utilities licensed program.

Even though the Change Telnet Attributes (CHGTELNA) command has an AUTOSTART parameter, that parameter is overridden or ignored by the STRTCPSVR command.

Step 2—ASCII—Setting the Number of Virtual Devices

Virtual devices are used by the server system to direct output to devices on your system. AS/400 Telnet server support automatically selects (and creates, if necessary) these devices for you. You may also choose to create your own virtual device under the QVIRCDnnnn virtual controller.

The option is available for you to allow the Telnet server support on the AS/400 system to automatically configure virtual controllers and devices. The QAUTOVRT system value specifies the maximum number of devices that are automatically configured by the system. Use the Change System Value (CHGSYSVAL) command to change the value of the QAUTOVRT system value. For example, entering the following command string changes the number of virtual devices that can be allocated on a system to 50:

```
CHGSYSVAL SYSVAL(QAUTOVRT) VALUE(50)
```

Note: QAUTOVRT has been modified for Version 4 Release 2 to support numeric values of 0 through 32500, and a special value of *NOMAX.

To determine and set the maximum number of users you want signed on to the AS/400 system at any time, do the following:

1. Set the QAUTOVRT value to 32500, the maximum value allowed, or use the *NOMAX value.
2. Let your users use pass-through, Telnet, and the virtual terminal application program interface until you decide that the number of virtual devices created is sufficient for normal system operation.
3. Change the QAUTOVRT value from 32500 to the number of virtual devices you require for normal system operation.

If you have never allowed automatic configuration of virtual devices on your system, the QAUTOVRT value is 0. A Telnet connection attempt with a dependence on automatic creation of the virtual device then fails because the Telnet server does not create more than the specified QAUTOVRT devices (zero). If you try to connect, you receive a message (TCP2504) indicating that the Telnet client session has ended and the connection is closed. In addition, the QTGTELNETS job in the QSYSWRK subsystem on the AS/400 Telnet server sends a message (CPF8940) indicating that a virtual device cannot be automatically selected.

If you change the QAUTOVRT value to 10, the next Telnet connection attempt causes the Telnet server to create a virtual device. This virtual device is created because the number of virtual devices on the controller (0) is less than the number specified in the QAUTOVRT (10). Even if you change the specified number to 0 again, the next user attempting a Telnet connection succeeds. When a Telnet connection attempt fails, the CPF87D7 message is sent to the system operator message queue on the Telnet server system. The CPF87D7 message indicates that the AS/400 server is not able to create a virtual device.

Security Considerations for ASCII Full-Screen Mode

The number of sign-on attempts allowed increases if virtual devices are automatically configured. The number of sign-on attempts is equal to the number of system sign-on attempts allowed multiplied by the number of virtual devices that can be created. The number of system sign-on attempts allowed is defined by the QMAXSIGN system value. The number of virtual devices that can be created is defined by the QAUTOVRT system value.

In Version 4 Release 2, the following level of support has been added with regard to security of virtual devices:

- With a user-supplied exit program, you can audit the number of sign-on attempts
- You have the ability to deny connections
- You have the ability to allow bypassing of the sign-on screen

For more information on Telnet exit points and how to use them, see “TELNET Exit Points” on page 541 in Appendix E. TCP/IP Application Exit Points and Programs.

Telnet and SNA 5250 Pass-Through Considerations for ASCII Full-Screen Mode

The AS/400 system supports 5250 pass-through. 5250 pass-through is similar to Telnet but runs on an SNA (Systems Network Architecture) protocol network rather than a TCP/IP network. 5250 pass-through uses virtual displays to direct output to the physical devices just as Telnet does. In 5250 pass-through, the AS/400 system automatically creates virtual devices in the same way that it does for Telnet. Therefore, the QAUTOVRT system value controls the number of automatically configured virtual devices for both 5250 pass-through and Telnet. For more information about 5250 pass-through, see the *Remote Work Station Support* book.

Step 3—ASCII—Setting the QLMTSECOFR Value

The OS/400 licensed program supports the limit security officer (QLMTSECOFR) system value, which limits the devices the security officer can sign on to. If the QLMTSECOFR value is greater than zero, the security officer must be authorized to use the virtual device descriptions. However, when this value is 0, the system does not limit the devices users with *ALLOBJ or *SERVICE special authority can sign on to.

On AS/400 systems with a QSECURITY value of 30 or greater, a user with security officer authority (*ALLOBJ) must be authorized to use devices before the system allows the user to use those devices. For example, each display device that a security officer wants to sign on to (local, remote, or virtual), must have had the following authority specified with the Grant Object Authority (GRTOBJAUT) command:

```
GRTOBJAUT OBJ(display_name) OBJTYPE(*DEVD)
          AUT(*CHANGE) USER(QSECOFR)
```

This procedure is very important because Telnet automatically configures virtual devices. If the QLMTSECOFR value is set to 0, all devices automatically configured by Telnet can be used by the security officer. If you set the QLMTSECOFR value to 1, your security officer is not able to use the virtual devices created by Telnet unless you grant object authority to the security officer for that virtual device. The automatic configuration support can delete and re-create the virtual device. If this occurs, authority must be granted to the security officer each time the virtual device is created.

Step 4—ASCII—Working with Associated System Values

In addition to the QAUTOVRT and QLMTSECOFR, the following system values are available for you to work with from the Configure TCP/IP Telnet (CFGTCPTELN) menu:

- QINACTIVT: Inactive job time-out
- QINACTMSGQ: Inactive job message queue

- QLMTDEVSSN: Limit device sessions
- QMAXSGNACN: Action to take for failed sign-on attempts
- QMAXSIGN: Maximum sign-on attempts allowed
- QRMTSIGN: Remote sign-on control
- QDEVRCYACN: Device I/O error action
- QDSCJOBTV: Time interval before disconnected jobs end

Setting the Telnet Timemark Timeout Value: You should also take into consideration the TIMMRKTIMO parameter.

The Telnet timemark timeout (TIMMRKTIMO) parameter specifies the number of seconds between TIMEMARK commands sent by the Telnet server. If Telnet is unable to send the TIMEMARK command, it closes the connection.

Step 5—ASCII—Creating Virtual Controllers and Devices

You can create virtual controllers and devices. If you create your own virtual devices, by allowing the system to automatically select the device name, you must be aware of the following:

- The virtual controller must be named QPACTLnn, where nn is a decimal number 01 or greater.
- The virtual device should be named QPADEVxxxx, where xxxx is an alphanumeric character from 0001 to ZZZZ.

Note: Starting with Version 4 Release 2, the xxxx are no longer only numeric characters, but also alphanumeric characters from 0001 to ZZZZ, allowing a maximum of 1,679,615 unique names (devices).

If you want to use more than 32500 devices, which is the maximum value for the QAUTOVRT system, you can set the QAUTOVRT system value to *NOMAX to allow additional devices to be created.

- The Telnet server reuses available existing virtual devices that were auto-created by selecting virtual devices of the same device type and model. When there are no more device type and model matches, but there are still available virtual devices, then the device type and model will be changed to match the client device and model negotiated. This is true only for auto-created (QPADEVnnn) virtual devices. Typically, the auto-created virtual device will use the AS/400 system values for keyboard type, character set, and code page. Optionally, these display device attributes may be more specifically defined through the exit program or device specified client subnegotiation. Devices can also be selected via the exit program interface as opposed to being negotiated.

Step 6—ASCII—Defining Workstations to Subsystems

When you use Telnet to sign-on to an AS/400 server, the sign-on screen may not be displayed on your workstation. Before a user can sign on to the AS/400 server, the workstation must be defined to the subsystem. If the workstation has not been defined to the subsystem, you need to add a workstation entry to the subsystem description under which you want your job to run on the AS/400 server. The workstation in this case is the virtual display device automatically created by the Telnet server (QPADEVxxxx). The workstation name or the workstation type must be specified in the subsystem description on the AS/400 server. Use the Display Subsystem Description (DSPSBSD) command to see the workstation entries defined to a subsystem. (This only applies to display devices. Printer devices

typically run in the QSPL subsystem.) The following command can be used to add all workstation types to a subsystem named QINTER:

```
ADDWSE SBSD(QINTER) WRKSTNTYPE(*ALL)
```

Note: The Add Work Station Entry (ADDWSE) command can be done when the subsystem is active. However, the changes may or may not take effect immediately. You may need to end and restart the subsystem.

Step 7—ASCII—Activating the QSYSWRK Subsystem

The QSYSWRK subsystem must be active. Use the Work with Subsystem (WRKSBS) command to display the status of the subsystem.

The Telnet server must also be started. The interactive subsystem, QINTER, which is used in previous examples in this chapter, needs to be started to run interactive jobs for Telnet sessions. The spooling subsystem (QSPL) needs to be active to run printer pass-through sessions.

Step 8—ASCII—Creating User Profiles for Telnet Users

At the server system, create one or more user profiles for Telnet users from other systems. The default user profile is *SYS. The following example shows a sample user profile:

```
CRTUSRPRF  USRPRF(CLERK1)
           PASSWORD(unique-password)
           JOBD(CLERKLIB/CLERKL1)
           TEXT('User profile for Clerks Group 1')
```

Step 9—Creating a Controller Description for a Virtual Workstation

To create a controller description for a virtual workstation, use the Create Controller Description (Virtual Work Station) CRTCTLVWS command as follows:

```
CRTCTLVWS CTLD(NVTCTL)
          TEXT('VIRTUAL CONTROLLER FOR NVT DEVICES')
```

Step 10—Creating a Network Virtual Device

To create a network virtual device, use the Create Device Description (Display) (CRTDEVDSP) command as follows:

```
CRTDEVDSP  DEVD(NVT1) DEVCLS(*VRT)
           TYPE(*NVT) MODEL(0) CTL(NVTCTL)
           TEXT('NVT Display Device for Telnet')
```

Note: When an ASCII line mode session is negotiated, the AS/400 server system tries to find an NVT workstation device on the system. If no NVT workstation device is available, or if no customer application is active and using the NVT workstation device, the Telnet connection ends.

Keyboard Mapping: When in Telnet ASCII line mode, there is no keyboard mapping.

Telnet Printer Pass-Through Mode

The Telnet printer pass-through mode (TPPT) allows the AS/400 user with a Telnet client that supports printer emulation, to attach printer devices on the AS/400 over the network. This support is accomplished by negotiating the 3812 or 5553 device type support with the remote client Telnet application.

If you intend to use the TCP/IP Telnet printer pass-through, check with the client vendor or with third parties that are known to provide 5250 clients for the availability of the printer pass-through function.

Clients that support this function are:

- IBM Client Access for Win95
- Personal Communications Version 4 Release 2

Telnet Printer Pass-Through (TPPT) delivers the printer data stream between the two systems as either EBCDIC or ASCII depending on the preferences of the requesting client.

This topic explains the special considerations for a Telnet user operating in TPPT mode.

Figure 125 shows a typical Telnet TPPT-mode environment.

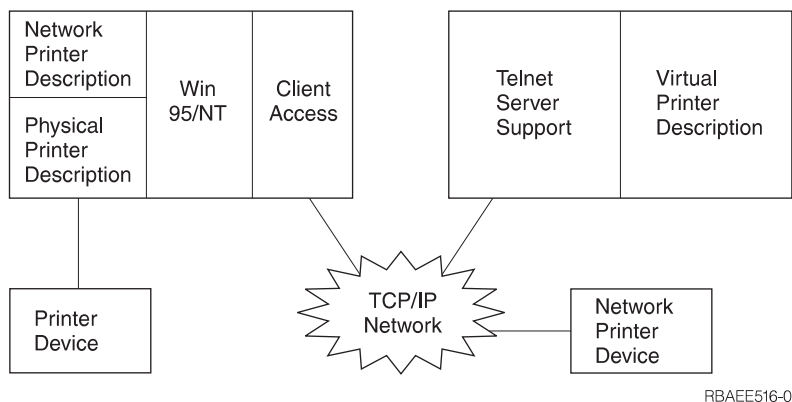


Figure 125. Telnet Printer Pass-Through-Mode Environment

TPPT printing support can only be negotiated with a Telnet client application on a system that supports Telnet printer emulation, for example, Client Access for Win95.

Note: When using Telnet printer pass-through, the print data must be spooled to a printer writer queue. (Direct printing to the device is not supported.) To ensure this, the print file used must specify *YES for the SPOOL parameter.

Telnet printer pass-through mode supports the following generic EBCDIC printer devices:

- IBM-3812-1 for single byte character set support (SBCS)
- IBM-5553-B01 for double byte character set devices (DBCS)

You can specify either of these generic device types more completely by requesting the AS/400 Host Print Transform (HPT) function and selecting the specific manufacturing type. AS/400 sends the printer data stream in ASCII.

Setting Up for Telnet Printer Pass-Through Mode

You can use the CFGTCPTELN command to set up your TPPT mode session.

Step 1—Telnet Printer Pass-Through—Starting the Telnet Server Job

The server job for a TCP/IP application must be started in the QSYSWRK subsystem. The Start TCP/IP Server (STRTCPSVR) command starts the servers that are shipped with the TCP/IP Utilities licensed program.

Even though the Change Telnet Attributes (CHGTELNA) command has an AUTOSTART parameter, that parameter is overridden or ignored by the STRTCPSVR command.

Step 2—Telnet Printer Pass-Through—Setting the Number of Virtual Devices

Virtual devices are used by the server system to direct output to devices on your system. AS/400 Telnet server support automatically selects (and creates, if necessary) these devices for you. You may also choose to create your own virtual device under the QVIRCDnnnn virtual controller.

The option is available for you to allow the Telnet server support on the AS/400 system to automatically configure virtual controllers and devices. The QAUTOVRT system value specifies the maximum number of devices that are automatically configured by the system. Use the Change System Value (CHGSYSVAL) command to change the value of the QAUTOVRT system value. For example, entering the following command string changes the number of virtual devices that can be allocated on a system to 50:

```
CHGSYSVAL SYSVAL(QAUTOVRT) VALUE(50)
```

Note: QAUTOVRT has been modified for Version 4 Release 2 to support numeric values of 0 through 32500, and a special value of *NOMAX.

To determine and set the maximum number of users you want signed on to the AS/400 system at any time, do the following:

1. Set the QAUTOVRT value to 32500, the maximum value allowed, or use the *NOMAX value.
2. Let your users use pass-through, Telnet, the virtual terminal application program interface and Telnet Printer pass-through until you decide that the number of virtual devices created is sufficient for normal system operation.
3. Change the QAUTOVRT value from 32500 to the number of virtual devices you require for normal system operation.

If you have never allowed automatic configuration of virtual devices on your system, the QAUTOVRT value is 0. A Telnet connection attempt with a dependence on automatic creation of the virtual device then fails because the Telnet server does not create more than the specified QAUTOVRT devices (zero). If you try to connect, you will receive a message (TCP2504) that indicates that the Telnet client session has ended and the connection is closed. In addition, the QTGTELNETS job in the

QSYSWRK subsystem on the AS/400 Telnet server sends a message (CPF8940) indicating that a virtual device cannot be automatically selected.

If you change the QAUTOVRT value to 10, the next Telnet connection attempt causes the Telnet server to create a virtual device. This virtual device is created because the number of virtual devices on the controller (0) is less than the number specified in the QAUTOVRT (10). Even if you change the specified number to 0 again, the next user attempting a Telnet connection succeeds. When a Telnet connection attempt fails, the CPF87D7 message is sent to the system operator message queue on the Telnet server system. The CPF87D7 message indicates that the AS/400 server is not able to create a virtual device.

Security Considerations for Telnet Printer Pass-Through Mode

Printer sessions are always made active, immediately after session initialization, which means that sign-on attempts are not necessary.

In Version 4 Release 2, the following level of support has been added with regard to security of printer devices:

- You have the ability to deny connections

For more information on Telnet exit points and how to use them, see “TELNET Exit Points” on page 541 in Appendix E. TCP/IP Application Exit Points and Programs.

Telnet and SNA 5250 Pass-Through Considerations for Telnet Printer Pass-Through Mode

The AS/400 system supports 5250 pass-through. 5250 pass-through is similar to Telnet; it runs on an SNA (Systems Network Architecture) protocol network rather than a TCP/IP network. 5250 pass-through uses virtual displays to direct output to the physical devices just as Telnet does. In 5250 pass-through, the AS/400 system automatically creates virtual devices in the same way that it does for Telnet. Therefore, the QAUTOVRT system value controls the number of automatically configured virtual devices for both 5250 pass-through and Telnet. Other applications use virtual devices also, for example, printer pass-through, Workstation Gateway, and other IBM and non-IBM applications. For more information about 5250 pass-through, see the *Remote Work Station Support* book.

Step 3—Setting the Telnet Timemark Timeout Value

You should take into consideration the TIMMRKTIMO parameter.

The Telnet timemark timeout (TIMMRKTIMO) parameter specifies the number of seconds between TIMEMARK commands sent by the Telnet server. If Telnet is unable to send the TIMEMARK command, it closes the connection.

Step 4—Telnet Printer Pass-Through—Creating Virtual Controllers and Devices

You can create virtual controllers and devices. If you create your own virtual devices, by allowing the system to automatically select the device name, you must be aware of the following:

- The virtual controller must be named QPACTLnn, where nn is a decimal number 01 or greater.
- The virtual device should be named QPADEVxxxx, where xxxx is an alphanumeric character from 0001 to ZZZZ.

Note: Starting with Version 4 Release 2, the xxxx are no longer only numeric characters, but also alphanumeric characters from 0001 to ZZZZ, allowing a maximum of 1,679,615 unique names (devices).

If you want to use more than 32500 devices, which is the maximum value for the QAUTOVRT system, you can set the QAUTOVRT system value to *NOMAX to allow additional devices to be created.

- The Telnet server uses (existing) virtual devices only if the device type and model are the same as the device type and model negotiated between the client and server systems. Devices can also be selected via the exit program interface as opposed to being negotiated.

Step 5—Telnet Printer Pass-Through—Activating the QSYSWRK Subsystem

The QSYSWRK subsystem must be active. Use the Work with Subsystem (WRKSBS) command to display the status of the subsystem.

The Telnet server must also be started. The interactive subsystem, QINTER, which is used in previous examples in this chapter, needs to be started to run interactive jobs for Telnet sessions. The spooling subsystem (QSPL) needs to be active to run printer pass-through sessions.

Telnet Printer Pass-Through Mode Server to Client Access Win95 Telnet Client

This topic describes some practical experiences of a TPPT mode Telnet server with the Client Access for Win95 client.

TPPT Mode Server to Client Access Win95 Telnet Client

The IBM Client Access for Win95 client provides both display emulation, 5250 full-screen Telnet client, and printer emulation. A printer session can be started by selecting:

1. "IBM for AS/400 Client Access", "Accessories", "Start or Configure Session" from the program start menu
2. Select the name of an AS/400 system to connect to from the System name pull-down

Use the workstation ID field to specifically request an AS/400 virtual device name or leave it blank and the Telnet server will auto-select a compatible virtual device (QPADEVxxxx) and return the name on the printer control panel.

For type of emulation:

1. Choose printer
2. Click on the set-up box to bring up the PC5250 printer emulation set-up screen

From the set-up screen, you may configure things such as font, the AS/400 message queue where printer messages are to be sent, and the "transform print data to ASCII on AS/400" (HPT) host function. If HPT is selected, this enables other configuration items, such as printer model and media tray selection options. There is also an auto-reconnect option, and an option to override the default AS/400 Telnet port number (23).

The session is ended by selecting "communication" followed by "disconnect" from the menu bar.

Ending a Telnet Server Session

This material is covered in the **AS/400e Information Center** under the **TCP/IP** topic. For more information see "TCP/IP Topics in the Information Center" on page xv.

Starting Cascaded Telnet or DSPT Sessions

This material is covered in the **AS/400e Information Center** under the **TCP/IP** topic. For more information see "TCP/IP Topics in the Information Center" on page xv.

Using System Request Options

This material is covered in the **AS/400e Information Center** under the **TCP/IP** topic. For more information see "TCP/IP Topics in the Information Center" on page xv.

Telnet Scenarios for Establishing Cascaded Sessions

In Figure 126, you are establishing a Telnet session from Tokyo to Geneva after you have established a Telnet session with Chicago.

1. From the Tokyo system, type STRTCPTELN CHICAGO.
2. From the Chicago system, type STRTCPTELN GENEVA.



Figure 126. Establishing a Telnet Session with Three Systems

You now decide to return to the Tokyo system as shown in Figure 127 on page 221.

1. Press the System Request key.
2. Select option 13 (Start system request at home system). You are shown the System Request menu on the Tokyo system.

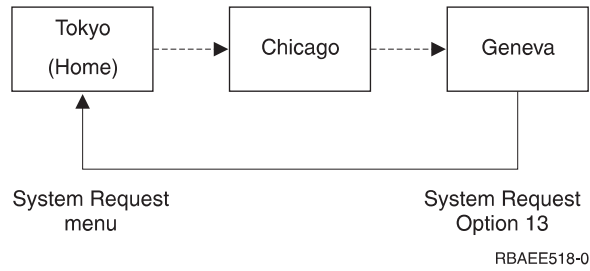


Figure 127. Returning to Home System

You can also return to the home system (Tokyo) using option 14 (Transfer to home system), shown in Figure 128.

1. Press the System Request key.
2. Select option 14 (Transfer to home system). You return to the alternate job on the Tokyo system.

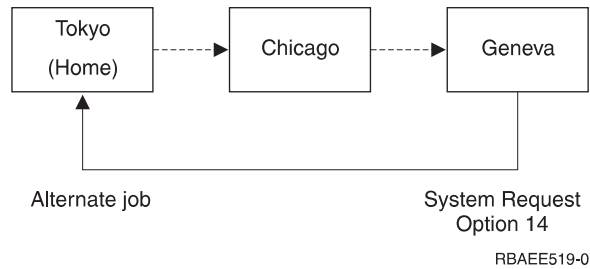


Figure 128. Returning to Alternate Job

Your Telnet sessions between Tokyo and Chicago and Chicago and Geneva are still active.

From the Geneva system you want to return to the Chicago system (Figure 129).

1. Press the System Request key.
2. Select option 10 (Start system request at previous system). You are shown the System Request menu on the Chicago system.

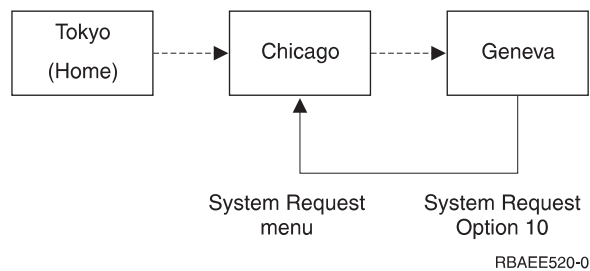


Figure 129. Starting System Request at Previous System

You can also return to the alternate job on the previous system (Chicago) using option 11 (Transfer to previous system), shown in Figure 130 on page 222.

1. Press the System Request key.

2. Select option 11 (Transfer to previous system). You return to the alternate job on the Chicago system.

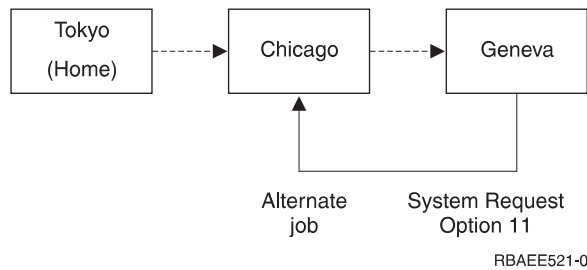


Figure 130. Transferring to Previous System

Now you want to end the sessions with Geneva and Chicago and return to the Tokyo system (Figure 131).

1. To end the session on the Chicago system from the Geneva system, type `SIGNOFF ENDCNN(*YES)`.
2. To end the session on the Tokyo system from the Chicago system, type `SIGNOFF ENDCNN(*YES)`. You return to the Tokyo system.

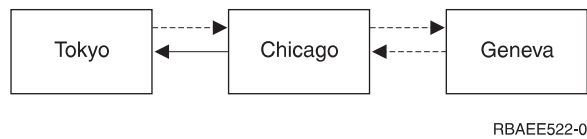


Figure 131. Ending a Session among Three Systems

In Figure 132, you are establishing a Telnet session from Tokyo to Chicago.

1. From the Tokyo system, type `STRTCPTELN CHICAGO`.

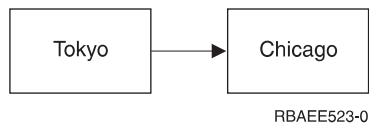


Figure 132. Establishing a Telnet Session between Two Systems

After establishing a Telnet session to Chicago, you want to return to the System Request menu at Tokyo (Figure 133 on page 223).

1. Press the System Request key.
2. Select option 10 (Start system request at previous system). You return to the System Request menu on the Tokyo system.

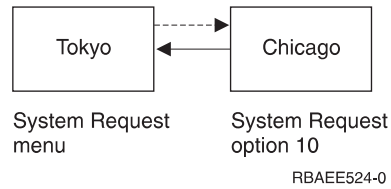


Figure 133. Starting the System Request at Previous System

You can also return to the Tokyo system using option 11 (Transfer to previous system). Option 11 sends you to the alternate job at the Tokyo system (Figure 134).

1. Press the System Request key.
2. Select option 11 (Transfer to previous system). You return to the Tokyo system.

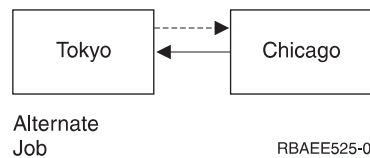


Figure 134. Transferring to the Previous System

From the alternate job at Tokyo, you now want to return to the Chicago system (Figure 135). You can return to the Chicago system using one of the following:

- SIGNOFF command
- TFRSECJOB command

You can use the Transfer Secondary Job (TFRSECJOB) command to transfer from an alternate job at a client system to the original job, which is running Telnet; therefore, you end up at the server system. In Figure 135, the TFRSECJOB command transfers you to your original job on the Tokyo system, which is still running Telnet; therefore, you end up on the Chicago system.

- Option 1 on the System Request menu (Display sign on for alternative job)

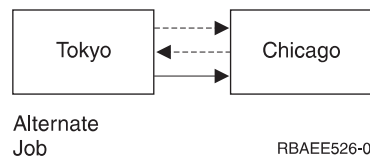


Figure 135. Transferring a Secondary Job

You want to return to the Chicago system after going back to the Tokyo System Request menu (Figure 136 on page 224). Do one of the following to return to the Chicago system:

- From the Tokyo System Request menu, press F3 (Exit).
- From the Tokyo System Request menu, select option 15 (Transfer to end system).

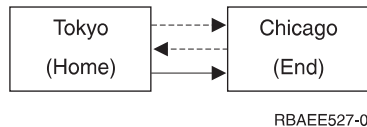


Figure 136. Returning to the End System

Now you want to end the session with Chicago and return to the Tokyo system (Figure 137).

1. From the Chicago system, type `SIGNOFF ENDCNN(*YES)`. You return to the Tokyo system.

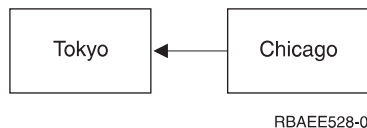


Figure 137. Ending a Session between Two Systems

Notes:

1. There is no limit to the number of systems to which you can establish a Telnet session.
2. You can mix Telnet and pass-through (DSPT) sessions. For example, you could go from Tokyo to Chicago using pass-through (DSPT) and from Chicago to Geneva using Telnet.
You can only use the `TFRPASTHR` command at a target system using the pass-through (DSPT) function. You cannot use it at a server system using Telnet. For information on using display station pass-through (DSPT), see the *Remote Work Station Support* book.
3. You can only have one pass-through (DSPT) or Telnet session per job. If you want to have multiple sessions from your home system, see “Using a Group Job—Scenario” on page 227.
4. The home system intercepts System Request options 13 and 14 if entered on the System Request input line. This function may be helpful if you establish a Telnet session with a system to which you cannot sign on. In this case, you can end the session to that system by doing the following:
 - a. Press the System Request key.
 - b. Type 13 (Start system request at home system) on the System Request input line.
 - c. Type 2 (End previous request) on the System Request menu.

All pass-through (DSPT) and Telnet sessions are stopped and you are returned to the home system.

5. If you use Telnet from a home system to another system, and want to establish a Telnet session with a third or an additional system, there is no requirement that the home system be at a certain release level. However, to pass through from an end system to another system, the end system must be running OS/400 Version 2 Release 3 or later. Similarly, to use System Request options 13, 14, and 15, the end system must be running OS/400 Version 2 Release 3 or later. And, finally, System Request options 10 and 11 work differently on systems prior to OS/400 Version 2 Release 3 (as described in the following topic).

System Request Processing—Scenarios

The system request processing for Telnet can differ from the system request processing for display station pass-through (DSPT) in all the following scenarios with the exception of the first scenario.

Scenario 1

All systems are AS/400 systems at Version 2 Release 3 (V2R3) or later. The system request processing for Telnet works the same as the system request processing for the pass-through (DSPT) support.

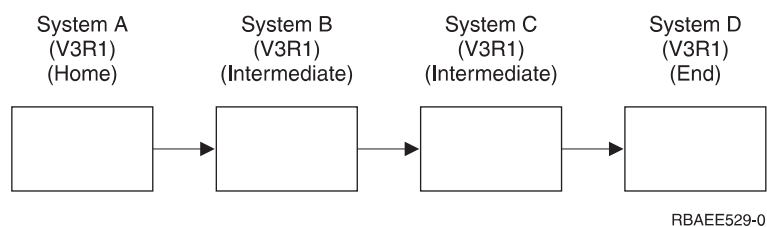


Figure 138. All AS/400 Systems at Version 2 Release 3 or later

Scenario 2

System A is an AS/400 system at Version 2 Release 2 (V2R2) or earlier.

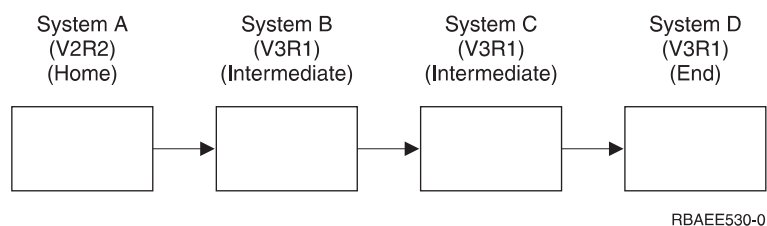


Figure 139. System A at Version 2 Release 2 or Earlier

The system request processing is the same as for the Version 2 Release 3 or later options in the first scenario with these exceptions:

- The System Request menu displayed on System A (the home system) uses option 11 instead of option 15 to transfer to the end system.
- If you type option 10 at the barred-input line at the bottom of the System Request menu, the System Request menu for System A is shown. If you type option 11, an alternate job is started on System A.

System request processing differs from the processing that occurs if you press the following keys: System Request, Enter, then type either option 10 or 11. In this case option 10 shows the System Request menu for System C, and option 11 starts an alternate job on System C.

Scenario 3

System D is an AS/400 system at Version 2 Release 2 or earlier.

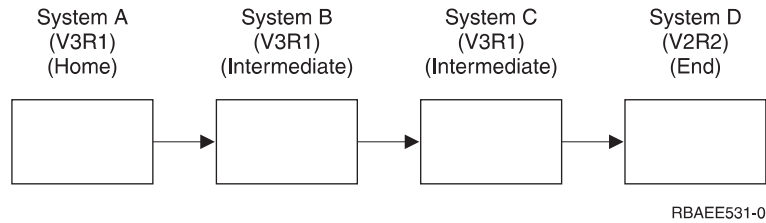


Figure 140. System D at Version 2 Release 2 or Earlier

The System Request menu for the end system will not have options 13 and 14. As a result, you can only use option 10 (or 11) to start a system request on (or transfer to) the previous system (System C). However, options 13 and 14 are available on the System Request menus of the intermediate systems.

Scenario 4

System A is a non-AS/400 system using 3270 or VTxxx Telnet.

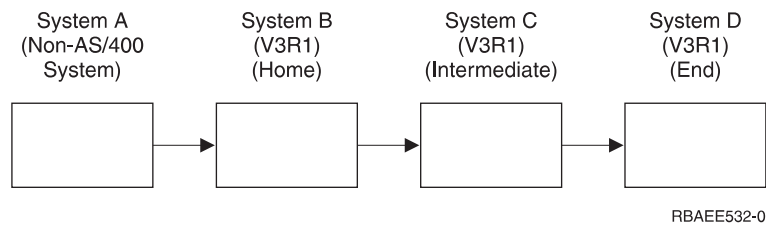


Figure 141. System A Is a Non-AS/400 System

The system request processing works like the first scenario except System B is considered the home system. All system requests sent to the home system are processed on System B.

Scenario 5

System D is a non-AS/400 system using 3270 or VTxxx Telnet.

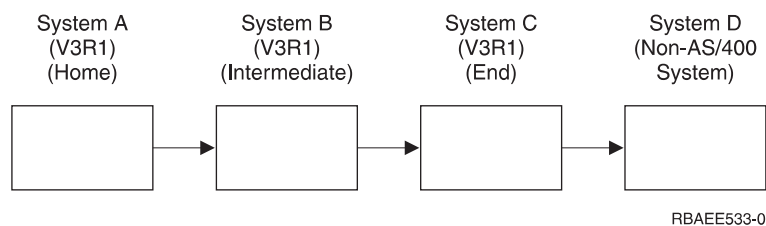


Figure 142. System D Is a Non-AS/400 System

The system request processing works like the first scenario except System C is considered to be the end system for all system request processing. If you press the System Request key, and then press the Enter key, the System Request menu for System C is shown.

Scenario 6

System C is a non-AS/400 system using 3270 or VTxxx Telnet.

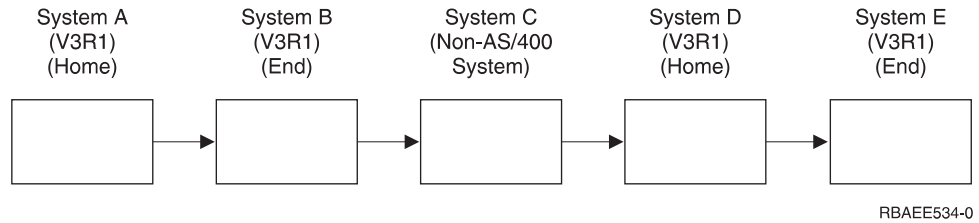


Figure 143. Non-AS/400 System Is an Intermediate System

The system request processing works like the first scenario except System B is considered the end system for system request processing. If you press the System Request key and then press the Enter key, the System Request menu for System B is shown.

If you want to send a system request to System E, you can map a function key on System D to the System Request key. If the mapping function is done, then System E is considered the end system, and System D is considered the home system.

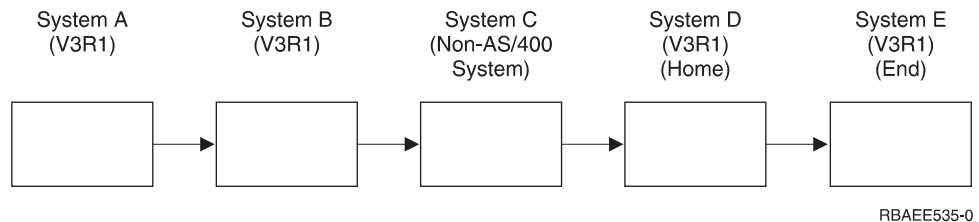


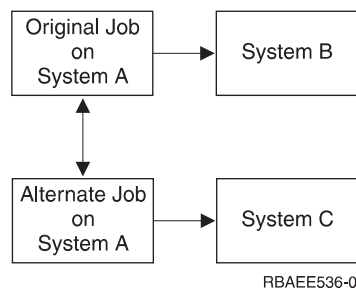
Figure 144. Non-AS/400 System Is an Intermediate System with Mapping Function

As an example of this mapping function for an AS/400 3270 Telnet server, the default keyboard mapping identifies the System Request key as a 3270 PF11 key. For an AS/400 3270 Telnet client, the F11 key is mapped to the 3270 PF11 key. If System C is a system using the 3270 data stream, pressing F11 maps to the System Request key on System D. The system request is sent to System E, and the System Request menu for System E is shown.

Note: This mapping function is complex especially if you are using the VTxxx data stream and are mapping between block data and character data.

Using a Group Job—Scenario

You can use Telnet and the alternate job to connect to multiple systems from your home system. Consider the following example:



Telnet is used to establish a session from System A to System B. You also want to go to System C and remain connected to System B. You can start an alternate job on System A using System Request option 11. Use the Telnet command to establish a session to System C. You can get to another system (System D, for example) by starting another Telnet session from System B or System C.

An alternative to using the alternate job is to use a **group job**. A group job is one of up to 16 interactive jobs that are associated in a group with the same workstation device and user. To set up a group job, do the following:

1. Change the current job to a group job using the Change Group Attributes (CHGGRPA) command.
 CHGGRPA GRPJOB(home)
2. Start a group job for System B using the Transfer to Group Job (TFRGRPJOB) command.
 TFRGRPJOB GRPJOB(SYSTEMB) INLGRPPGM(QCMD)
3. Establish a Telnet session to System B.
 Telnet SYSTEMB
4. Return to your home system by pressing the ATTN Key. Pressing the ATTN key shows you the Send Telnet Control Functions menu.
5. On the command line for the Send Telnet Control Functions menu, type:
 TFRGRPJOB GRPJOB(home)

This returns you to your original job.

Other group jobs and Telnet sessions can be started similarly.

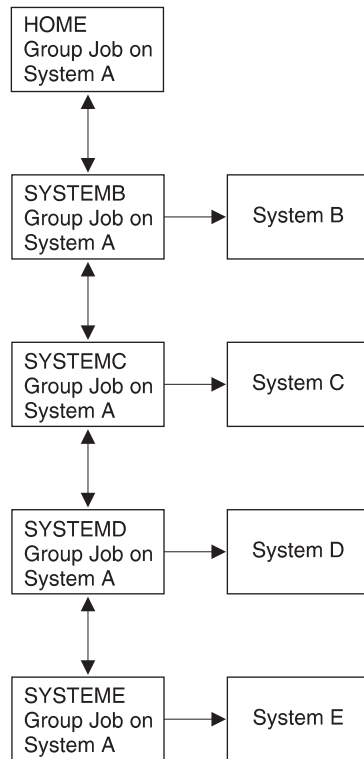
The TFRGRPJOB GRPJOB(*SELECT) command can be used to select which group job you want. For example, if group jobs are started with the names SYSTEMB, SYSTEMC, SYSTEMD, and SYSTEME, the TFRGRPJOB GRPJOB(*SELECT) command shows the following display:

```

                                Transfer to Group Job
                                System: SYS198
Active group job . . . . : HOME
Text . . . . . :
Type option, press Enter.
  1=Transfer to group job
-----Suspended Group Jobs-----
Opt   Group Job   Text
-     SYSTEME
-     SYSTEMD
-     SYSTEMC
-     SYSTEMB
Bottom F3=Exit  F5=Refresh  F6=Start a new group job  F12=Cancel

```

You can then use Telnet to establish a session with each system from the appropriate job. The group job scenario would look like this:



RBAEE537-0

When you want to end the group job, use the End Group Job (ENDGRPJOB) command.

When you are in a Telnet session, you can switch to another group job by pressing the ATTN key and then typing TFRGRPJOB on the command line.

Workstation Type Negotiations and Mappings

Table 19 shows a list of virtual display stations that the server system uses to match the physical display stations of the client system.

If you are not sure what emulation package you are running, you need to determine what your virtual display device is. You can use the Work with Job (WRKJOB) command to find out what it is. You are shown the job name at the top of the display. This is the name of the virtual display device associated with your job. By default, the naming convention is QPADEVxxxx, where xxxx is an alphanumeric character.

To determine the device type, type:

```
WRKCFGSTS *DEV QPADEVxxxx
```

You can work with your device description. Type an 8 (Work with description) next to the name of the device. You are shown the device type. You can then determine from the device type whether you are running in full-screen mode for 3270, 5250, VT100, or VT220.

Table 19. Full-Screen Workstation Mappings

Supported Workstation and (Model) ¹	Equivalent Type and (Model)	Internet Specification	Description
5251 (11)		IBM-5251-11	24 X 80 monochrome display
5291 (1)	5291 (2)	IBM-5291-1	24 X 80 monochrome display
5292 (2)		IBM-5292-2	24 X 80 color graphics display; this workstation type is also emulated by a graphics workstation function.
3196 (A1)	3196 (A2) 3196 (B1) 3196 (B2) 3476 (EA)	IBM-3196-A1	24 X 80 monochrome display; this workstation type is also emulated by a monochrome workstation function.
3486 (BA)		IBM-3486-BA	24 X 80 monochrome display
3487 (HA) ²	3487 (HG) ² 3487 (HW) ²	IBM-3487-HA	24 X 80 monochrome display; this workstation type is also emulated by a monochrome workstation function.
3487 (HC) ²		IBM-3487-HC	24 X 80 color display; this workstation type is also emulated by a color workstation function.
3179 (2)	3197 (C1) 3197 (C2) 3476 (EC) 5292 (1)	IBM-3179-2	24 X 80 color display; this workstation type is also emulated by a color workstation function.
3180 (2)	3197 (D1) 3197 (D2) 3197 (W1) 3197 (W2)	IBM-3180-2	27 X 132 monochrome display
5555 (B01)	5555 (E01)	IBM-5555-B01	24 X 80 double-byte character set (DBCS) monochrome display; this workstation type is emulated by a workstation function that supports DBCS display.
5555 (C01)	5555 (F01)	IBM-5555-C01	24 x 80 DBCS color display; this workstation type is emulated by a workstation function that supports DBCS display.
5555 (G01)		IBM-5555-G01	24 X 80 double-byte character set (DBCS) monochrome, graphics display; this workstation type is emulated by a workstation function that supports DBCS display.
5555 (G02)		IBM-5555-G02	24 x 80 DBCS color graphics display; this workstation type is emulated by a workstation function that supports DBCS display.
3477 (FC)		IBM-3477-FC	27 X 132 wide-screen color display
3477 (FG)	3477 (FA) 3477 (FD) 3477 (FW) 3477 (FE)	IBM-3477-FG	27 X 132 wide-screen monochrome display
3277 (0) ³	3277 (DHCF)	IBM-3277-2	24 X 80 monochrome display
3277 (0) ^{3, 4}	3278 (DHCF)	IBM-3278-2	24 X 80 monochrome display
3278 (0) ³		IBM-3278-2-E ⁵	24 x 80 monochrome display
3278 (0) ³		IBM-3278-3	24 x 80 monochrome display
3278 (0) ³		IBM-3278-4	24 x 80 monochrome display
3278 (0) ³		IBM-3278-5	24 x 80 monochrome display
3279 (0) ³	3279 (DHCF)	IBM-3279-2 IBM-3279-2-E ⁵	24 X 80 monochrome display
3279 (0) ³		IBM-3279-3	24 x 80 color display

Table 19. Full-Screen Workstation Mappings (continued)

Supported Workstation and (Model) ¹	Equivalent Type and (Model)	Internet Specification	Description
VT100 (*ASCII) ⁶		DEC-VT100 VT100 ⁷ VT102 DEC-VT102 DEC-VT200 DEC-VT220 VT200 ⁷ VT220 ⁷	24 x 80 monochrome ASCII display

Notes:

1. All 5250 workstations, except 5555 (B01) and 5555 (C01), can operate as 5251-11 workstations.
2. This workstation can be configured to be either 24 x 80 or 27 x 132. You must determine the mode of the workstation before setting the workstation type parameter value.
3. The AS/400 system supports only 24 X 80 screens in remote 327x workstations. Remote 3277 (both distributed host command facility (DHCF), and regular) workstations are mapped to IBM-3277-2. Remote 3278 workstations are mapped to IBM-3278-2. Remote 3279 workstations are mapped to IBM-3279-2.
4. Some TELNET 3270 full-screen (TN3270) or 3278-2 emulator packages do not support write structured fields correctly. Because of this, 3278-2 type devices are mapped to 3277-2 devices by the AS/400 TELNET server implementation to allow the AS/400 system to work with those TN3270 implementations.
5. The extended attributes highlighting is supported. Underline, blink, and reverse video are included. 3270 DBCS processing is also supported.
6. The VT100 virtual device supports VT220 devices.
7. VT100, VT200, and VT220 are not official terminal type names. However, some implementations negotiate using these names as the terminal type value.

System API Enhancement

The System API QDCRDEVD (Retrieve Device Description) provides the IP address of the Telnet client. There are new fields for display (*DSP) and print (*PRT) devices:

1. Network protocol
2. Network protocol address
3. IP internet address in dotted decimal form

These fields supply sockets level information to your application about the client's TCP/IP connection.

Dynamic Application Printing with TCP/IP

AS/400 TCP/IP supplies essential application print support to help control and direct printing over your TCP/IP networks. This includes dynamic print support for clients with variable Internet Protocol (IP) addresses. This support is supplied in the form of a system API enhancement that allows your applications to retrieve the IP address of your Telnet client.

The determination of a client's location through its IP address is an improvement over SNA- based pass-through connections which relied on the virtual device name. The AS/400 Telnet server stores the IP address of the client in the virtual device description.

IP Address Mapping

When you use the QDCRDEVD API, your application can map the IP address of the client to a particular printer. This allows your application to control where and how to send the print file. This mapping sends print files back to the client's workstation, to a network printer or to a printer on the application system.

Printer	Print IP	Client IP	User	DestType	Transform	Type/Model
hpjet	5.6.7.22	5.6.7.*	STEVENS	*OTHER	*YES	*HP560C
QPRINT	5.6.7.23	5.6.8.*	QUSER	OS400	*NO	IBM4029
LPTL	*CLIENT	*.*.*	MURPHY	*OTHER	*YES	IBM42011

IP Address Mapping Rules

1. A '*' in column 1 is a comment line, a '|' is a field delimiter.
2. A '*' character can be used as a wildcard in IP address hops.
3. *CLIENT for the printer IP field means substitute the client IP .
4. DestType must be: *SAME or a valid DESTTYPE parameter value.
5. Transform must be: *SAME or a valid TRANSFORM parameter value.
6. Type/Model must be: *SAME or a valid MFRTYPMDL parameter value.

Figure 145. IP Address Mapping

IP Address Mapping Scenarios

IP address mapping can support a variety of application controlled printing options as indicated in the three client scenarios that follow. Individual applications are not limited to these scenarios. It is up to you to decide what you need from your own application. Overall, the information included in this section provides you with the tools to restrict , track, or re-route print files as your business dictates.

Client 1: In this scenario, client 5.6.7.8 is at the office and is on subnet 5.6.7.*. All clients on subnet 5.6.7.* want printouts on a local network printer, hpjet at 5.6.7.22. This mapping program modifies the default *OUTQ for client 5.6.7.8 with the command:

```
CHGOUTQ OUTQ(default) RMTSYS('5.6.7.22') RMPRTQ('hpjet')
AUTOSTRWTR(1) CNNTYPE(*IP) TRANSFORM(*YES) MFRTYPMDL(*HP560C)
DESTTYPE(*OTHER)
```

Client 2: In this second scenario, clients on subnet 5.6.8.* handle confidential information and require a graphics printer that supports overlays (IFPDS type spooled files). All clients on subnet 5.6.8.* want printouts to secured printer QPRINT at 5.6.7.23. This mapping program modifies the default *OUTQ for this client with the command:

```
CHGOUTQ OUTQ(default) RMTSYS('5.6.7.23') RMPRTQ('QPRINT')
AUTOSTRWTR(1) CNNTYPE(*IP) TRANSFORM(*NO) MFRTYPMDL(*IBM4029)
DESTTYPE(*OS400)
```

Client 3: In this third scenario, client 1.2.3.4 is located at home and is using an Internet Service Provider (ISP). This person wants printouts sent to the LPD server running on their workstation. To do this, you could use all wildcards as a catchall entry (use it as the last one) for ISP clients, and have the IP printer address mapped to the same as the client workstation. This mapping program modifies the default *OUTQ for this client with the command:

```
CHGOUTQ OUTQ(default) RMTSYS('1.2.3.4') RMPRTQ('hpjet')
AUTOSTRWTR(1) CNNTYPE(*IP) TRANSFORM(*YES) MFRTYPMDL(*IBM42011)
DESTTYPE(*OTHER)
```

Firewalls: Firewalls and other tools can mask or manipulate IP addresses to appear other than the true Internet address. For that reason, the security of the client IP address is only as good as the security of your TCP/IP network. For information on firewall concepts and on an IBM firewall product for the AS/400, see the *Getting Started with IBM Firewall for AS/400*, SC41-5424-02 book or go to <http://www.as400.ibm.com/firewall>.

Exit Point Performance

The Telnet server response time for your initial session request will include any time that it takes for the QIBM_QTG_DEVINIT user exit program to be called, executed and returned. If your exit program is doing significant processing, the performance impact may result in a longer wait before your session is established.

Once the Telnet program is established by way of a sign-on panel or other AS/400 panel, there is no performance impact. When this occurs, the user exit program is no longer in the Telnet path. Established Telnet sessions experience no delays due to the QIBM_QTG_DEVINIT user exit program.

There is no user-visible performance impact that is associated with disconnecting the session. Disconnecting means that you end your terminal emulation session, not that you sign-off and return to the sign-on panel. If you disconnect, then the QIBM_QTG_DEVTERM user exit program is invoked, which will perform the termination processing for your session. Users will not see this because it occurs after the connection is broken.

Work Management

Key work management issues can be solved by using a Telnet exit program. These issues include the capability to request device descriptions other than QPADEVxxxx, opening up the door for work management control of interactive virtual terminal jobs and routing those jobs to specific subsystems.

Subsystem routing and device name selection

It is recommended that no more than 300 users be serviced by any given subsystem, for example, QBASE, QCMN, or QINTER. In the past, for Telnet, this posed problems because workstation entries did not work due to the QPADEVxxxx naming convention which meant that there was no way to subdivide the jobs easily.

Beginning with Version 4 Release 2, users can take advantage of better Telnet virtual device names and configure their interactive subsystems to subdivide the work, if necessary. This is done by using the *Add Work Station Entry* (ADDWSE) command to specify which devices a subsystem should or should not allocate a particular name of virtual terminal devices.

The following command has QINTER allocate all QPADEV* workstations, which means that all such devices are routed to the QINTER subsystem:

```
ADDWSE SBS(D(QINTER) WRKSTN(QPADEV*) AT(*SIGNON)
```

The following command has QINTER *not* allocating all QPADEV* workstations, which means that these devices can be allocated to a different subsystem:

```
ADDWSE SBS(DQINTER) WRKSTN(QPADEV*) AT(*ENTER)
```

Users can develop their own device naming conventions to subdivide the work. For example, one kind of subdivision is to route certain devices to national language support (NLS) related subsystems in two locations.

Example

For the purpose of this example, the two users are located in Endicott and Rochester. The users are assigned to AS/400 subsystems ENDICOTT and ROCHESTER, respectively, according to their geographic location. The characteristics of this example include:

- The IP addresses for Endicott start with 1.2.3.* .
- The IP addresses for Rochester start with 2.3.4.*.
- In order for all of the Endicott Telnet sessions to run in the ENDICOTT subsystem and the Rochester Telnet sessions to run in the ROCHESTER subsystem, the user exit program is employed to create a virtual device name that starts with 'ENDICOT' for all Telnet connections from 1.2.3. * and a virtual device name that starts with 'ROCHEST' for all connections from 2.3.4.*
- The user exit program assigns the virtual device name 'ENDICOT047' for an IP address of 1.2.3.47 and a virtual device name of 'ROCHEST048' for an IP address from 2.3.4.48 (using the last three digits of the IP address to complete a unique name for every user).

To insure that virtual devices ENDICOT047 and ROCHEST048 go into subsystems Endicott and Rochester, respectively, the workstations entries are set up as follows:

```
ADDWSE SBS(DQINTER) WRKSTN(ENDICOT*) AT(*ENTER)
ADDWSE SBS(DQINTER) WRKSTN(ROCHEST*) AT(*ENTER)
```

```
ADDWSE SBS(DENDICOTT) WRKSTN(ENDICOT*) AT(*SIGNON)
ADDWSE SBS(DROCHESTER) WRKSTN(ROCHEST*) AT(*SIGNON)
```

Chapter 7. File Transfer Protocol (FTP) Client

The File Transfer Protocol (FTP) client function allows you to send and receive copies of files to or from remote systems across a TCP/IP network. In addition, FTP client subcommands are provided for renaming, appending to, and deleting files.

The majority of File Transfer Protocol (FTP) Client documentation material is covered in the **AS/400e Information Center** under the **TCP/IP** topic. For more information see "TCP/IP Topics in the Information Center" on page xv.

Functions Supported by FTP Client

- Transferring files that are found in the Root, QOpenSys, QLANSrv file, and QFileSvr.400 systems.
- Running FTP unattended in batch.
- The automatic creation of sequence numbers when transferring data to a source physical file on AS/400 business computer systems.
- Sending or receiving save files and members in physical files, logical files, Distributed Data Management (DDM) files, and source physical files.
- Sending text files in EBCDIC format or converting them to ASCII (the default format).
- Transferring binary files as is.
- Transferring folders and document in the document library services (QDLS) file system.
- Double-byte character set (DBCS) support.
- Coded character set identifier (CCSID) support.

Functions Not Supported by FTP Client

- Transferring selected records within files.
- Selection or omission of fields within records when transferring files. You must transfer an entire physical or logical record.

FTP Client and Server-Overview

FTP consists of two parts, the client and the server as shown in Figure 146 on page 236.

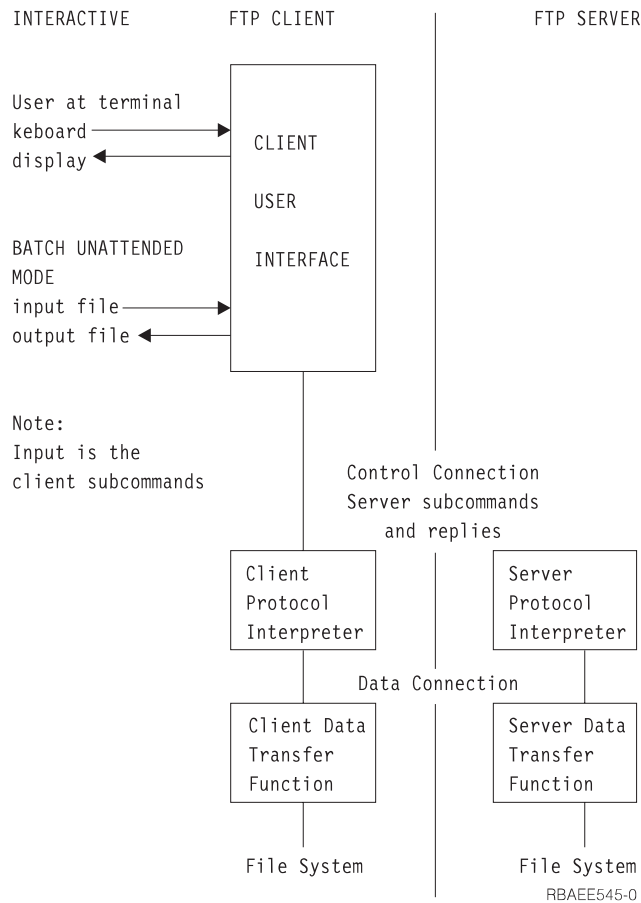


Figure 146. Relationship between FTP Client and FTP Server

The client has a user interface from which you can enter client subcommands for making requests to an FTP server. The results of these requests are then displayed.

One may interact with the client online or run the FTP client in an unattended batch mode where client subcommands are read from a file and the responses to these subcommands are written to a file.

To transfer files between the client and the server, two connections are established. The control connection is used to request services from the server with FTP server subcommands. The server sends replies back to the client to indicate how the request was handled. The second connection, called the data connection, is used for transferring lists of files and the actual file data.

Both the client and the server have a data transfer function that interfaces to the resident file systems. These functions read or write data to the local file systems and to and from the data connection.

Requests for transferring files originate from the client. The user makes these requests with the FTP client subcommands that are read by the client user interface function. The client subcommands are interpreted by the client protocol interpreter, which translates them into appropriate FTP server subcommands. The server protocol interpreter receives the subcommands from the control connection and processes it. The results of each server subcommand are transmitted back to the client in the form of an FTP server reply.

The distinction between FTP client and FTP server is from the viewpoint of where the FTP commands are initiated, and not from the viewpoint of where the data resides. Thus, the commands are initiated from the FTP client session, while the files being transferred may initially reside on either system.

Starting the FTP Client Session

This material is covered in the **AS/400e Information Center** under the **TCP/IP** topic. For more information see "TCP/IP Topics in the Information Center" on page xv.

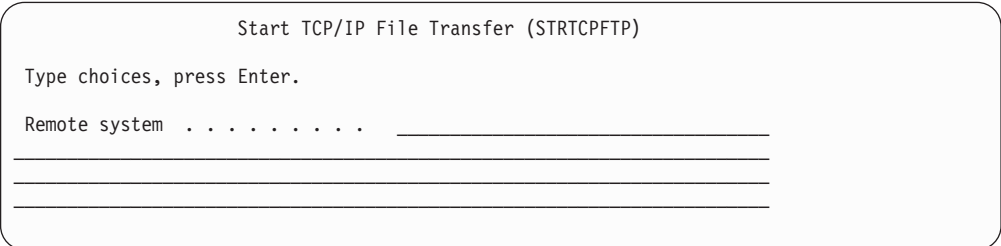
Alternative Start Commands

There are several ways to start an FTP client session:

- Use the Start TCP/IP File Transfer Protocol (STRTCPFTP) command (the instructions for using this command follow).
- The AS/400 system allows you to use the FTP command. This command performs the same functions as the STRTCPFTP command.
- Select option 9 (Start TCP/IP FTP session) from the TCP/IP Administration menu. To get to the TCP/IP Administration menu, type GO TCPADM on the command line of the AS/400 Main menu.

To start an FTP client session using the STRTCPFTP command:

1. Type STRTCPFTP on the command line.
2. Press F4 (Prompt). The prompt for the remote system is displayed as shown in Figure 147.



```
Start TCP/IP File Transfer (STRTCPFTP)

Type choices, press Enter.

Remote system . . . . . _____
_____
_____
_____
```

Figure 147. Prompt for the Remote System

3. Type in the name or internet address of the remote system with which you want to start your FTP session.
4. Press the Enter key.

A prompt for the coded character set identifier (CCSID) is displayed as shown in Figure 148 on page 238.

The CCSID parameter is displayed with the value *DFT. This parameter specifies the ASCII CCSID to be used for single-byte character set (SBCS) ASCII file transfers when the FTP TYPE mode is set to ASCII. When *DFT is specified, the default CCSID value is 819 (ISO 8859-1 8-bit ASCII).

For some AS/400 systems, character conversion between the EBCDIC CCSID specified by your job and the default CCSID 819 may not be available. When this is

the case, the STRTCPFTP command displays message TCP3C14: Unable to convert data from CCSID &1 to CCSID &2. The following sample message shows that &1 is the job CCSID and &2 is the ASCII CCSID specified in the STRTCPFTP command.

```
UNABLE TO CONVERT DATA FROM CCSID 5026 TO CCSID 819
```

In this situation, the STRTCPFTP command must be run again with an ASCII CCSID for which character conversion is supported. CCSID 850, which contains the PC Latin-1 coded character set and supports character conversion for all valid job CCSID values, is suggested.

```
Start TCP/IP File Transfer (STRTCPFTP)

Type choices, press Enter.

Remote system . . . . . > SYSNAM08.ENDICOTT.IBM.COM
_____
_____
_____
"Internet address . . . . ." _____
Coded character set identifier *DFT 1-65533, *DFT
```

Figure 148. Prompt for the Internet Address

After typing a remote system name or an internet address and pressing the Enter key, an FTP client session is started and the FTP client session user interface display is shown (Figure 149 on page 239). In the history area (Previous FTP subcommands and messages) are displayed messages indicating that a connection to the server host has been initiated and replies from the server acknowledging that a connection has been established. Also, messages that indicate the inactivity time-out value and the remote host operating system may be included.

If the only message that appears is the first one, 'Connecting...', it is possible that TCP/IP or the FTP server has not been started on the remote host. Also, it is possible that the network connection is not available. This can be verified using the PING command.

```
File Transfer Protocol

Previous FTP subcommands and messages:
Connecting to host name SYSNAM08 at address 9.125.87.146 using port 21.
220-QTCP at sysnam08.endicott.ibm.com.
220 Connection will close if idle more than 5 minutes.

Enter login ID (itso):
===>

F3=Exit      F6=Print      F9=Retrieve
F17=Top      F18=Bottom    F21=CL command line
```

Figure 149. FTP Session Acknowledgement and Logon Prompt to the Remote System

Logon to the Remote System (Server)

When the connection to the server is established, the client automatically initiates a USER subcommand and prompts you for a logon ID. The logon prompt shows your client AS/400 system logon ID (itso in Figure 149). If your ID on the remote server is the same, press the Enter key. This ID is used by the AS/400 FTP client to log you on to the remote server. If you do not have this ID on the remote server, you should type your logon ID on the remote FTP server at the logon prompt before pressing Enter. Depending on the security at the remote server, it may request a password to complete the logon.

If your logon attempt fails because you did not type your user ID or password correctly, you can try to sign on again by doing the following:

1. Type the USER subcommand followed by your user ID.
2. Press the Enter key.
3. Type your password (if requested)
4. Press the Enter key.

```
File Transfer Protocol

Previous FTP subcommands and messages:
Connecting to host name SYSNAM08 at address 9.125.87.146 using port 21.
220-QTCP at sysnam08.endicott.ibm.com.
220 Connection will close if idle more than 5 minutes.

>
331 Enter password.
230 ITSO logged on.
OS/400 is the remote operating system. The TCP/IP version is "V3R1M0".
250 Now using naming format "0".
257 "QGPL" is current library.

Enter password:
===>

F3=Exit      F6=Print      F9=Retrieve
F17=Top      F18=Bottom    F21=CL command line
```

Figure 150. Successful Logon to FTP Session

As is shown in Figure 150, the FTP server sends replies that are preceded by a three-digit numeric code to the client host as a response to each subcommand. For example, a 331 response may be sent to request a password when the user ID is entered, and a 230 response is sent after a successful logon.

If the server is an AS/400, the server replies show the current library (directory) and the file name format in use on the server.

When the logon process has completed, you can enter the FTP subcommands that are described in this chapter.

Connecting to Another Server without Ending the FTP Session

The CLOSE subcommand can also be used to close the connection with the remote server. This subcommand does not end the FTP client session. The OPEN subcommand can then establish a connection with another server. For example, the following opens a connection to SYSNAM01:

```
OPEN SYSNAM01
```

When the OPEN is successful, the FTP client prompts for a user ID, as it does at FTP startup.

Ending the FTP Client Session

The FTP session is ended with the QUIT subcommand. The QUIT subcommand closes the connection with the remote host and ends the FTP session on the AS/400 system. Alternatively, you can press F3 (Exit) and then confirm to end the FTP client session.

Transferring Files with File Transfer Protocol (FTP)

This material is covered in the **AS/400e Information Center** under the **TCP/IP** topic. For more information see “TCP/IP Topics in the Information Center” on page xv.

Naming Format Indicator for AS/400 Names

A naming indicator, called NAMEFMT, enables the user to specify library file system names in either the format used by FTP prior to V3R1 or in a path name format as used by the integrated file system. The user selects the desired format by assigning either a one or zero to the NAMEFMT value.

- 0** indicates that the name is in the original format that is supported by FTP for the library file system prior to V3R1. Only library file system names can be entered when NAMEFMT is 0.
- 1** indicates that a path name format is being used for the file name. This format applies to all AS/400 file systems supported by FTP, including the library file system (QSYS.LIB). This name format complies with the Integrated File System naming rules.

The generic format for NAMEFMT=1 is:

```
[/filesystemname/]{directory/}filename
```

Notes:

1. On AS/400 FTP servers prior to V3R1, the NAMEFMT is assumed to be '0'. For V3R1 and subsequent releases, AS/400 FTP servers may have a NAMEFMT of '0' or '1'. For non-AS/400 servers, the NAMEFMT value is considered as unknown. The value of the server NAMEFMT value affects the default file names created by the FTP client.
2. A fully-qualified name in NAMEFMT=1 begins with a slash followed by the name of the file system. For example:

```
/QDLS/FOLDER/DOCUMENT
```

3. For NAMEFMT = 0, a fully-qualified name begins with the library name. For example:

```
LIBNAME/FILENAME.MBRNAME
```

4. For NAMEFMT = 1, a partially-qualified name does not begin with a slash and does not include the name of the file system. For example:

```
FOLDER/DOCUMENT
```

(where the working directory is /QDLS)

```
LIBNAME.LIB/FILEXYZ.FILE/MEMBER1.MBR
```

(where the working directory is /QSYS.LIB)

5. For NAMEFMT = 0, a partially-qualified name begins with the file name without any slash. For example:

```
FILENAME.MBRNAME
```

(where the working directory is LIBNAME)

6. The NAMEFMT is set to “0” when an FTP session is started.

File Naming for the Library File System (QSYS.LIB)

This material is covered in the **AS/400e Information Center** under the **TCP/IP** topic. For more information see “TCP/IP Topics in the Information Center” on page xv.

Names for Document Library Services (QDLS) Folders and Documents

This material is covered in the **AS/400e Information Center** under the **TCP/IP** topic. For more information see “TCP/IP Topics in the Information Center” on page xv.

Names for “root,” QOpenSys, QLANSrv and QFileSvr.400 File Systems

This material is covered in the **AS/400e Information Center** under the **TCP/IP** topic. For more information see “TCP/IP Topics in the Information Center” on page xv.

Localfile and Remotefile Parameters for FTP Client Subcommands

This material is covered in the **AS/400e Information Center** under the **TCP/IP** topic. For more information see “TCP/IP Topics in the Information Center” on page xv.

Default File Names for Client Transfer Subcommands

The majority of this material is covered in the **AS/400e Information Center** under the **TCP/IP** topic. For more information see “TCP/IP Topics in the Information Center” on page xv.

Table 20 shows an example of default names.

Notes:

1. Save files do not have members. Therefore, default names for save files do not have a member part.
2. Default names are displayed when the DEBUG mode is turned on.

Table 20. Example Default Names for the PUT, APPEND, and MPUT Subcommands

Source Host (AS/400 Client)			Target Host (Server)		
Name Format	File Sys	File Name	Name Format	File Sys	Default Name
0	QSYS.LIB	libnam/ffff.mmmm	0	QSYS.LIB	ffff.mmmm
0	QSYS.LIB	libname/ffff	0	QSYS.LIB	ffff.ffff ¹
1	QSYS.LIB	libX.lib/ffff.file/mmmm.mbr	0	QSYS.LIB	ffff.mmmm ²
0	QSYS.LIB	libname/ffff.mmmm	— ³	— ³	ffff.mmmm
0	QSYS.LIB	libname/ffff	— ³	— ³	ffff
1	QSYS.LIB	libX.lib/ffff.file/mmmm.mbr	— ³	— ³	ffff.mmmm ²
0	QSYS.LIB	libx/ffff	0	QSYS.LIB	ffff ⁴

Table 20. Example Default Names for the PUT, APPEND, and MPUT Subcommands (continued)

Source Host (AS/400 Client)			Target Host (Server)		
Name Format	File Sys	File Name	Name Format	File Sys	Default Name
1	QSYS.LIB	libx.lib/ffff.savf	0	QSYS.LIB	ffff ⁴
1	QSYS.LIB	libx.lib/ffff.file	0	QSYS.LIB	ffff ⁴
1	QSYS.LIB	libx.lib/ffff.savf	1	QSYS.LIB	ffff.SAVF ⁴
1	QSYS.LIB	libx.lib/ffff.file	1	QSYS.LIB	ffff.FILE ⁴
0	QSYS.LIB	libx/ffff	1	QSYS.LIB	ffff.SAVF ⁴
1	QDLS	libname/filename.ext	1	QDLS	filename.ext
1	QDLS	libname/filename	1	QDLS	filename
1	"root"	/Directory1/FileA	1	QOpenSys	FileA ⁵
1	QOpenSys	/QOpenSys/Direct/fileABC	1	"root"	fileABC ⁵

Notes:

- Shows that the default member name is the same as the file name when no member name is specified. The PUT subcommand attempts to send a member of the same name as the file name.
- Occurs when the server is an AS/400 that does not support NAMEFMT 1.
- Indicates unknown. The name format is not defined for non-AS/400 servers and is considered unknown.
- Illustrates how a save file is processed. A save file default name does not have a member name because save files do not have members.
- The case of the characters in the default name is the same as that of the entered file name for the "root," QOpenSys, and QLANSrv file systems.

Table 21. Example Default Names for the GET and MGET Subcommands

Source Host (Server)			Target Host (AS/400 Client)		
Name Format	File Sys	File Name	Name Format	File Sys	Default Name
0	QSYS.LIB	libname/ffff.mmmm	0	QSYS.LIB	FFFF.MMMM
0	QSYS.LIB	libname/ffff	0	QSYS.LIB	FFFF.FFFF
0	QSYS.LIB	libname/ffff	1	QSYS.LIB	FFFF.FILE/FFFF.MBR
0	QSYS.LIB	libname/ffff.mmmm	1	QSYS.LIB	FFFF.FILE/MMMM.MBR
1	QSYS.LIB	libX.lib/ffff.file/mmmm.mbr	0	QSYS.LIB	FFFF.MMMM
1	QSYS.LIB	libX.lib/ffff.file/mmmm.mbr	1	QSYS.LIB	FFFF.FILE/MMMM.MBR
— ¹	— ¹	abcdefghijk	0	QSYS.LIB	ABCDEFGH.ABCDEFG ²
— ¹	— ¹	abcdefghijk.lmnopgrst	0	QSYS.LIB	ABCDEFGH.LMNOPQR ²
— ¹	— ¹	abcd	1	QSYS.LIB	ABCD.FILE/ABCD.MBR ²
— ¹	— ¹	abcd.efgh	1	QSYS.LIB	ABCD.FILE/EFGH.MBR ²
— ¹	— ¹	abcdefghijk	1	QDLS	ABCDEFGH ²
— ¹	— ¹	abcdefghijkl.mnopqrs	1	QDLS	ABCDEFGH.MNO ²
1	QOpenSys	/QOpenSys/Directory/FileA	1	"root"	FileA ³
1	"root"	/Directory/fileABC	1	QOpenSys	fileABC ³

Table 21. Example Default Names for the GET and MGET Subcommands (continued)

Source Host (Server)			Target Host (AS/400 Client)		
Name Format	File Sys	File Name	Name Format	File Sys	Default Name
Notes:					
1. Server name format and system name is not defined. This is the typical case for non-AS/400 servers.					
2. Illustrate how names are formed for a GET when the server is not an AS/400.					
3. The case of the characters in the default name are the same as that of the entered file name for the "root," QOpenSys and QLANSrv file systems.					

FTP Client Subcommands

This material is covered in the **AS/400e Information Center** under the **TCP/IP** topic. For more information see "TCP/IP Topics in the Information Center" on page xv.

FTP Examples

This topic shows four specific scenarios of using FTP to put and get files.

- AS/400 to AS/400
- AS/400 to VAX**/Wollongong
- AS/400 to AIX (UNIX)
- AS/400 to OS/2 using QDLS file names

The process of running FTP to other hosts is the same as running FTP to another AS/400. However, there are differences in file naming structures, directory structures, and the operation of some of the FTP subcommands on those hosts. These differences may vary the results for some FTP subcommands. For example, most other systems do not have file structures that have file members. Therefore, when file members are transferred from an AS/400 to other systems, each AS/400 file member creates or replaces a separate file on the other system.

This section describes how to transfer data to and from different remote system servers using the PUT, MPUT, GET, and MGET subcommands. The AS/400 objects that can be transferred with these subcommands can be one of the following:

- Members of physical files (source and data)
- Logical files (put and get operations of a logical file create a physical file according to the view defined in the logical file)
- Save files
- Objects in Hierarchical File Systems (HFS) such as those in QDLS folders and documents.

The following examples illustrate how to put AS/400 database file members and get remote files into database file members.

Note: The restrictions and considerations noted in the following topics were discovered from tests on the particular systems. The results may be different for other software releases or implementations.

AS/400-to-AS/400

This example shows how to transfer physical file members to and from another AS/400.

Put a File: The FTP subcommand PUT is used to copy a local file member into a file at the remote host.

To copy a local file member into a file at the remote host, you need write authority for the library where the file is put. A library at the AS/400 system is a logical placeholder for other objects such as programs, files, and commands. A library can be compared to a directory on other systems.

The syntax of the PUT subcommand is as follows:

```
PUT localfile [remotefile]
```

The remotefile is optional. If omitted, the current library (or current directory for QDLS objects) on the remote AS/400 is used with the file having the same name. You can change the current library or directory on the remote AS/400 with the CD subcommand.

For the AS/400 system, the syntax of a file name, using NAMEFMT 0 is:

```
library/file.member
```

For the AS/400 system, the syntax of a file name, using NAMEFMT 1 is:

```
/QSYS.LIB/libname.LIB/filename.FILE/mbrname.MBR
```

This refers to the same file as the NAMEFMT 0 example above.

On the AS/400 system, files may have one or more members. Each file member can consist of data records, or each may contain other records such as source programs or database definitions.

An example of the PUT subcommand is shown in Figure 151, below.

```
File Transfer Protocol

> put ITSOTST1/scrncpy.scrncpy itsotst2/scrncpy.scrncpy
200 PORT subcommand request successful.
150 Sending file to member SCRNCPY in file SCRNCPY in library ITSOTST2.
250 File transfer completed successfully.
286412 bytes transferred in 12.595 seconds. Transfer rate 22.741 KB/sec.
Enter an FTP subcommand.
==> put ITSOTST1/scrncpy.scrncpy itsotst2/scrncpy.scrncpy

F3=Exit      F6=Print      F9=Retrieve
F17=Top      F18=Bottom    F21=CL command line
```

Figure 151. FTP PUT Display

The PUT subcommand copies the file member SCRNCPY in file SCRNCPY in library ITSOTST1 at the local host to member SCRNCPY in file SCRNCPY in library ITSOTST2 on the remote system. If the member already exists at the remote host, the remote system will overwrite the existing member.

If you want to transfer several files to the remote host, the MPUT subcommand may be used to do this:

```
mput libm01/filem01.*
```

The subcommand MPUT sends multiple file members to the remote host. The MPUT subcommand copies all members in the file FILEM01 in library LIBM01 to file FILEM01 in the current library of the remote host. The local file name is required, but the library name may be omitted. If the library name is not specified, the local user's current library is used. If the member name is not specified, the file name is used as the member name

Note that it is not possible to specify the library name at the remote host on the MPUT command itself. The remote user's current library is used. This is set automatically to the one specified in the user's profile at sign-on time. The PWD subcommand may be used to determine the current library. The CD subcommand may be used to change it, as shown in Figure 152, prior to issuing the MPUT command.

```
File Transfer Protocol

Previous FTP subcommands and messages:

> PWD
257 "QGPL" is current library.
> CD ITSOLIB1
250 Current library changed to ITSOLIB1.
Enter an FTP subcommand.
===>

F3=Exit      F6=Print      F9=Retrieve
F17=Top      F18=Bottom    F21=CL command line
```

Figure 152. FTP Determine and Change Server's Current Library

Note: On other systems, the current library may be called the current directory.

If an existing remote file and member name are specified, the old member may be replaced or a new member may be written depending on the setting of the SUNIQUE option. When FTP is started, this option is set off. The option can be toggled on and off by the SUNIQUE subcommand as shown in Figure 153 on page 247.

When SUNIQUE is off, the old member is replaced. When SUNIQUE is on, a new file member is created. The name of the new member is the same as was specified, but the name has a sequence number that is appended to the end.

Note: The SUNIQUE option setting controls what server subcommand is used when a PUT or MPUT client subcommand is entered. The subcommands are sent from the client host to the server host and are run on the server. The server subcommands are STOR and STOU:

- The STOR subcommand causes the remote file (member) to be overwritten if the file already exists. STOR is sent to the server by the client when the subcommands PUT or MPUT are run and SUNIQUE is set off.

- The STOU subcommand causes a new file (member) to be created on the remote host if the specified one already exists. The name of the new member is the specified one with a sequence number that is appended to it. For example, if MBR were specified as the member name, MBR1 would be created. The name of the new member is returned to the user. The STOU subcommand is sent to the server when the SUNIQUE is set on.

```

File Transfer Protocol

Previous FTP subcommands and messages:

> SUNIQUE
  Store unique is on.
> SUNIQUE
  Store unique is off.
Enter an FTP subcommand.
==> SUNIQUE

F3=Exit      F6=Print      F9=Retrieve
F17=Top      F18=Bottom    F21=CL command line

```

Figure 153. FTP Setting S**UNIQUE** On and Off

Get Several File Members: The GET subcommand is used to copy one file member from a remote host into a file at the local host. To do this, read authority is required for the remote library from which the file is copied. The syntax of the GET subcommand is as follows:

```
GET remotefile [localfile] [(Replace)]
```

It is conceptually very similar to the PUT command in reverse.

The MGET subcommand is used to copy multiple file members or multiple files from a remote host. As with the GET subcommand, read authority at the remote host is required.

An example of the MGET command is shown in Figure 155 on page 248.

```

File Transfer Protocol

Previous FTP subcommands and messages:
> mget ITSOLIB1/qclsrc.*
200 PORT subcommand request successful.
125 List started.
250 List completed successfully.
250 Now using naming format "0".
257 "QGPL" is current library.
200 PORT subcommand request successful.
150 Retrieving member MEMB1 in file QCLSRC in library ITSOLIB1.
250 File transfer completed successfully.
13 bytes transferred in 1.297 seconds. Transfer rate 0.010 KB/sec.
200 PORT subcommand request successful.
150 Retrieving member MEMB2 in file QCLSRC in library ITSOLIB1.
250 File transfer completed successfully.
Enter an FTP subcommand.
===> mget ITSOLIB1/qclsrc.*

F3=Exit      F6=Print      F9=Retrieve
F17=Top      F18=Bottom    F21=CL command line

```

Figure 154. MGET Subcommand—Display 1

```

File Transfer Protocol

Previous FTP subcommands and messages:
257 "QGPL" is current library.
200 PORT subcommand request successful.
150 Retrieving member MEMB1 in file QCLSRC in library ITSOLIB1.
250 File transfer completed successfully.
13 bytes transferred in 1.297 seconds. Transfer rate 0.010 KB/sec.
200 PORT subcommand request successful.
150 Retrieving member MEMB2 in file QCLSRC in library ITSOLIB1.
250 File transfer completed successfully.
49 bytes transferred in 1.304 seconds. Transfer rate 0.038 KB/sec.
200 PORT subcommand request successful.
150 Retrieving member MEMB3 in file QCLSRC in library ITSOLIB1.
250 File transfer completed successfully.
49 bytes transferred in 1.239 seconds. Transfer rate 0.040 KB/sec.
Enter an FTP subcommand.
===>

F3=Exit      F6=Print      F9=Retrieve
F17=Top      F18=Bottom    F21=CL command line

```

Figure 155. MGET Subcommand—Display 2

The MGET command copies all the file members (MEMB1, MEMB2, MEMB3) in file QCLSRC in library ITSOLIB1 at the remote host. These file members are copied to identically named members in file QCLSRC in the current library at the local host. Figure 154 and Figure 155 show an example of an MGET display session.

The remote file name is required, but the library name may be omitted. If the library name is not specified, the remote user's current library is used.

Note that it is not possible to specify the library name at the local host on the MGET subcommand itself. The local user's working directory is used. This is set automatically at sign-on time. You can use the LCD subcommand to change it for the duration of the FTP Session. See Figure 156.

Note: The FTP Client Local Working Directory and the current library on the FTP client system may be different libraries. The LCD subcommand does not change the current library, but the Change Current Library (CHGCURLIB) CL command changes both the working directory and the current library.

For example, examine the results of this series of commands:

```
SYSCMD CHGCURLIB XYZLIB
```

After this command is issued, the working directory and the current library are both XYZLIB.

```
LCD GHSLIB
```

After this command is issued, the working directory is changed to GHSLIB, but the current library remains as XYZLIB.

```
SYSCMD CHGCURLIB ABCLIB
```

After this command is issued, both the working directory and the current library are changed to ABCLIB.

Also note that the CD subcommand changes the current library on the server system when the server is an AS/400.

```
File Transfer Protocol

Previous FTP subcommands and messages:
Connecting to host name SYSNAM02 at address 9.4.73.250 using port 21.
220-QTCP at SYSNAM02.
220 Connection will close if idle more than 5 minutes.
>
331 Enter password.
230 GWIL logged on.
OS/400 is the remote operating system. The TCP/IP version is "V4R2M0".
250 Now using naming format "0".
257 "QGPL" is current library.
> lcd gerrylib
Local working directory is GERRYLIB.

Enter an FTP subcommand.
===>

F3=Exit      F6=Print      F9=Retrieve
F17=Top      F18=Bottom    F21=CL command line
```

Figure 156. LCD Subcommand

If the local member name exists and REPLACE is specified, the old member is replaced with the new one. If REPLACE is not specified, an error message is displayed and the MGET operation does not run for any members that already exist.

If errors occur during a file transfer to the AS/400, look in the job logs of the remote AS/400 FTP servers for more information.

Get Several Files from a Folder: Figure 157 and Figure 158 on page 251 show the commands involved in doing an FTP MGET subcommand using NAMEFMT 1 to transfer files between folders. In the example, the connection has already been established between the two AS/400 systems. The example does the following:

- Changes the name format to 1 (NAMEFMT 1).
- Changes the transfer type to IMAGE with the BINARY subcommand.
- Changes the directory on the server to folder WS3FLR with the CD subcommand.
- Changes the directory on the client to folder GWIL with the LCD subcommand.
- Uses the MGET *.* subcommand to transfer all files in the specified server system folder (WS3FLR) to the specified client system folder (GWIL).

```
File Transfer Protocol

Previous FTP subcommands and messages:
> NAMEFMT 1
  250 Now using naming format "1".
  Server NAMEFMT is 1.
  Client NAMEFMT is 1.
> BINARY
  200 Representation type is IMAGE.
> CD /QDLS/WS3FLR
  250 Current directory changed to /QDLS/WS3FLR.
> LCD /QDLS/GWIL
  Local working directory is /QDLS/GWIL.
> MGET *.* (REPLACE
Enter an FTP subcommand.
==> MGET *.* (REPLACE

F3=Exit      F6=Print      F9=Retrieve
F17=Top      F18=Bottom   F21=CL command line
```

Figure 157. FTP MGET Subcommand using NAMEFMT 1 between Folders—Display 1

```
File Transfer Protocol

Previous FTP subcommands and messages:
200 Representation type is ASCII nonprint.
200 PORT subcommand request successful.
125 List started.
250 List completed successfully.
200 Representation type is IMAGE.
200 PORT subcommand request successful.
150 Retrieving file /QDLS/WS3FLR/DOC1.
250 File transfer completed successfully.
663 bytes transferred in 0.507 seconds. Transfer rate 1.308 KB/sec.
200 PORT subcommand request successful.
150 Retrieving file /QDLS/WS3FLR/DOC2.
250 File transfer completed successfully.
663 bytes transferred in 0.318 seconds. Transfer rate 2.083 KB/sec.
200 PORT subcommand request successful.
150 Retrieving file /QDLS/WS3FLR/DOC3.
250 File transfer completed successfully.
663 bytes transferred in 0.317 seconds. Transfer rate 2.089 KB/sec.
Enter an FTP subcommand.
==> MGET *.* (REPLACE

F3=Exit      F6=Print      F9=Retrieve
F17=Top      F18=Bottom    F21=CL command line
```

Figure 158. FTP MGET Subcommand using NAMEFMT 1 between Folders—Display 2

AS/400-to-VAX/Wollongong

This topic shows how to log on FTP to the VAX and how to use the PUT and GET subcommands.

Process for Logging On: Figure 159 on page 252 shows the FTP logon process to the VAX. Issue the command FTP MVAX. When prompted for the user ID, enter TESTER. When prompted for the password, enter your password.

```

File Transfer Protocol

Previous FTP subcommands and messages:
  Connecting to host name MVAX at address 9.4.6.252 using port 21.
  220 FTP Service Ready
> TESTER
  331 User name TESTER received, please send password
  230 TESTER logged in, directory $DISK1:[TESTER]

Enter password:
===>

F3=Exit      F6=Print      F9=Retrieve
F17=Top      F18=Bottom   F21=CL command line

```

Figure 159. VAX FTP Logon Process

This logon is very similar to the FTP logon to an AS/400. It is necessary to have a valid user ID and password for the VAX.

Figure 160 shows the results of the QUOTE HELP subcommand (entered on the AS/400), which lists the subcommands supported by the Wollongong server software.

```

File Transfer Protocol

Previous FTP subcommands and messages:
  220 FTP Service Ready
> TESTER
  331 User name TESTER received, please send password
  230 TESTER logged in, directory $DISK1:[TESTER]
> quote help
  211-The following commands are accepted:
  USER PASS ACCT MAIL MLFL PORT ABOR QUIT NOOP HELP TYPE MODE
  STRU BYTE ALLO XSEN XSEM XVRB XTER SYST REIN SITE STAT RETR
  STOR APPE DELE RNFR RNT0 LIST NLST REST CWD XCWD CDUP PWD
  XPWD MKD XMKD RMD
  All file specifications must be in VMS** or Unix format.
  211 Problems to Postmaster@mvax
Enter an FTP subcommand.
===> quote help

F3=Exit      F6=Print      F9=Retrieve
F17=Top      F18=Bottom   F21=CL command line

```

Figure 160. VAX FTP Server Subcommand Help Display

Help is not available for individual subcommands. For example, the QUOTE HELP APPE subcommand returns a message that indicates that no information is available.

To determine the functions of some of these remote commands, refer to the appropriate Wollongong reference manuals.

PUT and GET Subcommands: The put and get operations are performed the same way as the AS/400-to-AS/400 put and get operations, but with the following differences.

Figure 161 and Figure 162 show examples of the put operation to the VAX and the get operation from the VAX.

```
File Transfer Protocol

Previous FTP subcommands and messages:

> put GERRYLIB/SCREEN1.screen1 screen1.file
200 PORT Command OK.
125 ASCII transfer started for $DISK1:[TESTER]SCREEN1.FILE;
226 File transfer completed ok
265037 bytes transferred in 6.649 seconds. Transfer rate 39.859 KB/sec.
Enter an FTP subcommand.
==> put GERRYLIB/SCREEN1.screen1 screen1.file

F3=Exit      F6=Print      F9=Retrieve
F17=Top      F18=Bottom    F21=CL command line
```

Figure 161. Put Operation from AS/400 to VAX

```
File Transfer Protocol

Previous FTP subcommands and messages:

> get SCREEN1.FILE GERRYLIB/SCREEN1.SCREEN2
200 PORT Command OK.
125 ASCII transfer started for $DISK1:[TESTER]SCREEN1.FILE;1 (266586 bytes)
226 File transfer completed ok
265037 bytes transferred in 8.496 seconds. Transfer rate 31.195 KB/sec.
Enter an FTP subcommand.
==> get SCREEN1.FILE GERRYLIB/SCREEN1.SCREEN2

F3=Exit      F6=Print      F9=Retrieve
F17=Top      F18=Bottom    F21=CL command line
```

Figure 162. Get Operation to AS/400 from VAX

The specification of files on the VAX is as follows:

NODE::DEVICE:[DIRECTORY]FILENAME.TYPE;VERSION

where directory can be expressed as a subdirectory structure such as [AAA.BBB.CCC.DDD...].

This is the full specification possible. Certain parts can be omitted and defaults can be taken for them. In this example, \$DISK1:[TESTER] uses the default node, \$DISK1 for the device, and TESTER for the directory. Use the CD subcommand to change this directory.

Different versions of files can exist—files with the same name but different version number. This occurs when multiple put operations are done to the same file name.

Note: Because the VAX has this version number inherent in the file name, Wollongong TCP/IP does not support the STOU server subcommand. If SUNIQUE is turned on at the AS/400, the STOU subcommand is sent when PUT and MPUT are run, and Wollongong TCP/IP returns an error message from the VAX.

It is possible to put a logical file from the IBM AS/400 to the VAX. This creates a normal file on the VAX that contains data according to the view of the logical file.

AS/400-to-AIX (UNIX)

This topic shows how to log on FTP to the RISC System/6000 (RS/6000) system and how to use the PUT and GET subcommands.

Logon Process for the RISC System/6000 System: When starting FTP, the user has to log on to the remote host.

Figure 163 shows the logon process to the RS/6000 system.

```
File Transfer Protocol

Previous FTP subcommands and messages:
Connecting to host name SYSNAMRS.SYSNAM123.IBM.COM at address 9.4.73.198 using
port 21.
220 sysnamrs.sysnam123.ibm.com FTP server (Version 4.9 Thu Sep 2 20:35:07 CDT
1993) ready.
> root
331 Password required for root.
230 User root logged in.
UNIX Type: L8 Version: BSD-44

Enter an FTP subcommand.
===>

F3=Exit      F6=Print      F9=Retrieve
F17=Top      F18=Bottom   F21=CL command line
```

Figure 163. AIX FTP Logon Process

This logon is very similar to the FTP logon on an IBM AS/400 business computing system. It is necessary to have a valid user ID and password for the AIX.

Figure 164 on page 255 shows the results of a QUOTE HELP subcommand, which lists the subcommands supported by the AIX server software.

```

File Transfer Protocol

Previous FTP subcommands and messages:

> Quote HELP
214- The following commands are recognized (* =>'s unimplemented).
  USER  PORT  STOR  MSAM*  RNT0  NLST  MKD  CDUP
  PASS  PASV  APPE  MRSQ*  ABOR  SITE  XMKD  XCUP
  ACCT*  TYPE  MLFL*  MRCP*  DELE  SYST  RMD  STOU
  SMNT*  STRU  MAIL*  ALLO  CWD  STAT  XRMD  SIZE
  REIN  MODE  MSND*  REST  XCWD  HELP  PWD  MDTM
  QUIT  RETR  MSOM*  RNFR  LIST  NOOP  XPWD
214 Direct comments to ftp-bugs@sysnamrs.sysnam123.ibm.com.
Enter an FTP subcommand.
==> Quote HELP

F3=Exit      F6=Print      F9=Retrieve
F17=Top      F18=Bottom    F21=CL command line

```

Figure 164. AIX FTP Server Subcommand Help Display

Additional help information may be available for each server subcommand. For example, the following:

```
QUOTE HELP APPE
```

returns information on the parameters for the APPE subcommand.

PUT and GET Subcommands: The put and get operations are performed the same way as the AS/400-to-AS/400 put and get operations, but with the following differences.

The specification of files on AIX is as follows:

```
/DIRECTORY/FILENAME.TYPE
```

where directory can be expressed as a subdirectory structure such as /AAA/BBB/CCC/DDD/...

It is possible to put a logical file from the AS/400 to AIX. This creates a normal file on AIX that contains data according to the view of the logical file.

Figure 165 on page 256 and Figure 166 on page 256 show examples of put and get operations between an AS/400 and AIX.

```

File Transfer Protocol

Previous FTP subcommands and messages:
Connecting to host name SYSNAMRS.SYSNAM123.IBM.COM at address 9.4.73.198 using
port 21.
220 sysnamrs.sysnam123.ibm.com FTP server (Version 4.9 Thu Sep 2 20:35:07 CDT
1993) ready.
> root
331 Password required for root.
230 User root logged in.
UNIX Type: L8 Version: BSD-44
> put ITSOLIB1/QCLSRC.memb1 /tmp/MEMB.TXT
200 PORT command successful.
150 Opening data connection for /tmp/MEMB.TXT.
226 Transfer complete.
13 bytes transferred in 0.214 seconds. Transfer rate 0.061 KB/sec.
Enter an FTP subcommand.
==> put ITSOLIB1/QCLSRC.memb1 /tmp/MEMB.TXT

F3=Exit      F6=Print      F9=Retrieve
F17=Top      F18=Bottom    F21=CL command line

```

Figure 165. AIX Put Operation

```

File Transfer Protocol

Previous FTP subcommands and messages:
> get /tmp/MEMB.TXT ITSOLIB1/QCLSRC.MEMB7
Member MEMB7 in file QCLSRC in library ITSOLIB1 already exists.
Specify REPLACE as a subcommand option.
> get /tmp/MEMB.TXT GERRYLIB/QCLSRC.MEMB7 (REPLACE
200 PORT command successful.
150 Opening data connection for /tmp/MEMB.TXT (11 bytes).
226 Transfer complete.
13 bytes transferred in 0.536 seconds. Transfer rate 0.024 KB/sec.
Enter an FTP subcommand.
==>

F3=Exit      F6=Print      F9=Retrieve
F17=Top      F18=Bottom    F21=CL command line

```

Figure 166. AIX Get Operation

AS/400-to-OS/2

In this topic we see how to log on FTP to a PS/2 running OS/2 and how to use the PUT subcommand with NAMEFMT 1.

It is only meaningful to transfer certain types of files to a PS/2. In this case, we have transferred a batch file from one of the OS/2 system folders.

Logon: When starting FTP, the user has to log on to the remote host. With OS/2, this is no exception. The process is similar to other FTP server systems.

OS/2 Server Considerations: To use OS/2 as an FTP server, you need to set up a file called TRUSERS in the ETC subdirectory or the directory specified by the ETC environment variable. This file contains access authorization for users from client systems. The user IDs and passwords it contains are case sensitive.

Full details are in the *IBM TCP/IP V2.0 for OS/2* manual, SC31-6076. A sample TRUSERS file is shown in Figure 167.

```
[C:\]type c:\tcpip\etc\trusers
user:  GWIL XXXX
rd:    c:\
wr:    c:\
```

Figure 167. OS2 TRUSERS Sample File

The FTPD command starts the server on OS/2. As users log on to OS/2 FTP from client systems, the OS/2 FTP server session displays messages as shown in Figure 168.

```
FTPDC: spawned with socket 55
connection from 9.4.73.212 at Tue May 10 10:10:05 1994

FTP LOGIN FROM 9.4.73.212, GWIL
```

Figure 168. OS2 FTP Server Messages

FTP Put Process: Figure 169 on page 258 shows the PUT subcommand to OS/2 FTP and the associated subcommands that are required. The key points to note are that the transfer must be done with binary images and that name format 1 is required to use the QDLS file system. The section of this name following QDLS is the folder name, in this case QIWSFL2.

```

File Transfer Protocol

Previous FTP subcommands and messages:
 220 SCOTLAND FTP server (IBM OS/2 TCP/IP FTP Version 1.2) ready.
> GWIL
 331 Password required for GWIL.
 230 User GWIL logged in.
 OS/2 operating system
> BINARY
 200 Type set to I.
> NAMEFMT 1
 202 SITE not necessary; you may proceed.
 Client NAMEFMT is 1.
> put /QDLS/QIWSFL2/COPYWSFA.BAT C:\TEST\COPYWSFA.BAT
 200 PORT command successful.
 150 Opening BINARY mode data connection for C:\TEST\COPYWSFA.BAT.
 226 Transfer complete.
 159 bytes transferred in 0.984 seconds. Transfer rate 0.162 KB/sec.
Enter an FTP subcommand.
====> put /QDLS/QIWSFL2/COPYWSFA.BAT C:\TEST\COPYWSFA.BAT

F3=Exit      F6=Print      F9=Retrieve
F17=Top      F18=Bottom   F21=CL command line

```

Figure 169. OS/2 FTP FTP/PUT Process

FTP Considerations (for Both Client and Server)

This section contains additional detailed information that is pertinent to both the AS/400 FTP client and FTP server, including:

- Data transfer methods
- Transferring files that contain packed decimal data
- Transferring AS/400 save files
- Transferring HFS files
- Transferring QDLS documents
- Transferring “root,” QOpenSys and QLANSrv files
- Transferring files using QFileSvr.400
- Receiving text files
- Transferring QSYS.LIB files
- AS/400 file pre-creation
- Automatic sequencing of source files
- CCSID code page tagging for files on AS/400
- NLS considerations
- Effects of job wait time

Data Transfer Methods

The appropriate transmission attributes must be used to transfer files between two systems to preserve the content and structure of the files. A text file contains standard, displayable characters only. It does not include the end-of-record characters (EBCDIC '1E' and ASCII '0D'). A binary file may contain any characters.

Table 22. Recommended Methods for Data Transfer

Transfer between System Types	Data type	Transfer Type ¹ , Mode ²
AS/400 to AS/400	Text	EBCDIC, Stream
AS/400 to ASCII	Text	ASCII, Stream
ASCII to AS/400	Text	ASCII, Stream
ASCII to AS/400 to ASCII	All data	Binary, Stream
AS/400 to AS/400	Mixed data ³	Binary, Stream
AS/400 to AS/400	Mixed data ³	EBCDIC, Block

Notes:

1. The TYPE subcommand defines the way in which the data is to be represented.
2. The MODE subcommand specifies how the bits of data are to be transmitted.
3. See “Transferring Files that Contain Packed Decimal Data between AS/400 Systems”.

Transferring Files that Contain Packed Decimal Data between AS/400 Systems

There is no support in FTP for converting special numeric formats like packed decimal or zoned decimal.

The transfer of packed decimal or zoned decimal data is supported between AS/400 systems when you use either a transfer type of TYPE I (BINARY) or TYPE E (EBCDIC) with a transmission mode of BLOCK; these transfer types send the data as is without any conversion. The results of any other transfer type are unpredictable.

When transferring packed or zoned data in an externally-described QSYS.LIB file, the target file should be pre-created in the same manner as the source file. This restriction applies to data containing any special numeric format or when keyed access is required.

When transferring data with a transfer type of binary, the record length of the target file must be the same as the record length of the source file.

Before packed decimal or zoned decimal data can be transferred to or from other system architectures (such as S/390 or UNIX), you must convert the data to printable form.

Transferring Save Files

Save files must be sent as images and, therefore, require the FTP BINARY subcommand to be run before the GET or PUT subcommands.

When transferring a save file using name format 0, the save file on the receiving system must be pre-created. It is recommended that files are pre-created in other situations as well for reasons of performance and integrity.

The transfer of a save file—because it is a file format peculiar to AS/400—can only be made usable if the sending and receiving systems are both AS/400 systems. However, a save file could be sent to a non-AS/400 system and stored there for backup purposes. The save file could be transferred later to the AS/400 with FTP.

Transferring save files from VM to AS/400—Example: The following example shows how to transfer a save file from VM to an AS/400 for both NAMEFMT 0 and 1. The FTP session has already been initiated, the BINARY subcommand has been issued, and NAMEFMT 0 has been specified.

First, we transfer the file P162484 SAVF310L from our VM A disk to the AS/400. VM FTP requires that we insert a period between its file name and file type. We give it the file name P162484 in library P162484 on the AS/400, and specify REPLACE as it has been pre-created even if it has not been used before. You will recall that pre-creation is mandatory with NAMEFMT 0.

We then change the NAMEFMT to 1, and repeat the file transfer using the new name format. Once again we specify REPLACE as the file exists from the previous step.

Notes about transferring save files:

1. If we had not pre-created the file on the AS/400 before performing the transfer with NAMEFMT 0, the transfer would have appeared to have completed satisfactorily. However, on inspection of the file on the AS/400, it would be seen that a physical file (PF) has been created and not a save file (SAVF).
2. Some preprocessing may be necessary on the VM system depending on how the save file was sent to VM:
 - If FTP was used to send the save file to VM, you can just issue a GET subcommand to transfer it back to the AS/400.
 - If the Send Network File (SNDNETF) command was used to send the save file to VM, it is first necessary to convert the file on the VM system from a record format (RECFM) of variable to a RECFM of fixed before using FTP to transfer it back to the AS/400. To do this, use the COPYFILE command on VM. For example:

```
COPYFILE P162484 SAVF310L A = = = (RECFM F REPLACE
```

```
> GET P162484.SAVF310L P162484/P162484 (REPLACE
200 Port request OK.
150 Sending file 'P162484.SAVF310L'
250 Transfer completed successfully.
384912 bytes transferred in 3.625 seconds. Transfer rate 106.183 KB/sec

> namefmt 1
202 SITE not necessary; you may proceed
Client NAMEFMT is 1.
> GET P162484.SAVF310L/QSYS.LIB/P162484.LIB/P162484.savf (REPLACE
200 Port request OK.
150 Sending file 'P162484.SAVF310L'
250 Transfer completed successfully.
384912 bytes transferred in 3.569 seconds. Transfer rate 107.839 KB/sec
Enter an FTP subcommand.
====>
```

Figure 170. Transferring a save file from VM to AS/400 using NAMEFMT 0 and NAMEFMT 1

Transferring HFS Files

The transfer type must be set to IMAGE (using the TYPE I or BINARY subcommand) when transferring HFS files.

The directory entry attributes of HFS files are not transferred. Default attributes are assigned by the particular HFS file system (for example, QDLS file system) when the file is written at the receiving AS/400 system. These defaulted attributes may not be the same as those on the sending AS/400 system.

Depending on the particular file system (for example, /QDLS), attributes may affect file processing. For example, transferred final-form text (FFTDCA) documents are assigned the default document type PCFILE instead of FFTDCA. This prevents editing and viewing the document content using the Work with Documents (WRKDOC) CL command.

Transferring QDLS Documents

When a QDLS document is transferred, The QDLS directory entry attribute that indicates the type of document is defaulted to the document type PCFILE on the receiving AS/400 system for all document types except revisable-form text (RFT) documents. RFT documents are defaulted to the document type RFTDCA. RFTDCA type documents can be viewed and edited using the WRKDOC CL command. PCFILE type documents cannot be viewed or edited using the WRKDOC CL command.

Notes:

1. SMTP can be used to transfer final-form text (FFTDCA) documents.
2. For further information about document library services, refer to the *Office Services Concepts and Programmer's Guide*.

Transferring “root,” QOpenSys and QLANSrv Files

You must use stream mode (MODE S) and file structure (STRUCT F) when transferring files in the “root,” QOpenSys, and QLANSrv file systems.

“root” and QOpenSys files can exist in any valid code page. Files transferred to the QLANSrv file system are tagged with the code page defined for the network server description corresponding to the directory containing that file.

Data conversion and CCSID assignments vary depending on the transfer TYPE used (see “CCSID Code Page Tagging for New Files on the AS/400” on page 264). TYPE E is not supported for the QLANSrv file system.

When appending data to an existing file, the CCSID tag of that file is not changed. When appending data to an existing file using TYPE A, the data is converted to the code page of that file.

Transferring Files Using QFileSvr.400

The QFileSvr.400 file system provides access to other file systems on remote AS/400 systems. The transfer of files in the “root,” QOpenSys and QLANSrv file systems is supported. The transfer of files in the QSYS.LIB, QDLS and QOPT file systems is not supported.

You must use stream mode (MODE S) and file structure (STRUCT F).

For example, in Figure 171 on page 262, FILE.ABC is transferred to and from three different files systems on system AS012 using the QFileSvr.400 file system on system AS009.

After connecting to system AS009, the FTP client subcommands shown in Figure 172 on page 262 perform the data transfers.

Note: The userid and password on systems AS009 and AS012 must be the same.

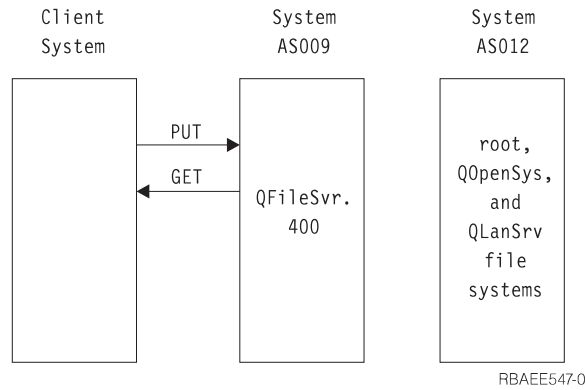


Figure 171. QFileSvr.400 File System Example

```

NAMEFMT 1
LCD /CLIENTDIR1
CD /QFileSvr.400/AS012/FLSDIR
PUT FILE.ABC
GET FILE.ABC /CLIENTDIR2/FILE.ABC
CD /QFileSvr.400/AS012/QOpenSys/FLSDIR
PUT FILE.ABC GET FILE.ABC /CLIENTDIR2/FILE.ABC (REPLACE
CD /QFileSvr.400/AS012/QLANsrv/NWS/LANSRV/DSK/K/FLSDIR
PUT FILE.ABC
GET FILE.ABC /CLIENTDIR2/FILE.ABC (REPLACE
SYSCMD RMVLNK '/CLIENTDIR2/FILE.ABC'
DELETE /QFileSvr.400/AS012/FLSDIR/FILE.ABC
DELETE /QFileSvr.400/AS012/QOpenSys/FLSDIR/FILE.ABC
DELETE /QFileSvr.400/AS012/QLANsrv/NWS/LANSRV/DSK/K/FLSDIR/FILE.ABC
QUIT

```

Figure 172. Subcommands to Transfer Files Using QFileSvr.400

Receiving Text Files on the AS/400 System to the QSYS.LIB File System

Because the AS/400 QSYS.LIB file system internally supports a record structure, the AS/400 FTP converts files received on the AS/400 system into a record structure and converts files sent from the AS/400 system into the FTP file structure. Text files received on the AS/400 system by FTP are converted into a record structure in the following manner:

- When AS/400 FTP receives a file and that file already exists on the AS/400 system, the record length of the existing file is used.
- When AS/400 FTP creates a new file on the AS/400 system, it uses the length (excluding trailing spaces) of the longest line or record in the file as the record length of the file.

Text files sent from the AS/400 system by FTP are converted into a file structure by removing the trailing blanks from each line or record and sending the truncated record.

Transferring QSYS.LIB Files

Table 23 and Table 24 on page 264 summarize FTP operations in stream transfer mode and in image transfer type for the QSYS.LIB file system. Keep the following in mind when using these tables:

- **Compatible record length and file size**
When you send data to a file that already exists, the record and file size of the receiving file must be compatible with the file being sent or a transfer error will occur. Both the record and file size of the receiving file must be greater than or equal to the source file record and file size. To determine if the existing file size is compatible you need to consider the current number of records, the number of extensions allowed, and the maximum record size allowed. You can view this information by entering the AS/400 Display File Description (DSPFD) command.
- **Automatic file creation on the AS/400 system**
When receiving a file, the AS/400 system automatically creates a physical file, if one does not already exist.
However, it is recommended that you pre-create the file on the AS/400. This is discussed in “AS/400 File Pre-Creation” on page 264.
- **When transferring data using TYPE I, the data is not converted.** If the file does not exist, it is tagged with CCSID 65535 when it is created.

Note: File pre-creation is advised when using the MGET and MPUT subcommands to transfer files with multiple members. When a file is not pre-created, FTP creates a file with a maximum record length equal to the longest record of the first member processed. If the record length of any other file member is longer, a data truncation error will occur when transferring that member. Pre-creating a file with a record size to accommodate all members will prevent this error.

Table 23. Stream Transfer Mode for QSYS.LIB File System

Compatible						
Library Exists	File Exists	Member Exists	Replace Selected	Record Length	File Size	Result
Yes	Yes	Yes	Yes	Yes	Yes	Data written to member
Yes	Yes	Yes	No	N/A	N/A	Transfer rejected and message sent.
Yes	Yes	No	N/A	No	Yes	File transfer completed, records truncated, and message returned.
Yes	Yes	No	Yes	No	Yes	File transfer completed, records truncated, and message returned.
Yes	Yes	No	N/A	Yes	Yes	Member created and data written to it
Yes	Yes	No	No	N/A	No	Transfer rejected and message sent
Yes	No	N/A	N/A	N/A	N/A	File created with record length equal to the maximum record length of the incoming file. Member created and data written to member.
No	N/A	N/A	N/A	N/A	N/A	Transfer rejected and message sent. Use the CRTLIB CL command to create a library on the remote AS/400 system.

Table 24. Image Transfer Type for QSYS.LIB File System

Library Exists	File Exists	Member Exists	Replace Selected	Result
Yes	Yes	Yes	Yes	Data written to member
Yes	Yes	Yes	No	Transfer rejected and message sent
Yes	Yes	No	N/A	Member created and data written to it
Yes	No	N/A	N/A	Data file created with record length equal to 512. Member created and data written to it.
No	N/A	N/A	N/A	Transfer rejected and message sent. Use the QUOTE CRTL subcommand to create a library on the remote AS/400 system.

AS/400 File Pre-Creation

We strongly recommend that you pre-create any files that are to be transferred to the AS/400. This is the best method of ensuring that your data is transferred reliably and effectively with optimal performance and integrity.

Be sure to allocate enough records to accommodate the entire file. On the AS/400 this is done in the SIZE parameter of the Create Physical File (CRTPF) command.

Ensure that the RCDLEN parameter of the Create Physical File (CRTPF) command is adequate to accommodate the maximum record length expected.

In addition, if the user profile doing the transfer does not have *NOMAX specified on the MAXSTG parameter, then the result of the calculation (MAXSTG - storage allocated) must be a value at least twice the size of the file being transferred.

Note: You can pre-create files on the FTP server system using the QUOTE subcommand. You can pre-create files on the FTP client system using the SYSCMD subcommand.

Source Files: Sequencing, Timestamp, Level

When you use FTP to transfer data into a source physical file member, the member is automatically resequenced. This automatic resequencing is the same as if you specified Y to the resequencing prompt when exiting from the Start Source Entry Utility display.

In addition, the date and time of the update operation, and the level identifier of the member, but not the file, are updated. This happens automatically and cannot be overridden.

In addition, the member type field is left blank, even if the transfer is from another AS/400.

CCSID Code Page Tagging for New Files on the AS/400

When FTP creates a new file on an AS/400 system, the file is tagged with a CCSID or the code page of that CCSID to identify the character data in that file. For "root," QOpenSys and QLANSrv, the file is always created new except when an append is done. When appending data to an existing file, the tag of the file is not changed. Table 25 on page 265 summarizes how FTP assigns these values for different file systems and transfer types.

Table 25. CCSID Code Page Tagging for New AS/400 Files

Receiving File System	FTP Transfer Type			
	TYPE A (ASCII)	TYPE C ('ccsid')	TYPE E (EBCDIC)	TYPE I (Image/Binary)
QSYS.LIB	Related default EBCDIC CCSID of FTP default ASCII CCSID. If conversion table, then 65535.	'ccsid' if EBCDIC CCSID. If ccid is ASCII, then related default EBCDIC CCSID.	65535	65535
QDLS, QOPT, and other HFS file systems	Not supported	Not supported	Not supported	Not assigned by FTP
"root," QOpenSys	Code page of default ASCII CCSID	'ccsid' value specified in TYPE C ccid# subcommand.	Code page of Job CCSID if it is not 65535. If Job CCSID is 65535, assign code page of Default Job CCSID.	Code page of default ASCII CCSID.
QLanSrv	ASCII code page of the network server description for file directory.	ASCII code page of the network server description for file directory.	Not supported	ASCII code page of the network server description for file directory.
<p>Note:</p> <p>The default ASCII CCSID is defined when the FTP job is started: For the client, the CCSID parameter of the STRTCPFTP (and FTP) command. For the server, the CCSID parameter of the FTP Configuration attributes which can be changed using the CHGFTP command.</p> <p>QFileSvr.400 file assignments depend on the file system receiving the file.</p> <p>See <i>Character Data Representation Architecture</i>, SC41-1390 for additional information</p> <p>When using TYPE C ccid#, the data transferred must be in the ccid# specified. No data conversion occurs when transferring data using TYPE C transfer option.</p>				

National Language Support Considerations for FTP

Be aware of the following when using FTP in an environment with different primary languages.

- When data is transferred using TYPE E (or EBCDIC) the data is stored as is and therefore will be in the EBCDIC code page of the file that it came from. This can result in the stored file being tagged with an inappropriate CCSID value when the primary language of the two AS/400 systems is different.

For example, when data in code page 273 is sent using TYPE E to the QSYS.LIB file system on a machine where the file does not exist, the data is stored as is in a new file tagged with CCSID 65535. If the receiving file already exists, then the data will be received as is and tagged with the existing file CCSID which may not be 273.

To avoid incorrect CCSID tagging, you can use the TYPE C *CCSID* subcommand (for example, TYPE C 273) to specify the CCSID of the data being transferred. When a CCSID is specified on a transfer and the data is written to an existing file, the data is converted to the CCSID of the existing file. If no target file exists before the transfer, a file is created and tagged with the specified CCSID.

In the preceding example, if the target file does not exist, a file with a CCSID of 273 is created on the receiving system. When the target file already exists, the data is converted from CCSID 273 to the CCSID of the target file.

- When starting the FTP client, message TCP3C14: Unable to convert data from CCSID &1; to CCSID &2, may be displayed. This occurs if no character conversion is available between the EBCDIC CCSID specified by your job and the ASCII CCSID specified for the this FTP session. You can change the ASCII CCSID by specifying a value for the coded character set identifier parameter of the STRTCPFTP CL command. CCSID 850, which contains the IBM Personal Computer Latin-1 coded character set, is an ASCII CCSID for which character conversions are available to all valid job CCSID values.
- When using FTP in ASCII mode between two EBCDIC systems, the data on the system sending the file is converted from its stored EBCDIC code page to ASCII, and then from ASCII to the EBCDIC code page of the receiving system. Usually this does not present a problem because the 7-bit ASCII code page used by the two systems is the same unless the EBCDIC characters on the sending system are not defined in the ASCII code page. Also, some characters in the ASCII code page may be mapped differently between the two different EBCDIC code pages. This might occur if some of the ASCII characters are *variant* (the character occupies a different hexadecimal code point in an EBCDIC code page). The variant character may be interpreted differently on the receiving system if the EBCDIC code page is different from that of the system sending the file.

Effects of Job Wait Time on FTP

FTP is dependent on the wait time of the job issuing an FTP request. Use option 3 (Display job run attributes, if active) from the Work with Job (WRKJOB) menu to view the value of the job wait time. To increase the value of the job wait time, use the Change Job (CHGJOB) command.

FTP Client Considerations

This section contains additional detailed information that is pertinent to the AS/400 FTP client including:

- File client naming considerations
- File structure and path name
- Mapping tables
- Server time-out
- Using server subcommands

FTP Client File Naming

When using the PUT and GET subcommands without specifying both the local and remote file name, a name is chosen for you. This name may not be the name you expect or want.

For example, you could type the following subcommand to get your PROFILE EXEC (file name, file type) from an IBM VM (Virtual Machine) system:

```
GET PROFILE.EXEC
```

VM sends the file to your AS/400 system and gives the file a name of PROFILE and a member name of EXEC. However, you may prefer that the file name be EXEC and the member name be PROFILE so that you could later add more members to this "exec" file.

To ensure that the desired name is assigned, you should specifically state the name as follows:

```
GET PROFILE.EXEC EXEC.PROFILE
```

In this example, member *EXEC* of remote file *PROFILE* is copied to your local system as member *PROFILE* of local file *EXEC*.

File Structure and Path Name

File system structure and file path name specifications differ from one system to another. For example, the AS/400 QSYS.LIB file system using NAMEFMT 0 has the format:

```
Libname/Filename.Mbrname
```

and using NAMEFMT 1 has the format:

```
/QSYS.LIB/Libname.LIB/Filename.FILE/Mbrname.MBR
```

and UNIX-based systems allow specifications such as:

```
/etc/hosts  
/usr/jack/test.c  
/usr/*
```

Not all systems use the same keyboards or character sets. This can cause difficulty when specifying path names. For example, some systems use the [character and the] character in the path name to indicate the directory portion of a file name. If you are using an IBM 52xx style keyboard or an IBM Personal Computer emulating a 52xx, these characters are displayed as a ↑ character and as a .. character. You also cannot type the [character and the] character on your keyboard. Instead, you must type the < character and the > character.

Specifying Mapping Tables

For FTP client, the ASCII mapping tables are specified in the FTP command. For FTP server this is done in the Change FTP Attributes (CHGFTPA) command. To specify the FTP client mapping tables:

1. Enter the command FTP.
2. Press PF4. The Start TCP/IP FTP display is shown.
3. Press F10. The prompts for outgoing and incoming ASCII/EBCDIC tables are displayed.

```

Start TCP/IP File Transfer (FTP)

Type choices, press Enter.

Remote system . . . . .

Internet address . . . . .
Coded character set identifier *DFT 1-65533, *DFT

Additional Parameters

Outgoing EBCDIC/ASCII table . . *CCSID Name, *CCSID, *DFT
Library . . . . . Name, *LIBL, *CURLIB
Incoming ASCII/EBCDIC table . . *CCSID Name, *CCSID, *DFT
Library . . . . . Name, *LIBL, *CURLIB

Bottom
F3=Exit F4=Prompt F5=Refresh F12=Cancel F13=How to use this display
F24=More keys

```

Figure 173. Specifying ASCII Mapping Tables with the *CCSID Value

Specify the CCSID (and hence the mapping tables) to be used for the FTP client. When the *DFT value is not changed, the CCSID value 00819 (ISO 8859-1 8 bit ASCII) is used. You may also specify a specific CCSID for both inbound and outbound transfers. The use of CCSIDs is discussed in the *International Application Development* book.

Notes:

1. Double-byte character set (DBCS) CCSID values are not permitted for the CCSID parameter on the CHGFTPA command. The DBCS CCSID values can be specified using the TYPE subcommand.
2. IBM includes mapping support in FTP to ensure compatibility with releases prior to V3R1. Use of mapping tables for incoming TYPE A file transfers results in the loss of CCSID tagging if the target file must be created. IBM strongly recommends that you use CCSID support for normal operations.

Server Time-out Considerations

The inactivity time-out value requires some consideration. This is the time in seconds without FTP server activity that will cause the server to close the session. Certain remote servers allow the client to change this value. For example, the AS/400 supports the FTP server TIME subcommand, which can be sent to the server with the FTP client QUOTE subcommand. UNIX servers often support the SITE IDLE subcommand.

When using local AS/400 subcommands with either the SYSCMD subcommand or F21, there is no interaction between the client and the server. Therefore, if the running of these local AS/400 commands exceeds the server inactivity time-out period, the server will close the connection. If you lose your connection, you must log on to the server again using the OPEN command (OPEN <remote system name>) and the USER command as described in the note to “Logon to the Remote System (Server)” on page 239.

Using Server Subcommands

The subcommands that are sent to the server can be created in two ways:

- By a client subcommand
- Explicitly typed in by the user (with the QUOTE subcommand)

In the first case, the user enters a client subcommand (other than QUOTE) that is read and converted into one or more appropriate server subcommands by the client program. In the second case, the user enters an explicit server subcommand prefaced by the QUOTE subcommand.

```
QUOTE server_subcommand_and_parameters
```

Whatever follows QUOTE is sent verbatim to the server; no conversion or interpretation is done by the client program.

FTP as Batch Job

In addition to running the FTP client interactively, you can run the FTP client in an unattended mode. Two examples of this method are included in this section: a simple example and a complex example.

Batch FTP: A Simple Example

The following is a simple example of a batch file transfer that involves the successful transfer of one file from a remote system.

The components are as follows:

- A CL program
- An input file of FTP commands
- An output file of FTP messages

The CL Program

```
*****
ITSOLIB2/QCLSRC BATCHFTP:
-----
      PGM
      OVRDBF  FILE(INPUT) TOFILE(ITSOLIB2/QCLSRC) MBR(FTPCMDS)
      OVRDBF  FILE(OUTPUT) TOFILE(ITSOLIB2/QCLSRC) MBR(OUT)
      FTP    RMTSYS(SYSxxx)
      ENDPGM
*****
```

The BATCHFTP program overrides the INPUT parameter to the source physical file ITSOLIB1/QCLSRC MBR(FTPCMDS). The output is sent to MBR(OUT).

The Input Commands File

```
*****
RLDICK/QCLSRC FTPCMDS:
-----
ITSO ITS0
CD ITSOLIB1
SYSCMD CHGCURLIB ITSOLIB2
GET QCLSRC.BATCHFTP QCLSRC.BATCHFTP (REPLACE
QUIT
*****
```

The FTP subcommands required are shown in the FTPCMDS file.

The Output Messages File

```
*****
FTP Output Redirected to a File
FTP Input from Overridden File
Connecting to host name SYSxxx
at address x.xxx.xx.xxx using port 21.
220-QTCP at SYSxxx.sysnam123.ibm.com.
220 Connection will close if idle more than 5 minutes.
Enter login ID (itso):
> ITSO ITSO
331 Enter password.
230 ITSO logged on.
OS/400 is the remote operating system. The TCP/IP version is "V3R1M0".
250 Now using naming format "0".
257 "QGPL" is current library.
Enter an FTP subcommand.
> CD ITSOLIB1
Enter an FTP subcommand.
250 Current library changed to ITSOLIB1.
> SYSCMD CHGCURLIB ITSOLIB2
Enter an FTP subcommand.
> GET QCLSRC.BATCHFTP QCLSRC.BATCHFTP (REPLACE
200 PORT subcommand request successful.
150 Retrieving member BATCHFTP in file QCLSRC in library ITSOLIB1.
250 File transfer completed successfully.
147 bytes transferred in 0.487 seconds. Transfer rate 0.302 KB/sec.
Enter an FTP subcommand.
> QUIT
221 QUIT subcommand received.
*****
```

The output file is shown. It is a straightforward matter to write a program to process this file and display an error message on QSYSOPR if there are any error messages. FTP error messages have numbers that start with a 4 or 5.

Batch FTP: A Complex Example

The following example shows how to retrieve files from several remote hosts to a central AS/400 in batch mode:

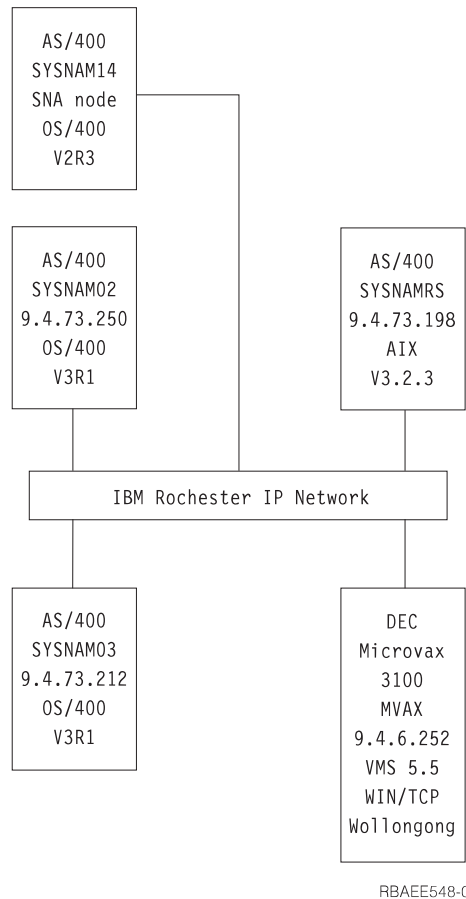


Figure 174. Network Example for Batch FTP

User GWIL on AS/400 SYSNAM03 wants to:

1. Retrieve files from hosts SYSNAMRS (RS/6000) and MVAX (VAX).
2. After retrieving the file from SYSNAMRS, the file should be transferred to SYSNAM02 (another AS/400) using FTP.
3. From there the file is to be sent using SNA to AS/400 SYSNAM14.

Create a CL Program to Start FTP

1. As we have seen in the previous example, FTP uses the display station for command INPUT and message OUTPUT, and this needs to be overridden for use in batch mode. We use the OVRDBF command to overwrite these files with the ones to be used in batch:

```
OVRDBF FILE(INPUT) TOFILE(GERRYLIB/QCLSRC) MBR(FTPCMDS)
OVRDBF FILE(OUTPUT) TOFILE(GERRYLIB/QCLSRC) MBR(FTPLOG)
```

2. A host name or an internet address is a required parameter for the STRTCPFTP command that is included in the CL program file. However, if one wants to specify the remote systems in the input commands file instead of the CL program file, then a dummy host name must be specified for the STRTCPFTP command to satisfy the required syntax. This dummy name may be a fictitious host name or a real host name. If it is a real name, then the first entry in the input commands file must be a user ID and a password, and the second entry must be the CLOSE subcommand. If it is not a real host name, then these entries are not required, and the first entry should be an OPEN subcommand to connect to the desired server system.

FTP RMTSYS(LOOPBACK)

FTP processes the input file and writes messages to the output file (FTPLOG).

3. After the FTP application ends, delete the overrides:

DLTOVR FILE(INPUT OUTPUT)

The CL program for batch FTP will look like the following example on system SYSNAM01:

```
Columns . . . : 1 71          Browse          GERRYLIB/QCLSRC
SEU==>
FMT **  ...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7
***** Beginning of data *****
0001.00 PGM
0002.00          OVRDBF  FILE(INPUT) TOFILE(GERRYLIB/QCLSRC) +
0003.00          MBR(FTPCMDS)
0004.00          OVRDBF  FILE(OUTPUT) TOFILE(GERRYLIB/QCLSRC) +
0005.00          MBR(FTPLOG)
0006.00          FTP     RMTSYS(LOOPBACK) /* (FTP CL Program) */
0007.00          DLTOVR  FILE(INPUT OUTPUT)
0008.00 ENDPGM
***** End of data *****

F3=Exit  F5=Refresh  F9=Retrieve  F10=Cursor  F12=Cancel
F16=Repeat find  F24=More keys

(C) COPYRIGHT IBM CORP. 1981, 1994.
```

Figure 175. CL Program FTPBATCH for Batch FTP

Create the FTP Input File (FTPCDMS)

This file has to contain all the FTP client subcommands necessary to connect and log on to the server, set up for and do the file transfers, close the server connection, and end the client session. The example in Figure 176 on page 273 shows the subcommands used for transferring files to two different remote systems.

```

Columns . . . : 1 71          Browse          GERRYLIB/QCLSRC
SEU==>          FTPCMDS
FMT **  ...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7
***** Beginning of data *****
0001.00 gwil ****
0002.00 close
0003.00 open sysnamrs
0004.00 user root root
0005.00 ascii
0006.00 syscmd dltf file(gerrylib/rs6)
0007.00 get /Itsotest gerrylib/rs6.rs6
0008.00 close
0009.00 open mvax
0010.00 user tester tester
0011.00 get screen1.file gerrylib/vax.vax (replace
0012.00 close
0013.00 open sysnam02
0014.00 user gwil ****
0015.00 ebcdic
0016.00 put gerrylib/rs6.rs6 gerrylib/rs6.rs6
0017.00 quote rcmd sndnetf file(gerrylib/rs6) tousrid((gwil sysnam14))
0018.00 close
0019.00 quit
***** End of data *****
F3=Exit   F5=Refresh   F9=Retrieve   F10=Cursor   F12=Cancel
F16=Repeat find   F24=More keys

```

Figure 176. FTP Client Subcommands (File Member FTPCMDS)

The explanation for the FTP client subcommands as shown in Figure 176 follows. The line numbers on the display correspond to the numbers that follow.

- 0001** User ID and password for dummy connection within client AS/400 SYSNAM03.
- 0002** Close dummy connection in AS/400 SYSNAM03.
- 0003** Open control connection to RISC System/6000 SYSNAMRS.
- 0004** USER subcommand with user ID and password for SYSNAMRS.

Note: When running FTP in batch mode, the USER subcommand must follow an OPEN subcommand. Both the logon user ID and password parameters for the USER subcommand should be provided. This is different when operating FTP interactively online. When FTP is run interactively online, then the client will automatically initiate a USER subcommand and prompt you for a logon ID. There is no automatic USER subcommand when running FTP in batch mode.

- 0005** Transfer ASCII data (will be converted on AS/400 to/from EBCDIC).
- 0006** CL command to be run on client AS/400: delete file. Instead parameter (REPLACE could be used with the next statement).
- 0007** Retrieve file from RISC System/6000 system
- 0008** Close control connection to RISC System/6000 SYSNAMRS.
- 0009** Open connection to VAX MVAX.
- 0010** USER subcommand with user ID and password for MVAX.
- 0011** Retrieve file from VAX replacing existing AS/400 file.
- 0012** Close control connection to VAX MVAX.
- 0013** Open control connection to remote AS/400 SYSNAM02.

- 0014** USER subcommand with user ID and password for SYSNAM02.
- 0015** Transfer EBCDIC data (as it is from AS/400 to AS/400).
- 0016** Send AS/400 file to AS/400 SYSNAM02 with TCP/IP.
- 0017** Send this file from server AS/400 SYSNAM03 to remote AS/400 SYSNAM14 through SNA network.
- 0018** Close control connection to AS/400 SYSNAM02.
- 0019** End FTP application.

Create CL Program for Submitting the FTPBATCH Job

To schedule the file transfers and run them unattended, create a CL program that submits the FTPBATCH job. In Figure 177, the file transfers are supposed to run the next Friday, 17:00 hour, in unattended mode.

```

Columns . . . : 1 71          Browse          GERRYLIB/QCLSRC
SEU==>          FTPSUBMIT
FMT **  ...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7
***** Beginning of data *****
0001.00 PGM
0002.00          SBMJOB      CMD(CALL PGM(GERRYLIB/FTPBATCH)) +
0003.00                                JOB(FTPFRIDAY) OUTQ(QUSRSYS/GERRYQ)  +
0004.00                                SCDDATE(*FRI) SCDTIME(170000) /* FTP for +
0005.00                                Friday, 5:00 in the afternoon */
0006.00 ENDPGM
***** End of data *****

F3=Exit  F5=Refresh  F9=Retrieve  F10=Cursor  F12=Cancel
F16=Repeat find      F24=More keys

(C) COPYRIGHT IBM CORP. 1981, 1994.

```

Figure 177. CL Program for Submitting Batch FTP Job.

Check the FTP Output File for Errors

While running at the scheduled time, FTP creates the data in file member FTPLOG shown in Figure 178 on page 275. The data in file member FTPLOG corresponds to original statements found in Figure 175 on page 272 and in Figure 176 on page 273.

```

Connecting to host name LOOPBACK at address 127.0.0.1 using port 21.
220-QTCP at localhost.
220 Connection will close if idle more than 5 minutes.
Enter login ID (gwil):

>>>GWIL ****
331 Enter password.
230 GWIL logged on.
OS/400 is the remote operating system. The TCP/IP version is "V4R2M0".
250 Now using naming format "0".
257 "QGPL" is current library.
Enter an FTP subcommand.

> CLOSE
221 QUIT subcommand received.
Enter an FTP subcommand.

> OPEN SYSNAMRS
Connecting to host name SYSNAMRS at address 9.4.73.198 using port 21.
220 sysnamrs.sysnam123.ibm.com FTP server (Version 4.9 Thu Sep 2 20:35:07 CDT
1993) ready.
Enter an FTP subcommand.

```

Figure 178. FTP Output (FTPLOG) After Running FTPBATCH Program (Part 1)

```

> USER root ****
331 Password required for root.
230 User root logged in.
UNIX Type: L8 Version: BSD-44
Enter an FTP subcommand.

> ASCII
200 Type set to A; form set to N.
Enter an FTP subcommand.

> SYSCMD DLTF FILE(GERRYLIB/RS6)
Enter an FTP subcommand.

> GET /Itsotest GERRYLIB/RS6/RS7
200 PORT command successful.
150 Opening data connection for /Itsotest (467 bytes).
226 Transfer complete.
467 bytes transferred in 2.845 seconds. Transfer rate 0.167 KB/sec.
Enter an FTP subcommand.

```

Figure 179. FTP Output (FTPLOG) after Running FTPBATCH Program (Part 2)

```

> CLOSE
221 Goodbye.
Enter an FTP subcommand.

> OPEN MVAX
Connecting to host system mvax at address 9.4.6.252 using port 21.
220 FTP Service Ready
Enter an FTP subcommand.

> USER TESTER *****
331 User name TESTER received, please send password
230 TESTER logged in, directory $DISK1:^TESTER|
Enter an FTP subcommand.

GET SCREEN1.FILE GERRYLIB/VAX.VAX (REPLACE
200 PORT Command OK.
125 ASCII transfer started for $DISK1:^TESTER|SCREEN1.FILE;1(266586 bytes)
226 File transfer completed ok.
265037 bytes transferred in 8.635 seconds. Transfer rate 30.694 KB/sec.
Enter an FTP subcommand.

> CLOSE
221 Goodbye.
Enter an FTP subcommand.

OPEN SYSNAM02
Connecting to host system SYSNAM02 at address 9.4.73.250 using port 21.
220-QTCP at SYSNAM02.sysnam123.ibm.com.
220 Connection will close if idle more than 5 minutes.
Enter an FTP subcommand.

```

Figure 180. FTP Output (FTPLOG) after Running FTPBATCH Program (Part 3)

```

> USER GWIL ****
331 Enter password.
230 GWIL logged on.
OS/400 is the remote operating system. The TCP/IP version is "V4R2M0".
250 Now using naming format "0".
257 "QGPL" is current library.
Enter an FTP subcommand.

> EBCDIC
200 Representation type is EBCDIC nonprint.
Enter an FTP subcommand.

> PUT GERRYLIB/RS6.RS6 GERRYLIB/RS6.RS6
200 PORT subcommand request successful.
150 Sending file to member RS6 in file RS6 in library GERRYLIB.
250 File transfer completed successfully.
467 bytes transferred in 0.148 seconds. Transfer rate 3.146 KB/sec.
Enter an FTP subcommand.

> RCMD SNDNETF FILE(GERRYLIB/RS6) TOUSRID((GERRYLIB SYSNAM14))
250 Command SNDNETF FILE(GERRYLIB/RS6) TOUSRID((GWIL SYSNAM14))
successful.
Enter an FTP subcommand.

```

Figure 181. FTP Output (FTPLOG) after Running FTPBATCH Program (Part 4)


```
> CLOSE
221 QUIT subcommand received.
Enter an FTP subcommand.
> QUIT
(This ends the FTP application)
```

Figure 182. FTP Output (FTPLOG) after Running FTPBATCH Program (Part 5)

You should check this output for errors that might have occurred during FTP processing. You can either check visually or run a program that tests for error reply codes. Three-digit FTP error reply codes start with 4 or 5. Be careful to avoid messages such as '467 bytes transferred...'.|
|
|

Sample Procedure: A sample REXX procedure and a sample physical file member are shipped as part of the TCP/IP product. File QATMPINC in library QTCP includes the following two members:

- BATCHFTP that contains REXX source code to specify the input and output batch files, and start FTP.
- BFTPFILE that contains the subcommands and data required for logon and running FTP.

Exit Points for FTP

This material is covered in the **AS/400e Information Center** under the **TCP/IP** topic. For more information see "TCP/IP Topics in the Information Center" on page xv.

Chapter 8. File Transfer Protocol (FTP) Server

The File Transfer Protocol (FTP) server function allows you to send or receive copies of files to or from server systems across a TCP/IP network. In addition, the FTP server provides functions for renaming, adding, and deleting files.

You can access FTP server functions via a command line interface or the Operations Navigator (graphical user interface). Not all FTP functions are available on both interfaces.

This chapter discusses how you start and stop the FTP server via the command line interface and through Operations Navigator. This chapter does not document any of the other Operations Navigator functions. This chapter deals primarily with command line interface functions. See the on-line Help for Operations Navigator for information about using the Operations Navigator for FTP functions.

Also, note that even though some of the functionality of the command line interface and the Operations Navigator is the same, the actual menu commands and parameters are not necessarily the same.

More File Transfer Protocol (FTP) Server documentation material is covered in the **AS/400e Information Center** under the **TCP/IP** topic. For more information see "TCP/IP Topics in the Information Center" on page xv.

FTP Server-What It Does and Does Not Support

The following sections provide a brief description of the functions of the FTP server, and some details of its limitations.

Functions Supported by AS/400 FTP Server

- Exit points to enable "Anonymous" FTP and FTP security controls (see "File Transfer Protocol (FTP) Exit Points" on page 553).
- Transferring files in the "root," QOpenSys, QLANSrv file, and QFileSvr.400 systems.
- Transferring folders and documents in the document library services (QDLS) file system.
- Coded character set identifier (CCSID) support
- Creating and deleting libraries, files, and members using the AS/400 FTP server subcommands.
- Creating and deleting folders using the AS/400 FTP server subcommands.
- Sending or receiving save files and members in physical files, logical files, Distributed Data Management (DDM) files, and source physical files.
- Sending text files in EBCDIC format or converting them to ASCII (the default format).
- Transferring binary files as is.
- Using the current library. The TCP/IP FTP server uses the current library specified in the AS/400 user profile. If no library is specified in the user profile, or if an error occurs, the QGPL library is used as the current library.

- Using the home directory. The TCP/IP FTP server uses the home directory specified in the AS/400 user profile. If the specified directory cannot be accessed, the root directory (/) is used as the home directory.
- Selecting the initial working directory. The initial working directory may be set to either the user's current library or the user's home directory.

Functions Not Supported by FTP Server

- Logging on the AS/400 FTP server is not the same as signing on the AS/400 system. The initial program to call (INLPGM) parameter does not run.
- Transferring selected records within files.
- Selection or omission of fields within records when transferring files. You must transfer an entire physical or logical record.

Configuring FTP Servers

The TCP/IP Connectivity Utilities for AS/400 licensed program comes with TCP/IP FTP servers configured. You can configure more servers with the Change FTP Attributes (CHGFTP) command. The display is shown in Figure 183 on page 281. The CHGFTP command updates a database file called QUSRSYS/QATMFTP.CONFIG, which is used by the FTP servers.

Starting FTP Servers

To use the command line interface to start the FTP server(s):

The server job for a TCP/IP application must be started in the QSYSWRK subsystem. If no FTP server jobs are running, the Start TCP/IP Server (STRTCPSVR) command starts the number of FTP servers that have been configured and specified to start automatically in the FTP server configuration. If you have not configured any FTP servers, the FTP servers that have been shipped with the TCP/IP Utilities licensed program will be started.

Typically, the STRTCP command will start the FTP server job. The only time that is not true is if you specify *NO for the AUTOSTART parameter in the FTP server configuration. In this case, the FTP server job is started when STRTCPSVR *FTP is issued. The STRTCPSVR command overrides the AUTOSTART parameter.

To manually start further FTP server jobs, use the command STRTCPSVR *FTP to start one additional FTP server.

FTP server jobs remain active for use by subsequent users after a client session has been ended.

Note: The FTP server cannot start if character conversion is not available between the system's default job CCSID and the ASCII CCSID specified by the FTP attributes. In this case, message TCP3C14, Unable to convert data from CCSID &1; to CCSID &2;, appears in the FTP server job log. If this situation occurs, you must specify a different ASCII CCSID for which character conversion is available by using the CHGFTP command. CCSID 850, which contains the IBM PC Latin-1 coded character set, is an ASCII CCSID for which character conversions are available to all valid job CCSID values.

Available FTP Servers

The minimum number of available FTP servers to be kept ready for future client connections is the number of initial servers specified in the NBRSVR parameter in the FTP server configuration. When a client connects to an AS/400 FTP server, the server examines the number of active servers that are not connected to a client and the value specified in the NBRSVR parameter. If the NBRSVR parameter is greater than the number of available servers, additional servers are started so that the two numbers are equal. If the NBRSVR parameter is less than the number of available servers, no action is taken.

For example, if there are five FTP client sessions established at the same time and the NBRSVR parameter is set at 10, there will be 15 FTP servers running. The 15 servers include five servers for the five active client sessions and ten available servers. The number of available servers can be larger than the NBRSVR parameter. In this same example, if the five clients end their sessions and no other sessions are started, there will be 15 available servers.

Changes to the NBRSVR parameter take effect at the time of the next client connection, when the above process is activated.

```
Change FTP Attributes (CHGFTPA)

Type choices, press Enter.

Autostart servers . . . . . *YES          *YES, *NO, *SAME
Number of initial servers . . . 3          1-20, *SAME, *DFT
Inactivity timeout . . . . . 300          0-2147483647, *SAME, *DFT
Coded character set identifier 00819      1-65533, *SAME, *DFT
Server mapping tables:
  Outgoing EBCDIC/ASCII table . *CCSID    Name, *SAME, *CCSID, *DFT
  Library . . . . .              Name, *LIBL, *CURLIB

  Incoming ASCII/EBCDIC table . *CCSID    Name, *SAME, *CCSID, *DFT
  Library . . . . .              Name, *LIBL, *CURLIB
Initial name format . . . . . *LIB       *LIB, *SAME, *PATH
Initial directory . . . . . *CURLIB     *CURLIB, *SAME, *HOMEDIR
Initial list format . . . . . *DFT      *DFT, *SAME, *UNIX
New file CCSID . . . . . *CALC        *1-65533, *SAME, *CALC

Bottom
F3=Exit  F4=Prompt  F5=Refresh  F12=Cancel  F13=How to use this display
F24=More keys
```

Figure 183. Change FTP Attributes (CHGFTPA) Display

Ending FTP Servers

To end all AS/400 FTP servers, type the following AS/400CL command:

```
ENDTCPSVR *FTP
```

It is not possible, using the ENDTCPSVR command, to selectively end only some servers of a given type. To do this, you need to use the End Job (ENDJOB) command against each server job.

Ending and Restarting FTP Server Jobs

To use the command line interface to end the FTP server:

1. Enter the following command to verify that FTP servers are currently running:

```
WRKACTJOB SBS(QSYSWRK)
```

If no jobs are listed with a name of the form QTFTPxxxxx (where xxxxx is a 5-digit number), no further action is required. (Make sure to page down to the bottom of the list, as the jobs may not be listed on the first page of the display.) If one or more jobs are listed with this name, continue with the steps that follow.

2. Enter the following command to verify that no users are logged on to the FTP server:

```
NETSTAT OPTION(*CNN)
```

Check that no connections are listed with a local port of ftp-con and a state of Established. If FTP connections are established, wait a few minutes and check again. (If you perform the following steps when there are established FTP connections, users on your system may lose data.) Again, make sure to page down to the bottom of the list.

3. Stop and restart the FTP servers by entering the following commands:

```
ENDTCPSVR SERVER(*FTP)  
STRTCPSVR SERVER(*FTP)
```

To use the Operations Navigator to stop the FTP server:

- Follow the path Network\Servers\TCP/IP
- Right-mouse click the FTP server
- Select Stop

FTP Server Subcommands

This material is covered in the **AS/400e Information Center** under the **TCP/IP** topic. For more information see "TCP/IP Topics in the Information Center" on page xv.

FTP Server Considerations

This section contains additional detailed information about the FTP server, including:

- FTP from non-AS/400 to AS/400
- FTP server jobs and job names
- Creating FTP server spooled job logs
- FTP server NAMEFMT

FTP Server Considerations for Non-AS/400 Clients

This topic covers the considerations for non-AS/400 clients. Two specific systems have been evaluated and the following points apply in both systems.

- VAX/Wollongong
- AIX (UNIX)

When using FTP from or to the AS/400, both VAX and AIX users should be aware of the AS/400 library/file.member structure and naming conventions when using naming format 0.

Points to Note:

1. When starting FTP from the VAX or AIX, the user is prompted for a name. This is actually the user ID for the AS/400.
2. The CD subcommand is used to change the current library on the AS/400. (This is the equivalent to the AS/400 CHGCURLIB command.)
3. The DIR subcommand is used to display the contents of the current library. This command may take some time to run if the current library contains a large number of objects.
4. Some of the error messages returned may not be very clear. For example, if you try to delete the current library, the message returned is “Unable to delete library.”
5. If a file with a name but no type is put to the AS/400, a file and member of that name will be created or replaced. For example:

```
PUT TEST
```


will create or replace a file named TEST with a member named TEST in the current library.
6. The AIX operating system is case sensitive.

FTP Server NAMEFMT

When an FTP server session is started, NAMEFMT is set to the value specified by the INLNAMFMT setting in the FTP configuration. You can change the NAMEFMT value by using the SITE subcommand.

The server automatically switches from the default of NAMEFMT 0 to NAMEFMT 1 when the ‘first’ file or pathname parameter received in a subcommand either:

- Starts with a slash (/) or a tilde (~) character
or
- Is blank (except for the LIST and NLST subcommands).

Any subsequent server subcommands with a file or path name parameter will not affect the NAMEFMT value. In addition to changing the NAMEFMT, the server reply for the subcommand will include a statement saying that the NAMEFMT value has been changed.

For example, the server NAMEFMT value will be changed to “1” if the first server subcommand with a file or path name is:

```
CWD /DIR1/DIR2A
```

The server reply will be:

```
250-NAMEFMT set to 1.  
250 Current directory changed to /DIR1/DIR2A.
```

Note: This capability enables the typical Web browser, which requires NAMEFMT 1, to interact with AS/400 FTP servers without issuing a SITE NAMEFMT 1 subcommand.

Exit Points for FTP Server Security and Anonymous FTP

This material is covered in the **AS/400e Information Center** under the **TCP/IP** topic. For more information see "TCP/IP Topics in the Information Center" on page xv.

Chapter 9. Post Office Protocol (POP) Mail Server

Note:

You can access POP server functions via a command line interface or the Operations Navigator (graphical user interface). Not all POP functions are available on both interfaces.

This chapter discusses how you start and stop the POP server via the command line interface and through Operations Navigator. This chapter does not document any of the other Operations Navigator functions. This chapter deals primarily with command line interface functions. See the online Help for Operations Navigator for information about using the Operations Navigator for POP functions.

Also, note that even though some of the functionality of the command line interface and the Operations Navigator is the same, the actual menu commands and parameters are not necessarily the same.

The **Post Office Protocol (POP) server** is the AS/400 implementation of the Post Office Protocol Version 3 mail interface. This server allows AS/400 systems to act as POP servers for any clients that support the POP mail interface. This includes clients running on Windows, OS/2, AIX and Macintosh.

Multipurpose Internet Mail Extensions (MIME) is the Internet standard for sending mail with headers that describe the contents of the mail messages to the receiving client. These messages can be video, image, audio, or binary files, or text messages. The POP server allows users to exchange mail (including MIME mail) between OfficeVision/400 and POP clients by using the AnyMail/400 mail server framework. This support is provided by **user exits** or **snap-ins** that are provided by the POP server to run in the AnyMail/400 mail server framework.

POP mail clients use *verbs* to communicate with the POP server. Verbs supported by the AS/400 POP server are described in "Supported POP Verbs" on page 296.

The POP Version 3 mail interface is defined in RFC 1725. RFC stands for Request for Comments. RFCs are the vehicles that are used to define evolving Internet standards.

How the POP Server Works

The POP server is a simple store-and-forward mail system. It provides electronic mailboxes on AS/400 from which clients can retrieve mail. It uses the AnyMail/400 mail server framework and the system distribution directory to process and distribute E-mail. It uses simple mail transfer protocol (SMTP) to forward mail.

The **system distribution directory** is an IBM-supplied function that allows you to create entries for user IDs or system addresses specific to your network.

Figure 184 on page 286 provides an overview of the standard POP server components. If you are using AS/400 Client Access with a MAPI mail client instead of a standard POP client, the processing is slightly different. Figure 185 on page 287

page 287 provides an overview of the Client Access-based POP server components.

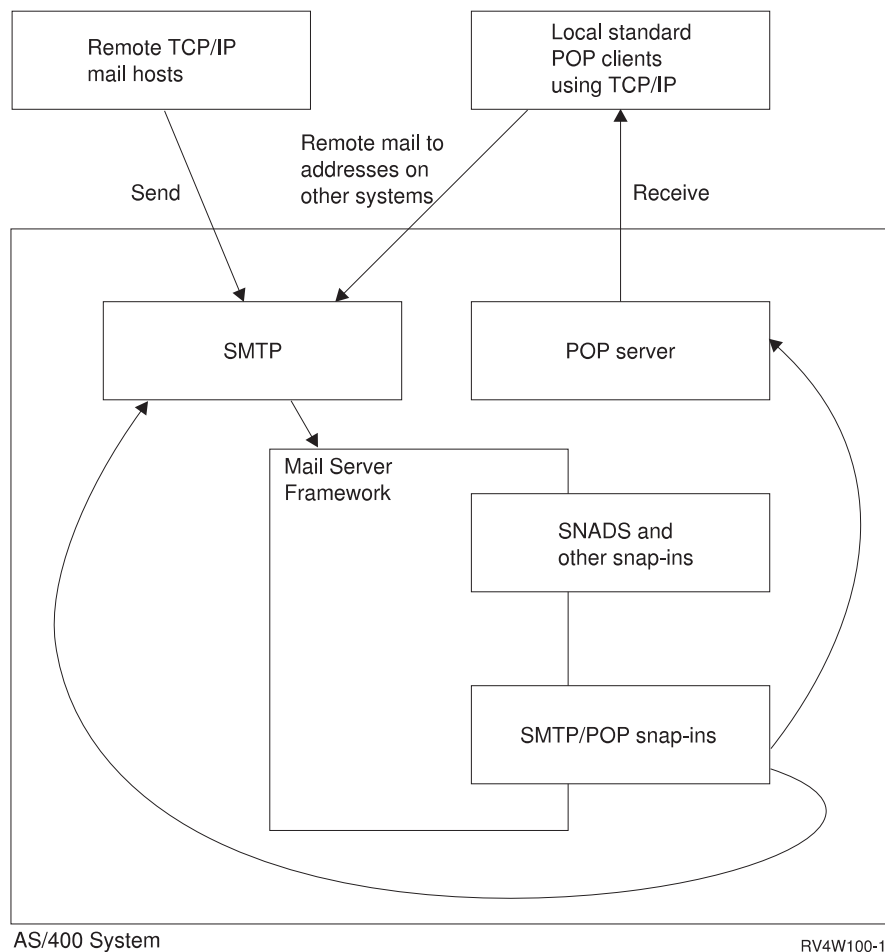


Figure 184. Overview of POP Server Components. Shows “standard” POP clients such as Netscape** and Eudora**.

All incoming mail from SMTP for local users (users with mail accounts on this AS/400) is processed by the AnyMail/400 framework. The mail server framework is a mail distribution structure that allows the distribution of E-mail. The mail server framework calls exit programs or snap-ins to handle specific mail types.

For the client/server interface to work, SMTP must be running for the following reasons:

- Both Internet mail and mail that is sent to clients on the same system go through SMTP.
- Any mail that goes through the mail server framework needs to go through SMTP (through a snap-in) to be delivered to external users.

The POP server serves as a temporary holding area for mail until it is retrieved by the mail client — it does not provide a “mail store” function. When the mail client connects to the server, it queries the contents of its mailbox to see if there is any mail to retrieve. If there is, it retrieves the mail one message at a time. Once a message has been retrieved, the client normally instructs the server to mark that message for deletion when the client session is complete. The client retrieves all of

the messages in the mailbox and then issues a command (in the form of a QUIT verb) that tells the server to delete all of the messages that are marked for deletion and to disconnect from the client.

The POP Server and Client Access-based Mail

The AS/400 POP server can act as a messaging and address book server for MAPI-based Client Access for Windows 95/NT clients. With this support, all mail is sent to the POP server on the AS/400 by way of extensions to the standard POP client/server interface. No SMTP connection on the client is required. This is shown in Figure 185.

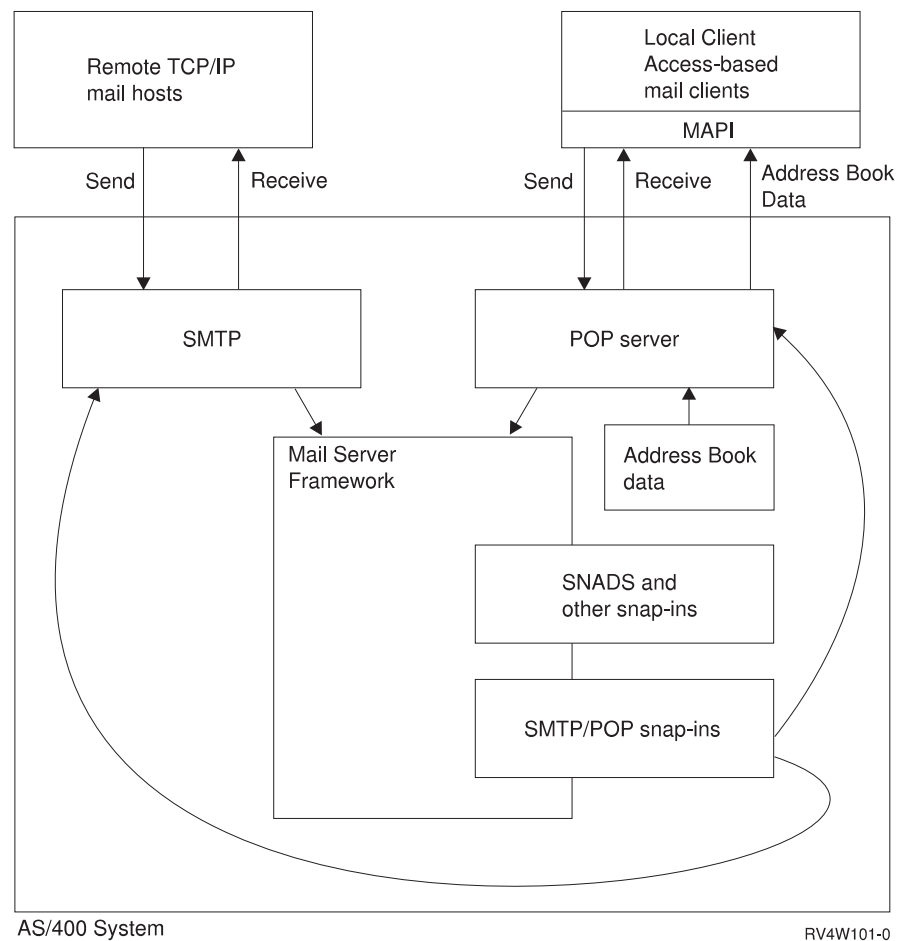


Figure 185. Overview of POP Server Components (with MAPI Service Providers). Shows additional function provided with the Client Access for Windows 95/NT product.

Client Access-based clients can send and receive mail through the POP server with any of these address types:

- INTERNET (the standard Internet format, sometimes referred to as an SMTP address)
- OFFICEVISION (the SNADS address itself, not an SMTP address that is converted to SNADS. This type also includes AS/400 distribution lists.)
- AS400FAX (the dialing sequence as defined by the Facsimile Support for OS/400 LPP).

This support also includes an address book function that provides high-performance client/server access to an address book that is periodically refreshed from the AS/400 system distribution directory.

Finally, the following connection types are supported between the Client Access-based client and the POP server:

- TCP/IP protocol
- IPX/SPX protocol
- SNA protocol.

When you connect to the POP server using Client Access, you gain the benefit of secure logon - the password encryption that Client Access provides.

See "AS/400 Address Book" on page 311 for more information on the supported address types, and for information about how data is mapped from the system distribution directory to the address book cache. See "Configuring POP for Client Access-Based Mail Users" on page 294 for information on how to configure this support.

How to Get the POP Server Up and Running

To get the POP server up and running, you must do the following:

1. Install TCP/IP Connectivity Utilities for AS/400.
 - Use option 12 of the Configure TCP/IP (CFGTCP) command to display the local domain and host names used to identify the local system.
If local domain and host names are present, record them for later use. If they are not present, fill this information in, and record it. You will need this information when you set up clients.
2. Use the Work with Active Jobs (WRKACTJOB) command to determine if the QSNADS subsystem is running. If not, use the Start Subsystem (STRSBS) command to start the QSNADS subsystem:

```
STRSBS QSNADS
```
3. With QSNADS running, use the WRKACTJOB command to determine if the mail server framework is running (look in subsystem QSYSWRK for jobs named QMSF). If the framework is not running, use the Start Mail Server Framework (STRMSF) command to start it.
4. If you are using a Client Access connection, install the following products on your system:
 - SS1 product, Host Servers
 - XA1 product, Client Access/400 Base Family
 - XD1 product, Client Access/400 for Windows 95/NT
5. If you are using Client Access for Windows 95/NT to connect to the POP server with the IPX protocol, define and start IPX. See *Internetwork Packet Exchange (IPX) Support* for information on how to do this.
6. If you are using Client Access connections (of any type), run the Start Host Server (STRHOSTSVR) command, as follows:

```
STRHOSTSVR *ALL
```

For more information on setting up your Client Access client, see the online *User's Guide*, the online information for Windows 95, and the information shipped with Lotus** :Mail**.

7. If SMTP is not set up, you must do the following:
 - Get SMTP up and running with entries in the System Distribution Directory for local users and users of other systems in the network that you might want to send E-mail to.

8. Set up some system distribution directory entries:

You need a user profile and a directory entry for each local POP user.

The *Mail service level* must be set to 2 (System message store) and the *Preferred address* should usually be set to 3 (SMTP name). See "Adding POP Mail Users to the System Distribution Directory" for directions on how to do this. (If you are using SNADS and not TCP/IP, you can set the *Preferred address* to 1 (User ID/address).)

For security reasons, you may want to set up these user profiles as *SIGNOFF profiles, since POP mail users do not need to actually sign on to the AS/400 system.

Note: If you use *SIGNOFF profiles, the system administrator will need to manage password expiration, since *SIGNOFF users will not know when their passwords expire.

9. Set up your clients.
10. Know what kind of client you are connecting. Make sure that the right connection type is configured for the server (using the CHGPOPA command).
11. Start the POP Server (see "Starting the POP Server" on page 296).

The remainder of this chapter provides more detailed information about setup, configuration, exit programs, and ASCII/EBCDIC conversion.

Setting Up Your System and Users

The system distribution directory (SDD) contains entries for users who are authorized to send and receive messages, data, or objects on the network. These users are on both the local system and remote systems. Entries in the SDD are for individual users or groups of users. Entries determine where mail is delivered (allowing users to send and receive mail from OfficeVision/400, SNADS, and SMTP mail systems). For more detailed information on the SDD, see the *SNA Distribution Services* book.

Adding POP Mail Users to the System Distribution Directory

Before you can add users to the system distribution directory (SDD), they must have a user profile. Once they have a user profile, you can use the Work with Directory Entries (WRKDIRE) command to add them to the directory.

1. Type WRKDIRE to display the Work with Directory Entries menu.

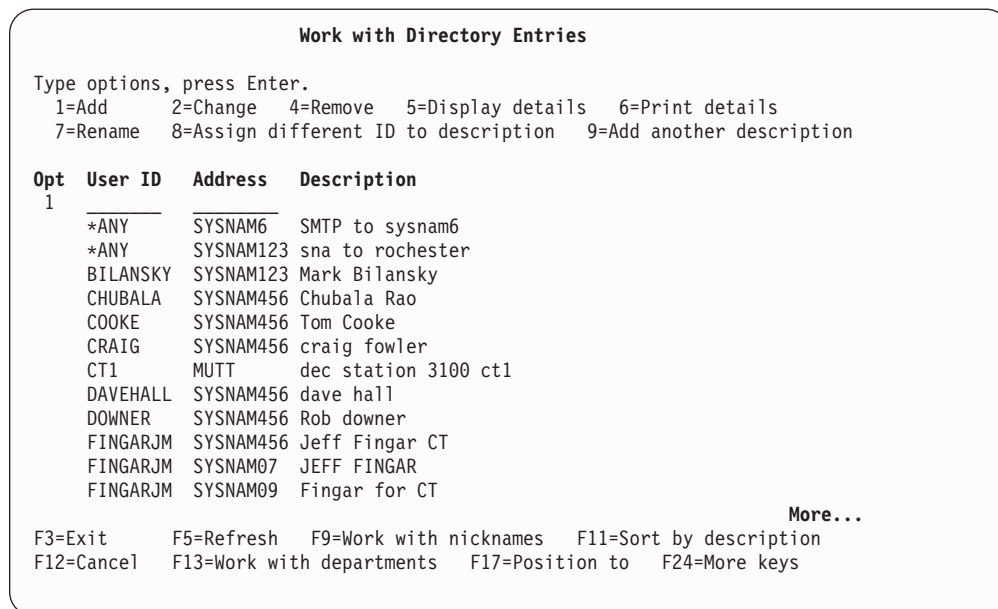


Figure 186. Work with Directory Entries (WRKDIRE) — Display 1

2. Type option 1 (Add) on the first line of the display.
3. Press the Enter key to see the Add Directory Entry display.

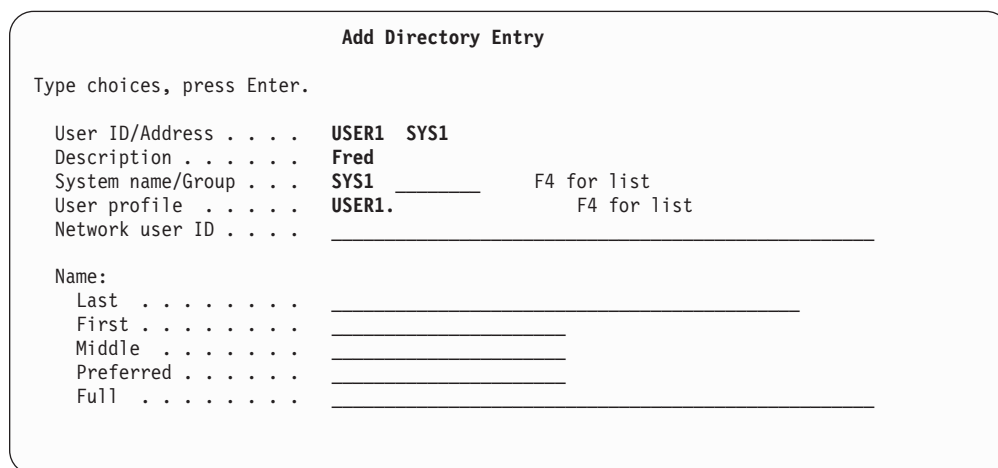


Figure 187. Add Directory Entry (ADDDIRE) — Display 1

4. Type the directory entry information. For POP mail users, pay particular attention to these fields (which are further down on the entry display):
 - *Mail service level.* Set this to 2 (System message store)
 - *Preferred address.* Set this to 3 (SMTP name). If you are using SNADS and not TCP/IP, set the *Preferred address* to 1 (User ID/Address).

```

Type choices, press Enter

Mail service level . . . 2
                                1=User index
                                2=System message store
                                3=Other mail service

For choice 3=Other mail service:
Field name . . . . . _____ F4 for list

Preferred address: . . . 3
                                1=User ID/Address
                                2=O/R Name
                                3=SMTP Name
                                4=Other preferred address
                                F4 for list

Address type . . . . . _____
For choice 4=Other preferred address:
Field name . . . . . _____ F4 for list
                                More...

F3=Exit  F4=Prompt  F5=Refresh  F12=Cancel  F18=Display location details
F19=Add name for SMTP

```

Figure 188. Add Directory Entry (ADDDIRE) — Display 2

5. If you specified 3 (SMTP) for *Preferred address*, follow the remaining steps for SMTP. If you specified 1 (User ID/Address) for *Preferred address*; skip to the end of this section.
6. Press F19 (Add name for SMTP). Fill in the *SMTP user ID* and *SMTP domain* fields. Press Enter.

```

Add Name for SMTP

Type choices, press Enter.

User ID . . . . . : USER1
Address . . . . . : SYS1

SMTP user ID . . . . . : USER1
SMTP domain . . . . . : SYS1.MyTown.MyCompany.COM

SMTP route . . . . . :

```

Figure 189. Add Name for SMTP Display

Important!:

IBM recommends that you use the same name for your user profile and your SMTP user ID. IBM also recommends that this name be the same as your ID in the system distribution directory.

The SMTP domain field must be in the format:

hostname.domain

where hostname is the AS/400 local host name and domain is the local domain name, both from the TCP/IP configuration. In the example Figure 189 on page 291, the host name is SYS1 and the TCP/IP domain name is MYTOWN.MYCOMPANY.COM.

7. When you see the message SMTP Table Updates Pending - Press Enter to Update, press Enter. This creates the directory entry and adds the TCP/IP address to the system alias table (which the POP server needs).

Changes to the system distribution directory take effect immediately; you do not need to restart either SMTP or the POP server.

There is a delay between the time when entries are changed in the system distribution directory and when they are reflected in the address book provided with Client Access for Windows 95/NT clients. This delay (or refresh interval) is set with the ADRBOOK parameter of the CHGPOPA command. See "Configuring POP for Client Access-Based Mail Users" on page 294.

POP Mailboxes

Once there is an entry in the system distribution directory for a POP mail user, the mailbox for that user is created automatically either the first time the client logs on successfully or when mail is received for the client.

The POP server uses a directory structure for mailboxes. These mailboxes are used as transient locations for storing the mail. The directory structure is set up by the system when the TCP/IP Connectivity Utilities for AS/400 LP is installed (at the same time the POP server is installed).

Setting Up Standard POP Mail Clients

Although each client product is different, you will normally need to provide the following information to set up standard (non-Client Access) POP mail clients:

- **User ID and host name.** This is typically in the form:
`<user ID>@<Host name>`

For example, user ID and host name might be:

```
jsmith@sysnam.mytown.company.com
```

In this example, the AS/400 host name is sysnam and the domain is mytown.company.com. jsmith is the AS/400 profile name for the user on the AS/400 system.

This is the user's E-mail address for receiving mail and should match the SMTP address that is entered for the system distribution directory SMTP address.

On some clients, the host IP address may have to be entered several times; once to specify the POP server's host for receiving mail, once to specify SMTP's host for sending mail, and once (sometimes more) as a way to identify the sender of mail to the recipients.

- **POP user or account name.** For the AS/400 POP mail server, this is the same as the AS/400 user profile name.
- **The password to use.** This must be the AS/400 user profile password.

Setting Up Client Access-Based Mail Clients

Set up for Client Access-based clients is minimal. When you install Client Access, click on the mail client and follow the directions for configuration. You will need to provide an AS/400 system name for your mail server.

Configuring the POP Server

The Change POP Attributes (CHGPOPA) command allows you to configure the POP server. You can get to the CHGPOPA command prompt display in either of the following ways:

- Enter the Configure TCP/IP command (CFGTCP) and select option 20 (Configure TCP/IP applications) and then select Option 16, Change POP server attributes
- Enter the Configure TCP/IP Applications (CFGTCPAPP) command from the command line, and select Option 16, Change POP server attributes
- Enter the Change POP Mail Server Attributes (CHGPOPA) command directly from the command line.

The following screen is displayed:

```
Change POP Server Attributes (CHGPOPA)

Type choices, press Enter.

Autostart servers. . . . . *YES *YES, *NO, *SAME
Number of initial servers. . . . 3 1-20, *SAME, *DFT
Inactivity time out. . . . . 600 10-65535 seconds, *SAME, *DFT
Message split size . . . . . 32 32-2048 kilobytes, *SAME, *DFT
MIME CCSID:
  Coded character set identifier 00819 *SAME, *DFT, 00819, 00912...
  When to use . . . . . *BESTFIT *SAME, *BESTFIT, *ALWAYS
Allow standard POP connection . *YES *SAME, *YES, *NO
Host server connection . *NONE *SAME, *NONE, *ALL, *IP...
      + for more values _____

Address book:
  Enabled: . . . . . *NO *SAME, *NO, *YES
  Refresh interval . . . . . _____ 1-65535 minutes, *NONE

F3=Exit F4=Prompt F5=Refresh F12=Cancel F13=How to use this display
F24=More keys
```

If you are going to use standard POP (TCP/IP) connections, specify *YES for the Allow standard POP connection (ALWSTDCNN) parameter. This is the normal socket interface used by standard POP clients, and is the default. Specify *NO if you will only be using the Client Access for Windows 95/NT clients.

To activate changes made in the configuration of the POP server, you must end and restart the following:

- The POP server
- The mail server framework

For more information on starting and stopping the mail server framework using the Start Mail Server Framework (STRMSF) and End Mail Server Framework (ENDMSF) commands, see *AnyMail/400 Mail Server Framework Support*.

Configuring POP for Client Access-Based Mail Users

There are two parameters on the CHGPOPA command that configure the POP server for MAPI-based clients:

HOSTSVRCNN

This parameter identifies the types of connection protocols to be supported for Client Access for Windows 95/NT clients connecting to the POP server. Use this parameter if you are using Client Access to connect to the POP server. You can specify any combination of the following protocols:

*IP

Support TCP/IP protocol for Client Access for OS/400 clients.

*IPX

Support IPX/SPX protocol for Client Access for OS/400 clients.

*SNA

Support SNA protocol for AS/400 Client Access clients. If you are using the SNA protocol, see "Setting the Number of SNA Servers" on page 295.

You can also specify *ALL (for all three protocols) or *NONE. The default is *NONE.

ADRBOOK

This parameter enables address book support for the client-based IBM AS/400 MAPI address book service provider, which is part of Client Access for Windows 95/NT. There are two elements: **Enable Address Book Support** and **Refresh interval**.

Set **Enable Address Book Support** to *YES. If you set this parameter element to *NO, clients will still be able to use POP server mail support, but they will not be able to access address book data.

When you set ADRBOOK to *YES, the POP server builds and maintains an address book cache. This is described in "AS/400 Address Book" on page 311.

For performance reasons, do not enable address book support unless you plan to use it.

The **Refresh interval** parameter element specifies, in minutes, how often you want the POP server to check to see if the address book cache is current, and if not, to refresh it from the AS/400 system distribution directory. If you want the POP address book to be refreshed only when the POP server starts or is restarted, specify *NONE for this parameter element. The default is 60 minutes.

The refresh interval is a trade-off between timely availability of changes to the system distribution directory, and processor utilization. You may want to refresh large address books less frequently because of the processor time required to do a refresh. Small address books can be refreshed more

frequently without greatly affecting processor utilization. The interval you choose should be based on your own situation, and the size of your address books.

Regardless of what the refresh interval is set to, if there have been no changes to the system distribution directory since the last time the address book cache was refreshed, a refresh is not performed. The refresh interval specifies how often the POP server checks to see if the cache is still current; if it is not current, it is refreshed.

If you plan to use only the Client Access MAPI-based clients, consider setting the Allow standard connection (ALWSTDCNN) parameter of CHGPOPA to *NO in order to save resources. This parameter defaults to *YES, which specifies the normal socket interface used by standard clients.

Removing POP Mail Users from the System

If you sign on to the client, the best way to remove users is to:

1. Sign on
2. Remove the user from the system distribution directory (this prevents any more mail from being delivered)
3. Connect a mail client as the user to be removed and receive any mail left in the mailbox
4. Delete the user's profile from the AS/400 system

Setting the Number of SNA Servers

The number of servers (NBRSVR) parameter of the CHGPOPA command determines how many POP server jobs to start when the POP server is started. The number of SNA servers is not affected by this parameter.

The number of SNA servers is determined by prestart job configuration, and set on the Change Prestart Job Entry (CHGPJE) command. You can accept the defaults on this command, or change the number of SNA servers. To change the number of SNA servers, use these parameter values:

- For the qualified subsystem name (SBSD), specify QSYSWRK.
- For the program name (PGM), specify library QTCP and program name QTMMSRVR.

Use these parameters to set the number of SNA servers:

- INLJOBS — the initial number of prestart jobs to start for the subsystem
- THRESHOLD — the minimum number of prestart jobs that must be available before additional prestart jobs are started
- ADLJOBS — the additional number of prestart jobs to start when the number of prestart jobs drops below the value of the THRESHOLD parameter
- MAXJOBS — the maximum number of prestart jobs that can be active at the same time for this prestart entry.

Stop and restart the POP servers for any changes to take effect.

Starting the POP Server

To use the command line interface: you can start the POP server in several ways:

- Use the STRTCPSVR command with the SERVER attribute set to *POP:
STRTCPSVR SERVER(*POP)
- Use the Change POP Mail Server Attributes (CHGPOPA) command to restart the POP server whenever the Start TCP/IP Servers (STRTCPSVR) command is run.
- Use the AUTOSTART option of the Start TCP/IP (STRTCP) command.

To use the Operations Navigator to start the POP server:

- Follow the path Network\Servers\TCP/IP
- Right-mouse click the POP server
- Select Start

Ending the POP Server

To use the command line interface to end the POP server: enter the TCP/IP Server (ENDTCPSVR) command with the server attribute set to *POP:

```
ENDTCPSVR SERVER(*POP)
```

You can also end the POP server in the following ways:

- Enter the ENDTCPSVR command without parameters (and all servers are stopped)
- Run the End TCP/IP (ENDTCP) command.

To use the Operations Navigator to stop the POP server:

- Follow the path Network\Servers\TCP/IP
- Right-mouse click the POP server
- Select Stop

Supported POP Verbs

The client software uses commands called *verbs* to communicate with the POP server. These commands are defined in RFC 1725. The AS/400 POP server supports the following verbs:

Verb and parameters	Description
USER <id>	Pass user ID
PASS <password>	Password
STAT	Query mailbox
LIST <opt msg #>	Query message statistics
RETR <msg #>	Retrieve message
DELE <msg #>	Delete message
RSET	Reset message delete status
TOP <msg #> <lines>	Retrieve message header and data
UIDL <opt msg #>	Get message unique ID listing
NOOP	No operation
QUIT	Quit client session

How the POP Server Uses the Mail Server Framework

The POP server uses snap-ins to perform POP server mail functions. **Snap-ins** are user exit programs that are called by the mail server framework. For more information about snap-ins and mail server framework processing, see *AnyMail/400 Mail Server Framework Support*.

The POP mail server uses the following snap-ins:

- Address Resolution
- Envelope Processing
- Attachment Conversion
- Local Delivery
- Non-Delivery
- Message Forwarding
- Attachment Management

Exchanging Mail with OfficeVision

Configuring Both POP and SMTP

For TCP/IP mail to be received correctly by OfficeVision or processed by SNADS, both SMTP and the POP server must be correctly configured. This allows SMTP to process simple SMTP mail and the POP server to process MIME-formatted mail. For mail sent to or from OfficeVision you may want to alter the following SMTP attributes (using the CHGSMTPA command):

- User ID delimiter (USRIDDELIM)
- Coded character set identifier (CCSID)
- Outgoing and incoming mapping tables (TBLSMTPOUT, TBLSMTPIN)

POP attributes that could affect mail to or from OfficeVision (which you can change with the CHGPOPA command) include:

- Message split size (MSGSPPLIT)
- MIME CCSID conversion parameters (MIMECCSID).

Using *ANY Support with the POP Server

You can use *ANY support when sending mail between two systems to avoid entering information about every mail recipient.

A POP client using SMTP can send mail to any remote SMTP user without having a directory entry (directory entries are only required on systems where you receive the mail). However, an OfficeVision (SNADS) user must have a directory entry on the sending system for every person he wants to send to (local or remote).

To make this easier, and to reduce the number of entries on the sending system, a SNADS user can enter an *ANY for the entire remote system. An entry where the User ID is *ANY and the Address is SYSTEMA allows OfficeVision users to send mail to anyone with an address of SYSTEMA, for example:

```
CAROL  SYSTEMA
```

If the recommended connection between two systems is actually an SMTP connection instead of a SNADS connection, you can still use the *ANY entry by adding an SMTP address to it. The SMTP user ID is left blank and the SMTP domain is filled in with the host.domain information for SYSTEMA, for example:

```
SYSTEMA.MYTOWN.MYCOMPANY.COM
```

To tell OS/400 to use SMTP instead of SNADS to deliver mail between the two systems, set the *Mail service level* to 2 (System message store) and set *Preferred address* to 3 (SMTP name).

For more information on *ANY support, see *SNA Distribution Services*, SC41-5410-01 .

A Client Access-based POP mail user can also send mail to an OS/400 distribution list.

MIME Mail Sent To OfficeVision

When a POP client sends mail to another POP client, the note is delivered without any conversions or transformations. The content of a POP-to-POP message is not affected by the POP server.

When mail is sent from POP to OfficeVision clients, the MIME parser snap-in converts MIME notes to OfficeVision/400 notes and documents. The snap-in converts any MIME text attachments into a single OfficeVision note and converts any binary attachments into documents with a type of IBM Personal Computer file. References to the binary attachments are noted in the OfficeVision Text Note in the same order in which they were encountered in the original MIME note.

While OfficeVision/400 is not MIME-compliant, the MIME conversion that occurs in the mail server framework allows users to view the text portions of MIME notes. To control the size and margins of lines so that lines do not appear truncated when viewed in OfficeVision/400, see "Long Line Conversion" on page 299, later in this section. OfficeVision users may also be able to view the binary portions of the MIME notes if they are using a graphical user interface such as IBM Current-OfficeVision/400. With this interface you can download the documents and start a handler to process the file. The extension of the file name determines which handler is called.

The POP server also preserves the original MIME type and subtype of the binary attachment. It tries to derive the file name of the original document. This is useful because many mail clients use the extension of the document as a means of "typing" the file and starting the proper handler for that attachment when the icon for the attachment is clicked. This derived file name is also used as the Subject after it has been enclosed by parentheses. The graphical OfficeVision interface called Current-OfficeVision/400 uses the file name to name downloaded files for processing at the workstation level.

Because OfficeVision is not MIME-compliant, no attachment icons are displayed and OfficeVision does *not* attempt to process the binary documents. If you try to view these documents from the mail menu of OfficeVision, OfficeVision displays a message saying that the document cannot be viewed. The binary attachments can be forwarded to MIME-compliant mail users who can display them. They can also be copied to local folders and accessed by using AS/400 Client Access network drives.

Long Line Conversion

In order to make ASCII client mail more readable in OfficeVision environments, the AS/400 Mail Server Framework is able to modify messages as they are converted from ASCII/MIME to EBCDIC/FFT, in Version 4 Release 2. The two modifications allow you to do the following:

1. Split long lines into two or more shorter lines.
2. Add FFT commands to messages, overriding parameters in the OfficeVision text profile active for the recipient of the new mail. With these FFT commands, wide messages can be viewed without changing OfficeVision text profiles.

The steps that are outlined below will enable either of the two modifications to occur:

1. Create a character data area that is named QUSRSYS/QMAILFMT. For example:

```
CRTDTAARA DTAARA(QUSRSYS/QMAILFMT) TYPE(*CHAR) LEN(100) VALUE('73/55/0')
```

In this example, length of 100 on the CRTDTAARA (Create Data Area) is arbitrary, but it should be sufficient for any combination of values to be put in the data area. All other parameters, except for the value, should be typed exactly as shown.

2. Change the object owner to QSYS. For example:

```
CHGOBJOWN OBJ(QUSRSYS/QMAILFMT) OBJTYPE(*DTAARA) NEWOWN(QSYS)
```
3. Grant the proper authority to the data area. For example:

```
GRTOBJAUT OBJ(QUSRSYS/QMAILFMT) OBJTYPE(*DTAARA) USER(PUBLIC) AUT(*ALL)
```
4. Stop and restart the Mail Server Framework by using the ENDMSF and STRMSF commands.

If you choose to make changes to the original modifications that you created, for subsequent uses, follow these steps:

1. change the value in the QUSRSYS/QMAILFMT data area. For example:

```
CHGDTAARA DTAARA(QUSRSYS/QMAILFMT) VALUE('80/55/1')
```
2. stop and restart the Mail Server Framework by using the ENDMSF and STRMSF commands.

Data Area Values

The value in the QMAILFMT data area can contain up to eight numbers that are defined as follows:

```
LL/PL/FFF/LM/RM/GFID/FWD/FA
```

Note: A forward slash, /, separates each field. A double forward slash, //, is an empty field, and defaults will be used. A space in any position ends the parameter string. Any numbers or characters after a space will be ignored.

- LL Line Length, from 1 to 255, default is 255.
The maximum number of characters in a line. A long line will be split at a space character preceding this count or at this count for lines without a space. Tab characters count 1 character.
- PL Page Length, from 1 to 32752, default is 32752.
The maximum number of lines on each page. If more than this number of lines are received without a form feed, one will be inserted at this point.

FFF Further Formatting Flag, 1 for yes, 0 for no, default is 0. If this flag is set to 1 (one), two FFT commands, SHM (Set Horizontal Margins) and SFG (Set FID through GFID) will be inserted in all messages converted from ASCII to EBCDIC. If this flag is set to 0 (zero), SHM and SFG will not be inserted and the remaining parameters are ignored.

NOTE: The SHM command is used to set two values: LM (Left Margin) and RM (Right Margin). The SFG command is used to set three values: global font identification (GFID), font width (FWD), and font attributes (FA)

LM Left Margin, from 0 to 32767, default is 1. Left margin value to be inserted in the SHM FFT command, in units of 1/1440 of an inch. A value of 0 will cause the left margin in the users text profile to be used. A value of 0 will cause the left margin in the users text profile to be used. A value of 1 represents the left edge of the screen. Ignored if FF is 0.

RM Right margin, from 0 to 32767, default 30480. Right margin value to be inserted in the SHM FFT command, in units of 1/1440 of an inch. A value of 0 will cause the right margin in the users text profile to be used. Ignored if FFF is 0.

GFID Global font ID, from 1 to 65534, default is 85. Font ID to be inserted into the SFG FFT command. 85 is a Courier 12-pitch font. Ignored if FFF is 0.

FWD Font width, from 1 to 1440, default is 120. Font width to be inserted into the SFG FFT command. 120 represents 120/1440, or 1/12 of an inch. Ignored if FFF is 0.

FA Font attributes, 1 for fixed or 2 for proportional, default is 1. Font attributes to be inserted into the SFG FFF command. Ignored if FF is 0.

The following examples include some values to consider for the QMAILFMT data area:

'73/55/0'

Lines over 73 characters long will be split into multiple lines so that all text can be viewed with the default OfficeVision text profile. Page breaks will be added every 55 lines, and the users OV/400 text profile will not be overridden.

'80/55/1'

Lines over 80 characters long will be split into multiple lines. This is a good choice if cc:Mail for the Internet or Lotus Mail is widely used, since those clients send lines whose maximum length are 80 characters. The right-most few characters of 80-character lines would not be visible with the default OfficeVision text profile. But the default FFT commands inserted in the message override the users text profile and the message can be scrolled to the right to view all data.

'//1'

All default values are used, but insert FFT commands in the

message. This is useful if some application on your system needs long lines to be kept intact. OfficeVision users will be able to scroll to the right to see parts of a message which would otherwise be hidden.

'255/32752/0/1/30480/85/120/1'
These are the default values.

'////////' or '///' or ' '
No number means use all default values.

Note: Note that the MAILFMT settings will affect all mail that comes from ASCII into EBCDIC.

Calculating FFT Values

For many mail environments, it will be necessary to calculate values for LM, RM, GFID, FWD, and FA. The default values allow mail of the maximum width (255 characters) to be viewed.

If you choose to set the margins to specific columns, you will need the following information:

- left column number
- right column number
- pitch of the font (in characters per inch)

You can calculate values for the margins, LM and RM, using this formula:

$LM = (\text{left column number} - 1) / (\text{characters per inch}) * 1440$
 $RM = (\text{right column number} - 1) / (\text{characters per inch}) * 1440$

Exception: use a margin of 1, not 0, for column 1.

For more examples of calculating values for margins, refer to the three examples below:

Example 1:

Left column number = 1
Right column number = 255
GFID = 85, Courier 12
Font pitch = 12 characters per inch

LM = 1
RM = $((255-1)/12*1440 = 30480$
FWD = $1440/12 = 120$

Assuming 55 lines /page, the QMAILFMT value would be:
'255/55/1/1/30480/85/120/1'

Example 2:

The default OV/400 Systems text profile has margins set at columns 19 and 91 for a line length of 73 characters. To set up the equivalent mail formatting using the Courier 10 font below, do the following:

Left column number = 19
Right column number = 91
GFID = 11, Courier 10
Font pitch = 10 characters per inch

LM = ((10-1)/10)*1440 = 1.8*1440 = 2592
RM = ((91-1)/10)*1440 = 9.0*1440 = 12960
FWD = 1440/10 = 144

Assuming 55 lines /page, the QMAILFMT value would be:
'73/55/1/2592/12960/11/144/1'

Example 3:

Left column number = 1
Right column number = 132
GFID = 223, Courier 15
Font pitch = 15 characters per inch

LM = 1
RM = ((132-1)/15)*1440 = 12576
FWD = 1440/15 = 96

Assuming 55 lines/page, the QMAILFMT value would be:
'132/55/1/1/12576/223/96/1'

The next three examples show Set FID through GFID (SFG) parameters:

	FONT STYLE	GFID	FWD	FA	COMMENTS
a).	Courier 10	11	144	1	Alias Courier 72. 10 ch/in.
b).	Courier 12	85	120	1	12 ch/in.
c).	Courier 15	223	96	1	15 ch/in.

For FWD = 10 ch/in set it to 144.
12 ch/in set it to 120.
15 ch/in set it to 96.

MIME Content Types

Standard Internet text notes consist of a general header and a text body. MIME notes, however, can contain multiple parts, which allows multimedia attachments to be included with the text.

If the general header contains a content type of *Multipart/Mixed*, one or more attachments follow. There are beginning and ending boundaries for each attachment. The boundary identifier is set on the *boundary=* parameter that follows the "Content-Type" header tag. See Figure 190 on page 303 for an example of a multipart MIME note. In this example, each part has a content type, and that each text content type can optionally have a character set (*charset*) defined.

```

From @SYSNAM6.CITY.COMPANY.COM:popct08@SYSNAM6.city.company.com Wed Jan 10
11:33:18 1996
Return-Path: <@SYSNAM6.CITY.COMPANY.COM:popct08@SYSNAM6.city.company.com>
Received: from SYSNAM6.city.company.com by fakeps2.city.company.com (COMPANY OS/2 SENDMAIL
VERSION 1.3.2)/1.0)
id AA0329; Wed, 10 Jan 96 11:33:18 -0500
Date: Wed, 10 Jan 96 11:33:18 -0500
Message-Id: <9601101633.AA0329@fakeps2.city.company.com>
Received: from endmail9 by SYSNAM6.CITY.COMPANY.
(IBM OS/400 SMTP V03R02M00) with TCP; Wed, 10 Jan 1996
10:23:42 +0000.
X-Sender: popct08@SYSNAM6.city.ibm.com (Unverified)
X-Mailer: Windows Eudora Pro Version 2.1.2
Mime-Version: 1.0
Content-Type: multipart/mixed; boundary="====_821301929=="
To: fake@fakeps2.city.company.com
From: endmail9 <popct08@SYSNAM6.city.company.com>
Subject: eudora attachments
X-Attachments: C:\EUDORA\ARGYLE.BMP;
-----_821301929==
Content-Type: text/plain; charset="us-ascii"

```

An example of using Eudora to send a text and bitmap.

```

-----_821301929==
Content-Type: application/octet-stream; name="ARGYLE.BMP";
x-mac-type="424D5070"; x-mac-creator="4A565752"
Content-Transfer-Encoding: base64
Content-Disposition: attachment; filename="ARGYLE.BMP"

```

```

Qk12AgAAAAAAAAHYAAAAoAAAAIAAAACAAAAABAAQAAAAAAAAACAAAAAAAAAAAAAAAAAAAAQAAAAAAAA
AAAAGAAAgAAAAICAAIAAAACAAIAAgIAAAICAgADAwMAAAD/AA/AAAA//8A/wAAAP8A/wD//wAA
///AE1ERERERERERERERERE1E1ERERERERsZERERERETURE1EREREREGxsZERERERNRERE1ERE
REbGxsZERERE1ERERE1ERERSbGxsZERETURERERE1ERGxsGxsZERNRERERERE1EbGxsGxsZE1E
RERERERE1sbGxsGxsBUREREREREREG1sbGxsGxtZEREREREREbG1sbGxsG1sZERERERERSbG1s
bGxsWxsZEREREREGxsG1sbGxtbGxsZEREREbGxsG1sbG1sbGxsZERERSbGxsG1sbWxsGxsZE
RGxsGxsG1tbGxsGxsZEBGxsGxsG1sbGxsGxsZEBGxsGxsG1sbGxsG1sbGxsG1sbGxsG1sbGxsG
1sbGxsG1sbGxsG1sbGxsG1sbGxsG1sbGxsG1sbGxsG1sbGxsG1sbGxsG1sbGxsG1sbGxsG1sbG
REbG1sbGxsG1sZEREREREREbWxsGxsG1kRERERERERNbGxsG1sbG1ERERERERE1EbGxsGxs
ZE1ERERERETUREbGxsG1sbG1EREREREREREbGxsG1sbG1ERERERERE1ERERE1ERERE1ERERE1ERE
TUREREbGxsG1sbG1ERERERERE1EREREREREbG1sbG1ERERERERE1ERERERERE3URERERERERERE
RERE
-----_821301929== --

```

Figure 190. Example of a Multipart MIME Note

Supported Content Types of the POP Server

The POP server can handle almost any type. If the part does not have a type of Text with a subtype of plain or enriched or a type of Message with a subtype of RFC822, the part is stored as a binary PC file. A notation is made in the OfficeVision text note that is generated when a MIME note is converted.

What about the Type/Subtype of Message/RFC822? This is a commonly-used method of embedding a note within a note that is often used when forwarding a note. The POP server handles this type by prefixing any text and binary file notations within the OfficeVision note with the greater-than (>) character. The number of prefixed greater-than characters indicates the embed level of the RFC822 message.

The Type/Subtype of Message/RFC822 handles up to 10 levels of embeds. After the tenth embed, the note content is copied verbatim to the text OfficeVision note. The following shows a message with three levels of embeds:

```
Message-Id:<30F6D7CA.193F@SYSNAM8.city.ibm.com>
Date: Fri, 12 Jan 1996 16:51:06 -0500
From: Bob Smith <bsmith@SYSNAM8.city.company.com>
Mime-Version: 1.0
To: gbrett@SYSNAM7.city.company.com
Subject: [Fwd: [Fwd: [Fwd: Text and Image]]]
-----
3 level imbed
--
Bob Smith
Dept 250
>
>
>-----
> Sender: fake@SYSNAM8.city.company.com
> Message-Id:<30F6D747.64FC@SYSNAM8.city.company.com>
> Date: Fri, 12 Jan 1996 16:48:55 -0500
> From: Ed Bailey <ebailey@SYSNAM8.city.company.com>
> Mime-Version: 1.0
> To: bsmith@SYSNAM8.city.company.com > Subject: [Fwd: [Fwd: Text and Image]]
>
>
>-----
>Next level 1
>--
>Ed Bailey
>Dept 129
>>
>>
>>-----
>> Sender: fake@SYSNAM8.city.company.com
>> Message-Id:<30F6D471.784F@SYSNAM8.city.company.com>
>> Date: Fri, 12 Jan 1996 16:36:49 -0500
>> From: C. Kent <ckent@SYSNAM7.city.company.com>
>> Mime-Version: 1.0
>> To: ebailey@SYSNAM8.city.company.com
>> Subject: [Fwd: Text and Image]
>>
>>
>>-----
>>Next level 2
>>--
>>Clark Kent
>>Dailey Times
>>
>>
>>-----
>>> From: halljd@SYSNAM7.city.company.com
>>> Message-Id:<9601121417.AC0017@sonofzoo.city.company.com>
>>> Mime-Version: 1.0
>>> Date: Fri, 12 Jan 96 09:16:24 +0600
>>> To: ckent@SYSNAM7.city.company.com
>>> Subject: Text and Image
>>>
>>>
>>>-----
>>>Next level 3
>>> Some text and an embedded image.
>>>//-----
>>>// J. D. Hall
>>>// Dept 360
>>>
>>>
```

```
>>-----  
>>Type/Subtype: image/GIF  
>>>Description: (BRICK.BMP)  
>>>
```

How the File Name is Derived

During the POP (MIME)-note-to-OfficeVision-note conversion process, multimedia subparts are converted to OfficeVision/400 document PC files. These OfficeVision document PC files are binary representations of the particular multimedia subpart that they represent. When these OfficeVision document PC files are generated, a file name is derived to store in the Document Interchange Architecture (DIA) portion of the OfficeVision document. When the document is sent to a POP mail user, the file name is used for the content description and also as the *name=* parameter content for the content type. This allows MIME-compliant clients to use the file name to determine what type of object the attachment is and which handler to run against it.

A MIME-compliant POP client can set the file name in the following ways:

- Include the file name as the *filename=* parameter of the Content-Disposition header.
- Specify the file name on the *name=* parameter of the Content-Type header.
- Place the *filename.ext* in the Content-Description header.

The POP server tries to determine the file name by successively looking in these places. When a header is found, the MIME parser attempts to parse the file name. If a blank string is returned, the internal file name that is used in the conversion process is used. If a file name is parsed, then the *filename* portion of the *filename.ext* is limited to 8 characters and the *ext* is limited to 3 characters. This allows all file allocation table (FAT) file systems to process these files.

MIME Content Types

Table 26 on page 306 shows how each type and subtype of the primary header are handled. These are defined in the content-type header. These tables are for reference only, and show which MIME Content-Type headers the POP server supports.

Table 27 on page 306 shows how each type and subtype of the individual part headers are handled. These are defined in the Content-type header.

What Happens When You Send OfficeVision Mail to POP Clients

When you send an OfficeVision note to a POP user, the POP server sends the entire contents of the OfficeVision note and includes any headers that OfficeVision may have inserted. The character set that is used for the *charset=* parameter is determined by the value of the MIMECCSID *When_to_Use* parameter element of the Change POP Attributes (CHGPOPA) command. If this parameter is set to **ALWAYS*, the EBCDIC Coded Character Set Identifier (CCSID) is forced to the ASCII CCSID that is defined in the CHGPOPA menu. If the parameter is set to **BESTFIT*, the appropriate ASCII CCSID that produces the best fit based on the EBCDIC CCSID is used. The ASCII CCSID value is then used to look up the equivalent ISO-8859-x character set for the *charset* parameter.

See Figure 191 on page 307 for an example of a note generated by the AnyMail/400 framework to a POP user. In this example, the memo slips will always be in the default MIME charset US-ASCII. The contents of the note will be in the best fit or forced fit character set as determined by the POP3 Server's configuration. The text tags that are generated by OfficeVision in the content of the note will not be removed, hence you may see the subject and other tags twice.

Binary attachments are converted into a MIME part of a multipart/mixed note and the stored type and subtype are placed in the Content-Type header. A *name=* parameter is added to the Content-Type header with the stored *filename.ext*. This name is also used as the Content-Description.

Table 26. MIME Primary Heading Content Types

Type	Subtype	OfficeVision Counterpart	Notes
None	None	Note	This is a simple SMTP note that uses only RFC822 headings.
Text	Plain	Note	This is treated the same as SMTP simple note; however, a character set can be used.
Multipart	Mix	Note and possibly Document	Each part's type and subtype determines how it is handled. See Table 27.
Multipart	Parallel or Digest or Alternative or Extension- token	Note and Document	This combination cannot be parsed, its contents are spooled to an OfficeVision Document, PC file.
Message	RFC822	Note and possibly Document	This is usually an embedded note and could be a MIME with a content type of MultiPart Mixed. See the multipart explanation in this table.
Message	Partial	Note and possibly Document	This combination cannot be parsed. Its contents are spooled to an OfficeVision Document, PC file.
Message	External- body	Note and Document	This combination cannot be parsed. Its contents are spooled to an OfficeVision Document, PC file.

Table 27. Mapping MIME Note's Part Type and Subtypes to OfficeVision

Type	Subtype	OfficeVision Counterpart	Notes
None	None	Note or part of the OfficeVision note	Treated as simple note; assumes the character set is US-ASCII
Text	Plain	Note or part of the OfficeVision note	Same as a simple note, but the character set is honored if it is present as an attribute to the Content-type header.
Text	Enhanced	Note or part of the OfficeVision note	Same as Simple note, but the character set is honored if it is present as an attribute to the Content-type header. The text is passed as is. Enhanced tags will be present in text.
Image	Any subtype honored	OfficeVision Document PC file	Type and subtype are preserved along with the derived file name.
Audio	Any subtype honored	OfficeVision Document PC file	Type and subtype are preserved along with the derived file name.
Video	Any subtype honored	OfficeVision Document PC file	Type and subtype are preserved along with the derived file name.
Application	Any subtype honored	OfficeVision Document PC file	Type and subtype are preserved along with the derived file name.

Table 27. Mapping MIME Note's Part Type and Subtypes to OfficeVision (continued)

Type	Subtype	OfficeVision Counterpart	Notes
Message	RFC822		Appends to existing note and may add documents
Not defined in this table	Not defined in this table	OfficeVision Document PC file	MIME Parser does not handle these combinations and spools them to a PC file.

```

From FAKEOV@SYSNAM7.CITY.COMPANY.COM Thu Feb 01 07:54:03 1996
Return-Path: <FAKEOV@SYSNAM7.CITY.COMPANY.COM>
Received: from SYSNAM7.city.company.com by fakeps2.city.company.com (COMPANY OS/2 SENDMAIL VERSION 1.3.2)/1.0
id AA0202; Thu, 01 Feb 96 07:54:03 -0500
Message-Id: <9602011254.AA0202@fakeps2.city.company.com>
Mime-Version: 1.0
Date: Thu, 1 Feb 1996 07:55:16 +0000
Subject: This is the Subject Line
Reference: This is the Reference Line
Sensitivity: none
Priority: normal
Importance: high
From: FAKEOV@SYSNAM7.CITY.COMPANY.COM
To: fake@fakeps2.city.company.com
Content-Type: multipart/mixed;
  boundary="PART.BOUNDARY.1"

```

```

> THIS IS A MESSAGE IN 'MIME' FORMAT. Your mail reader does not support MIME.
> You may not be able to read some parts of this message.

```

```

--PART.BOUNDARY.1
Content-ID: <1_1>
Content-Type: text/plain

```

```

For your information
Here's a Memo Slip which I've attached.
--PART.BOUNDARY.1
Content-Type: text/plain; charset=ISO-8859-1
Content-Transfer-Encoding: quoted-printable

```

```

TO: FAKE FAKEPS2 Fakes ps2
FROM: FAKEOV SYSNAM7 FAKEOV
DATE: February 1, 1996
SUBJECT: This is the Subject Line
REFERENCE: This is the Reference Line

```

This is the main body of the note.

```

--PART.BOUNDARY.1--

```

Figure 191. Example of a MIME Note Going from OfficeVision/400 to a POP Mail User

Setting Up MIME Headers to Differentiate between Recipients

The Change Distribution Attributes (CHGDSTA) command changes the content of message services attributes (X.400 support) for mail distributions and OfficeVision/400. The Keep Recipient (KEEPRCP) parameter specifies which recipient information is stored and sent within each mail distribution. The setting of this parameter affects how the MIME headers get created for a note from OfficeVision.

In order for CC and BCC tags to show up in MIME headers (and client screens), you must set the KEEPRCP parameter to *ALL. BCC recipients are not shown regardless of the setting of this parameter because they are not intended to be. The TO and CC recipients will show up in the text of the OfficeVision note.

Sending MIME (POP Server) Mail across a SNADS Network

This section applies only to those situations where:

- You have AS/400 systems connected to each other across a SNADS network
- You are adding (or have added) POP mail clients
- You want these POP mail clients to be able to send MIME mail to each other between these systems without the mail being handled as OfficeVision mail.

When MIME mail is sent over AS/400 systems connected with SNADS, MIME text attachments are parsed and converted, and video, audio and image files are decoded and reformatted as PC files. Because of this processing, a single MIME message can be converted to many SNADS messages, and some information can be lost in the conversion. This happens even if a POP mail user is sending mail to another POP mail user across a network; SNADS assumes that the mail is OfficeVision mail and handles it accordingly.

SNADS tunneling provides an alternative; it allows POP users to send MIME mail between AS/400 systems across a SNADS network without the mail being converted to OfficeVision mail.

How SNADS Tunneling Works

Tunneling refers to a technique in which you send protocol traffic across another protocol or send data across a protocol that the data was not intended to be sent across. With SNADS tunneling, a MIME message is encapsulated as a SNADS object distribution data file. Because object distribution files are treated as binary or unformatted data, they are passed along as is; no parsing or transformation takes place.

How to Configure System Distribution Directory Entries for SNADS Tunneling

Configuring SNADS mail clients to use SNADS tunneling is very simple. All you need to do is set the *Address type* field in the system distribution directory entry for the mail clients or system receiving the mail to TNDENDGN. Since you will normally be using *ANY entries to transport mail to another system, changing this one field ensures that all MIME mail sent to users on that system (who do not have different instructions associated with their own individual entries in the system distribution directory) will be tunneled.

Although mail clients still need to be configured to receive mail, you do not need to do anything differently for these clients to receive mail that is tunneled through SNADS (SNA distribution services). Tunneling is determined by the directory entry for the system to which you are sending mail and only affects the way the mail is transported (not delivered). Mail recipients, as long as they are using V3R2 or V3R7 and later versions of the TCP/IP Connectivity Utilities for AS/400, are able to receive both OfficeVision and MIME mail (regardless of whether it is tunneled).

Example

You have two systems, SYSTEM1 and SYSTEM2, both with POP mail users. They are connected with SNADS. You want the MIME mail that POP users on SYSTEM1 are sending to POP mail users on SYSTEM2 to be tunneled.

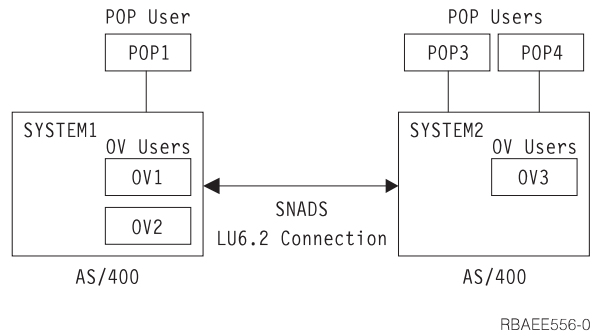


Figure 192. SNADS Network with OfficeVision and POP Mail Users

Assuming that you have a *ANY entry for all mail going to SYSTEM2, you need to change the *Address type* field in this entry.

1. Type WRKDIRE to display the Work with Directory Entries menu.

```

Work with Directory Entries

Type options, press Enter.
 1=Add      2=Change  4=Remove  5=Display details  6=Print details
 7=Rename   8=Assign different ID to description  9=Add another description

Opt  User ID  Address  Description
--  -
 1  AABOND   SYSNAM3  Bond, Alan
 2  *ANY     SYSTEM2  System2 Mail Users
 3  ACER     SYSNAM3  Acer, Rodney G. (Rod)
 4  ADAMSJA  SYSNAM1  Adams, Jane
 5  ALBERT   SYSNAM1  Schweitzer, Albert
 6  APPLETN  SYSNAM1  Appleton, John
 7  BONNER   SYSNAM3  Bonner, Heinz
 8  BRIGHT  SYSNAM1  Bright, Jeremiah (Jerry)
 9  BYERS    SYSNAM1  Byers, Andrea
10  CARLTON  SYSNAM3  Lewis, Carlton
11  CHUCK    SYSNAM2  Mortimer, Charles (Chuck)
12  DAVIS    SYSNAM1  Davis, Martin

More...
F3=Exit    F5=Refresh  F9=Work with nicknames  F11=Sort by description
  
```

Figure 193. Work with Directory Entries (WRKDIRE) Display — Display 2

2. Type option 2 (Change) next to the SYSTEM2 entry
3. Press the Enter key to see the Change Directory Entry display.

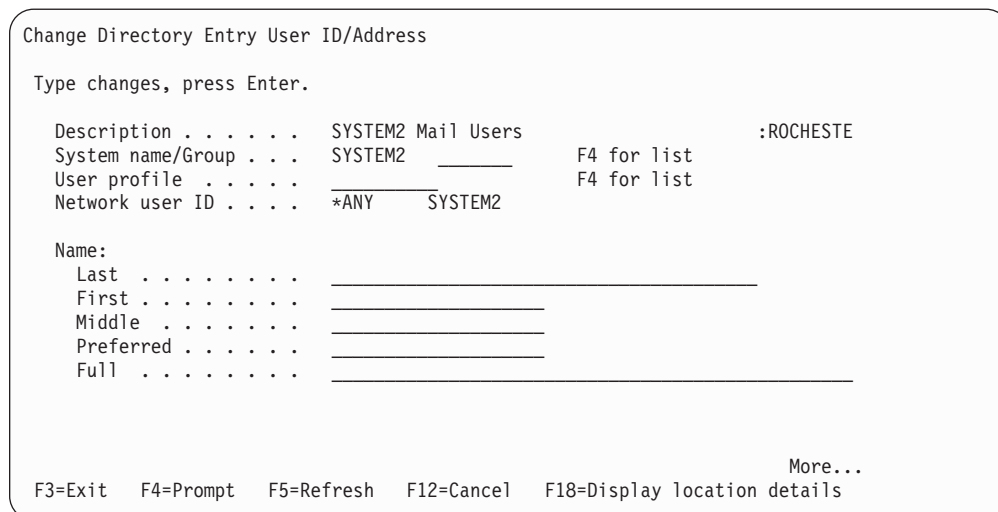


Figure 194. Change Directory Entry Display — Display 1

- Page down to the display that shows the *Preferred address* and *Address type* fields

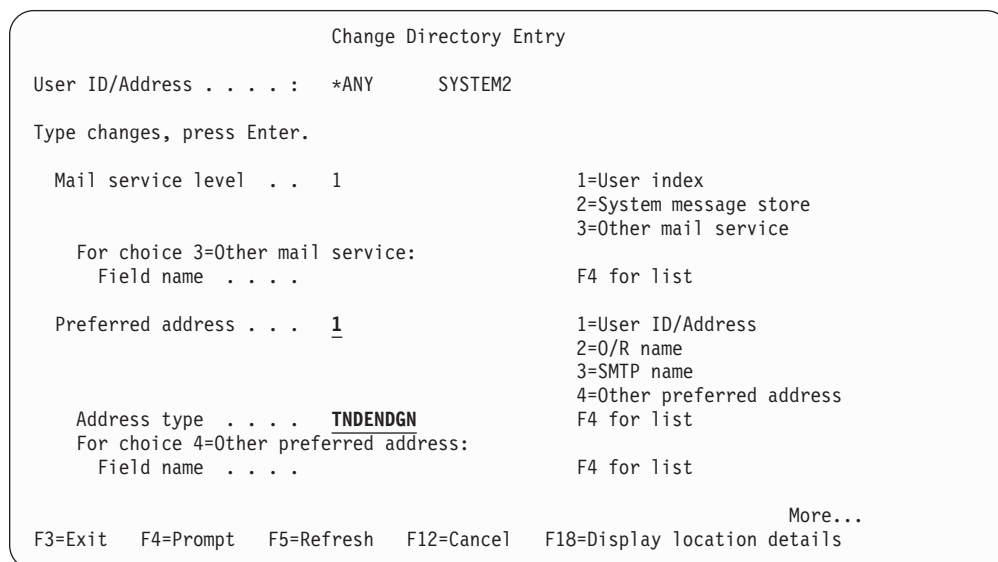


Figure 195. Change Directory Entry Display — Display 2

- Since this is a SNADS system, the *Preferred address* for this entry should already be set to 1 (User ID/Address).
- Set *Address type* to TNDENDGN.
- Press Enter to save your changes.

Address Types

Standard POP implementations can address mail only with Internet addresses. If an Internet address needs to be converted to a different type of address, the conversion is performed by a gateway somewhere in the network. (The gateway may be the AS/400 Mail Server Framework). Client Access-based mail clients, in

conjunction with the AS/400 POP server, have another option. They can address mail with several different types of addresses.

Table 28 shows each type of address supported by Client Access-based mail. The MAPI interface, implemented by Client Access-based mail service providers, allows any mail-enabled application to address mail with any of these types of addresses. The type names shown in the table are the actual MAPI address-type strings that Client Access-based mail registers with MAPI. The table simply shows the valid address types and their corresponding formats.

AS/400 Address Book

The POP Server and the Client Access-based MAPI address book provider also provide a public address book. Mail-enabled applications on the client can view entries in the address book or send mail to users listed in the address book. For example, the Microsoft Exchange client talks to a MAPI interface. Therefore, a Client Access user running Microsoft Exchange can select an entry from the address book and send mail to the user represented by that entry.

The addresses in the AS/400 address book are from the system distribution directory and distribution Lists on the AS/400. Table 29 on page 313 shows where data in the address book comes from. The address type of an AS/400 address book entry is determined by the preferred address in the system distribution directory. Distribution lists always have an address type of OFFICEVISION. The **Preferred address** in Table 28 gives the preferred address of each type of address in the AS/400 address book.

Table 28. MAPI Address Type Definitions

MAPI Address Type	Preferred Address	Format and Description
INTERNET	SMTP name	<p>Format: <userid>@<domain></p> <p>This is the standard internet format. From the system distribution directory, <userid> is the <i>SMTP user ID</i> field and <domain> is the <i>SMTP domain</i> field. For example:</p> <pre>system5.endicott.ibm.com aol.com gd1vm6</pre> <p><userid> and <domain> must be separated by a single '@' character, and blanks are not allowed within or between the parts. Leading and trailing blanks to the whole address should be tolerated and ignored.</p> <p>Examples:</p> <pre>mandydog@system1.endicott.ibm.com lisa@system5 joecaldwell@vnet.ibm.com</pre>

Table 28. MAPI Address Type Definitions (continued)

MAPI Address Type	Preferred Address	Format and Description
OFFICEVISION	User ID/Address (for individual directory entries), or List ID and List ID qualifier (for distribution lists)	<p>Format: <UUUUUUUU> <AAAAAAA></p> <p>This type is also called the “SNADS address” or “DEN/DGN” by some. From the system distribution directory, <UUUUUU> is the <i>User</i> field, and <AAAAAA> is the <i>Address</i> field. Both values can be a maximum of eight characters long (and can be shorter than eight characters). Neither <UUUUUU> nor <AAAAAA> can contain the blank character. They must be separated by at least one blank character. Leading and trailing blanks to the whole address should be tolerated and ignored.</p> <p>Examples:</p> <pre>MANDY SYSTEM1 LISA SYSTEM5 JAMIE GRADE5 ELYSE GRADE1 CALDWELJ SYSTEM2</pre>
AS400FAX	Other preferred address (FAXTELNR)	<p>Format: <facsimile-telephone-number></p> <p>Within the system distribution directory, this is considered one of the “Other” address types. (Set <i>Preferred address</i> to 4 (Other preferred address).) The actual<facsimile-telephone-number> used as the address is found in the system distribution directory <i>FAX telephone number</i>. The address is a “dialing sequence”, including access code sequences. It is expected to follow the rules for the Facsimile Support for OS/400 LPP <i>telephone-number</i>.¹ Leading and trailing blanks to the whole address should be tolerated or ignored.</p> <p>Examples:</p> <pre>7525421 9=16077525421 8+8525421 *70/18005551212</pre>
<p>Notes:</p> <p>1. The <i>telephone number</i>, made up of dialing and control codes, is described in the <i>Facsimile Support for the OS/400 Programmer's Guide and Reference</i>, SC41-0572. See the detailed description of the SNDFAX command. Also see the <i>Facsimile Support for OS/400 Installation Guide</i>, SC41-0570 for more information on creating FAX entries in the system distribution directory.</p>		

The entries described in Table 28 on page 311 are built into an **address book cache** that includes these address types and E-mail addresses as well as other information from the system distribution directory.

The Address Book Cache

The POP server does not read the system distribution directory every time a client requests something from it. Instead, a cache is built from the system distribution directory entries and distribution lists. The POP server uses this cache to retrieve address book data for clients.

The address book cache is built and maintained by the POP server when the ADRBOOK parameter is set to *YES. The Refresh interval element of the ADRBOOK parameter determines how often the address book is updated from the system distribution directory. (See “Configuring POP for Client Access-Based Mail Users” on page 294 for a description of the ADRBOOK parameter.)

Table 29. Data mapping from System Distribution Directory to POP Server Address Book Cache

Field in Address Book Cache	Fields in the System Distribution Directory
<display-name>	<p>Full name or Description</p> <p>If Full name is not blank, it is used. If Full name is blank, Description is used.</p> <p>For Distribution Lists, the Description field is always used (Distribution List entries do not have a Full name field).</p>
<address-type>	<p>For individual system distribution directory entries, use Preferred address to determine the type.</p> <p>The cache <address-type> field is filled in using the following rules:</p> <ul style="list-style-type: none"> • If Preferred address is “*USRID” (User ID/Address), use MAPI address type OFFICEVISION • If Preferred address is “*SMTP” (SMTP), use MAPI address type INTERNET • If Preferred address is “FAXTELNB” (considered an “Other” address type), use MAPI address type AS400FAX • If Preferred address is not one of the values above, the address type is not supported by AS/400 MAPI service providers and the entry is <u>not</u> put into the address book cache. <p>For AS/400 Distribution lists, <address-type> is OFFICEVISION.</p>
<email-address>	<p>(User ID -and- Address) or (SMTP user ID -and- SMTP domain) or FAX telephone number</p> <p>The cache <email-address> field is filled using the following rules:</p> <ul style="list-style-type: none"> • If the address book cache <address-type> is now “OFFICEVISION”, concatenate the following: <ol style="list-style-type: none"> 1. The 8-character system distribution directory User ID (including trailing blanks) for individual SDD entries, or List ID for AS/400 Distribution Lists. 2. A single blank 3. The Address (trailing blanks not required) for individual system distribution directory entries, and List ID qualifier for AS/400 Distribution Lists. • If the address book cache <address-type> is now “INTERNET”, concatenate the following: <ol style="list-style-type: none"> 1. SMTP user ID (without trailing blanks) 2. A single “@” character 3. The SMTP domain (trailing blanks not required). • If the address book cache <address-type> is now “AS400FAX”, use system distribution directory FAX telephone number (trailing blanks not required).
<comment>	No system distribution directory data is currently being extracted for this field.

ASCII-EBCDIC Conversion and National Language Support

This topic describes how the POP server handles the conversion of mail from ASCII to EBCDIC and from EBCDIC to ASCII. This conversion is part of national language support (NLS).

EBCDIC-to-ASCII Conversion

Figure 196 shows how the POP server handles mail going from EBCDIC (such as OfficeVision mail or control language interface) to ASCII (such as Internet or POP mail).

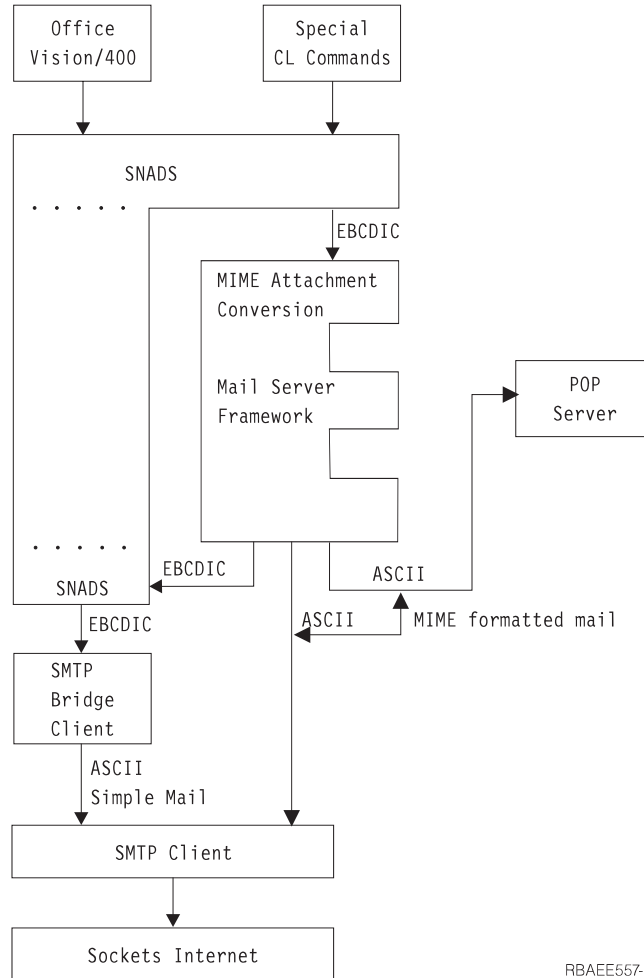


Figure 196. EBCDIC-to-ASCII Conversion

There are two paths by which text is converted from OfficeVision format (EBCDIC) to Internet format (ASCII); the SMTP bridge and the Mail Server Framework. Before the introduction of MIME support and the POP server in V3R2 and V3R7, all mail which needed conversion went through the SMTP bridge. Mail is converted in the SMTP bridge using the SMTP configuration information (CHGSMTPA). Mail is converted in the framework using the POP configuration information (CHGPOPA).

To configure your outbound mail to go through the SMTP bridge, set the following fields on the Add Directory Entry (ADDDIRE) command of the recipient of the mail:

1. Set *System name* to TCP/IP as shown in Figure 197 on page 315:

```

                                Add Directory Entry

Type choices, press Enter.

User ID/Address . . . . USER1 SYS1
Description . . . . . Fred
System name/Group . . . TCPIP _____ F4 for list
User profile . . . . . USER1. _____ F4 for list
Network user ID . . . . _____

Name:
Last . . . . . _____
First . . . . . _____
Middle . . . . . _____
Preferred . . . . . _____
Full . . . . . _____

```

Figure 197. Add Directory Entry (ADDDIRE) — Display 3

2. Set *Mail service level* to 1 (User index) and *Preferred address* to 1 (User ID/Address) as shown.

```

Type choices, press Enter

Mail service level . . . 1                1=User index
                                           2=System message store
                                           3=Other mail service

For choice 3=Other mail service:
Field name . . . . . _____ F4 for list

Preferred address: . . . 1                1=User ID/Address
                                           2=O/R Name
                                           3=SMTP Name
                                           4=Other preferred address
                                           F4 for list

Address type . . . . . _____
For choice 4=Other preferred address:
Field name . . . . . _____ F4 for list

More...
F3=Exit F4=Prompt F5=Refresh F12=Cancel F18=Display location details
F19=Add name for SMTP

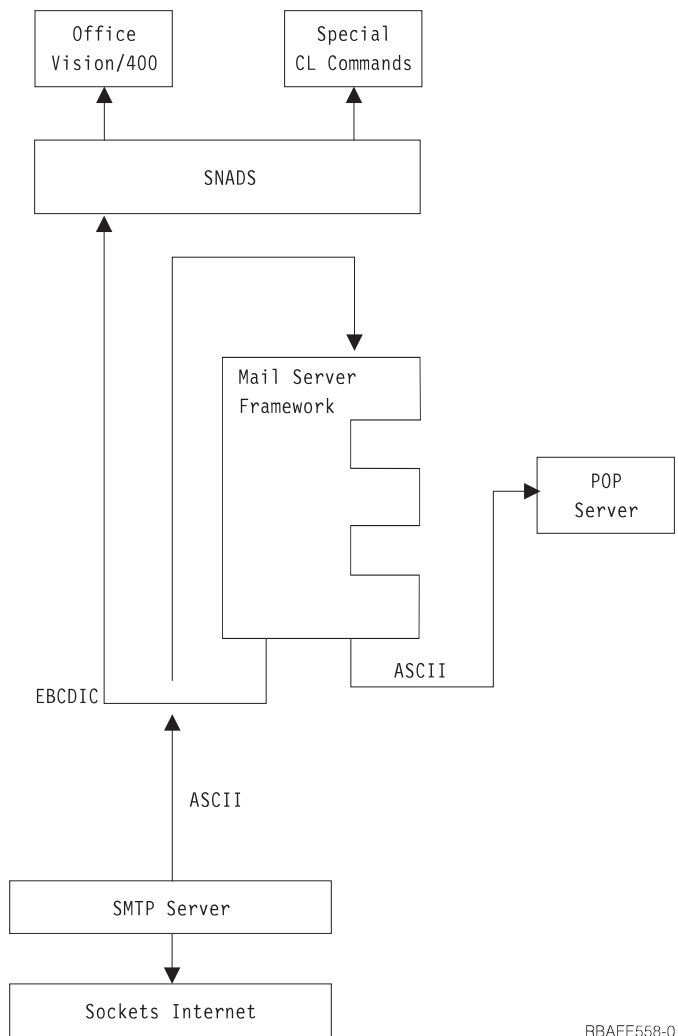
```

Figure 198. Add Directory Entry (ADDDIRE) — Display 4

If the text is converted to a code page other than US_ASCII or Latin_1 (ISO-8859-1), the OfficeVision/400 non-convert option for FFT must be set. Otherwise the results are unpredictable.

ASCII-to-EBCDIC Conversion

Figure 199 on page 316 shows how the POP server handles mail going from ASCII to POP mail or EBCDIC.



RBAEE558-0

Figure 199. ASCII-to-EBCDIC Conversion

No conversion is done on incoming mail from SMTP if it is delivered to a POP mailbox.

The path to SNADS does ASCII-to-EBCDIC conversion and DIA/FFT formatting. Simple SMTP is converted using the SMTP configuration information (CHGSMTPA). MIME mail is converted using the POP configuration information (CHGPOPA).

All TCP/IP mail sent to OfficeVision/400, both MIME and non-MIME, must be converted from ASCII to EBCDIC. For this conversion to work in any country both SMTP and the POP server must be configured.

What are CCSIDs

ISO standards and Internet RFCs define a list of graphic symbols or characters used to represent the written language of one or more countries. This list is called a character set and is identified by a number (the Coded Character Set Identifier, or CCSID). OS/400 supports a large number of character sets.

Character sets, when given binary values for each character, are called code pages. Each language supported by OS/400 has a character set and code page.

For the United States, the character set is 697 and the code page is 37 (CCSID 37). This character set and code page combination can be converted into any other code page that uses the same character set. The ISO language standard 8859-1 uses or defines the same characters as character set 697, so United States English can be converted into ISO-8859-1 and back again without the loss of information. If a character set for a country is the same as, or is a subset of the MIME-standard character set, data can be converted to the MIME code page without the loss of information.

For more information on national language support and what is supported by OS/400, see *National Language Support* book.

CCSIDs Supported by the POP Server

The MIME coded character set identifier configuration parameter (MIMECCSID) on the Change POP Attributes (CHGPOPA) command determines which CCSID to use for translation, and under what conditions to use it. Once you specify a CCSID, you can choose to use it all the time (*ALWAYS) , or only if a better fit cannot be found (*BESTFIT).

The CCSIDs supported for conversion are MIME- or Internet-defined standard character sets:

Table 30. Supported MIME Standard Code Pages

MIME Standard	Name	ASCII CCSID	Character Set	EBCDIC CCSID
US-ASCII	US English	00367	103	00500
ISO-8859-1	Latin-1	00819	697	00500
ISO-8859-2	Latin-2	00912	959	00870
ISO-8859-5	Cyrillic	00915	1150	01025
ISO-8859-6	Arabic	1089	1271	420
ISO-8859-7	Greek	00813	925	00875
ISO-8859-8	Hebrew	00916	941	00424
ISO-8859-9	Latin-5	00920	1152	01026
ISO-2022-JP	Japan MBCS	05052	1064,1062,1121,1120	05026

Chapter 10. Workstation Gateway Server

The workstation gateway server (WSG) is a TCP/IP application that transforms AS/400 5250 data streams to Hypertext Markup Language (HTML) for dynamic display on Web browsers. This allows you to run AS/400 applications from any workstation that has a Web browser.

The workstation gateway server provides capabilities that are similar to those when you use the character interface to an AS/400 system or when you use the Graphical Access function of Client Access.

To access an AS/400 system from a Web browser, specify the URL of the workstation gateway server.

Without a workstation gateway server, your applications send out 5250 data streams that an emulator converts to text for display on a workstation. With a workstation gateway server, your applications send out the same 5250 data streams, but the AS/400 system converts the data stream for display by Web browsers. The AS/400 system converts the 5250 data to HTML, which the Web browsers use for their displays. Figure 200 shows you how the AS/400 system does this.

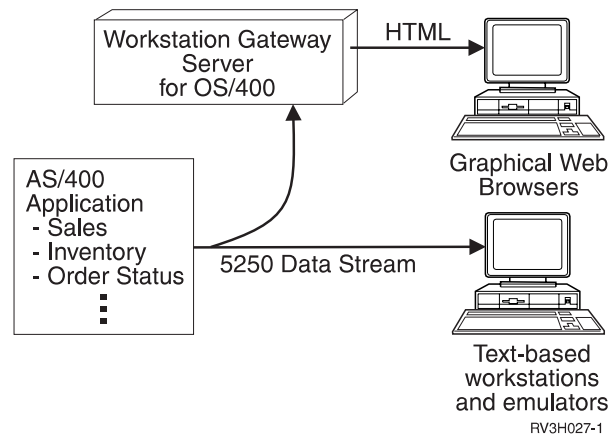


Figure 200. AS/400 Support of workstation gateway server

Your applications continue to work, unchanged, but you can add graphics to them if you use the workstation gateway server to access them. The book *DDS Reference* provides more information about adding graphics to applications.

Accessing Workstation Gateway Functions through Operations Navigator

You can access workstation gateway server functions from a command line interface or from Operations Navigator. Not all of the workstation gateway server functions are available on both interfaces. However, most workstation gateway server functions are available *only* through Operations Navigator.

This chapter discusses how to start, stop, and configure the workstation gateway server from the command line interface. It does not document any of the Operations

Navigators functions. See the online Help in Operations Navigator for information about using Operations Navigator for workstation gateway server functions.

To access workstation gateway server server functions through Operations Navigator, perform the following steps:

1. Double-click your AS/400 server in the main tree view of Operations Navigator.
2. Double-click **Network**.
3. Double-click **Servers**.
4. Double-click **TCP/IP**.
5. Double-click **Workstation gateway**.

Note: Although some of the functionality of the command line interface and the Operations Navigator is the same, the actual menu commands and parameters are not necessarily the same.

The online help in Operations Navigator assists you with the following:

- Starting the workstation gateway server.
- Automatically starting the workstation gateway server whenever TCP/IP is started.
- Ending the workstation gateway server.
- Configuring the workstation gateway server.

Starting the Workstation Gateway Server

To start the workstation gateway server, specify the following STRTCPSVR command with the SERVER attribute set to *WSG:

```
STRTCPSVR SERVER(*WSG)
```

Specify the *Change WSG Attributes* (CHGWSGA) command to restart the server whenever you run the *Start TCP/IP Servers* (STRTCPSVR) command:

```
CHGWSGA AUTOSTART(*YES)
```

Automatically Starting the Workstation Gateway Server

The AUTOSTART parameter controls when the AS/400 system starts the workstation gateway server. Specify *YES if you want the workstation gateway server to start whenever TCP/IP is started. If you use the STRTCPSVR command, this parameter is ignored, and the system starts the workstation gateway server. If the workstation gateway server is active, the system ignores any subsequent STRTCPSVR requests for workstation gateway.

Ending the Workstation Gateway Server

To end the workstation gateway server, specify the *TCP/IP Server* (ENDTCPSVR) command with the server attribute set to *WSG:

```
ENDTCPSVR SERVER(*WSG)
```

Specify the ENDTCPSVR command without parameters. This stops all TCP/IP servers.

Configuring the Workstation Gateway Server

You do not need to configure much to begin using the workstation gateway server. Most configuration options have default settings.

To configure the workstation gateway server, use the *Configure TCP/IP Workstation Gateway* (CFGTCPWSG) command.

Before you start the workstation gateway server for the first time, perform the following steps:

1. Change the *Display Sign-on Panel* option to *YES.

For information on making this change, see Figure 202 on page 324. Look for the option marked **5**, explained in “Display Sign-on Panel (DSPSGN)” on page 325.

2. Configure and use the user exit.

For information about configuring and using the user exit for the workstation gateway server, see “Using a WSG exit program to bypass the AS/400 Sign-on Display” on page 571.

The changes take effect the next time you start the workstation gateway server. For information on starting the workstation gateway server, see “Starting the Workstation Gateway Server” on page 320.

Note: The workstation gateway server might not recognize some customized sign-on panels. This is because the workstation gateway server expects input fields in certain locations. If the input fields differ greatly from the default sign-on panel shipped with the AS/400 system, a possibility exists that the system might fail to recognize the display as a sign-on panel. If, for example, a user exit returns a bad user profile, password, or other field, then the sign-on attempt fails. However, the workstation gateway server does not recognize the display as a sign-on panel, so the workstation gateway server continues processing and sends the sign-on panel to the Web browser. This is why you need to test your customized sign-on panel with the workstation gateway server and ensure that the workstation gateway server recognizes it before you start using the workstation gateway server.

To test your customized sign-on panel, start with the default sign-on panel. Make small changes and test each change. When the workstation gateway server no longer sends out the AS/400 sign-on panel, this means it now recognizes your customized display.

If you use *N0 for the *Display Sign-on Panel* parameter, the workstation gateway server only serves sign-on panels through a user exit. If no user exit is configured, the workstation gateway server does nothing. If you have configured a user exit, then the workstation gateway server attempts to identify any sign-on attempt that was not successful. An unsuccessful sign-on attempt occurs, for example, when the user exit provides a bad user profile or password. In these cases, the workstation gateway server does not send a sign-on panel to the requester.

Managing Virtual Devices for the Client

The workstation gateway server uses virtual devices to direct output to client devices. The AS/400 workstation gateway server support automatically selects (and creates, if necessary) these devices for you.

You must allow the workstation gateway server to configure virtual controllers and devices automatically. The QAUTOVRT system value specifies the maximum number of devices that the system automatically configures. Use the *Change System Value* (CHGSYSVAL) command to change the value of the QAUTOVRT system value. For example, specifying the following command string changes the number of virtual devices that you can allocate on a system to 50:

```
CHGSYSVAL SYSVAL(QAUTOVRT) VALUE(50)
```

Note: QAUTOVRT has been changed for Version 4 Release 1 to support numeric values of 0 through 32500 and a special value of *NOMAX.

To determine and set the maximum number of users you want signed on to the AS/400 system at any time, perform the following steps:

1. Set the QAUTOVRT value to the maximum value of 32500 or use the *NOMAX value.
2. Let your users use pass-through, TELNET, the workstation gateway server, and the virtual terminal application program interface until you decide that the number of virtual devices created is sufficient for normal system operation.
3. Change the QAUTOVRT value from 32500 to the number of virtual devices you require for normal system operation.

If you have never allowed automatic configuration of virtual devices on your system, the QAUTOVRT value is 0. A workstation gateway server connection attempt then fails because the workstation gateway server does not create more than the specified QAUTOVRT devices (zero). If you try to sign on, you receive a message (TCP2504) indicating that the Web browser session has ended and the connection is closed. In addition, the QTGTELNETS job in the QSYSWRK subsystem on the AS/400 TELNET server sends a message (CPF8940) indicating that a virtual device cannot be automatically selected.

If you change the QAUTOVRT value to 10, the next Web browser connection attempt causes the workstation gateway server to create a virtual device. The server creates this virtual device because the number of virtual devices on the controller (0) is less than the number specified in the QAUTOVRT (10). Even if you change the specified number to 0 again, the next user attempting a Web connection succeeds. When a Web connection attempt fails because the AS/400 server cannot create a virtual device, the system sends the CPF87D7 message the system operator message queue.

The workstation gateway server uses the following conventions for naming virtual controllers and devices:

- Virtual controllers are named QPACTLnn.
- Virtual device descriptions are named QPADEVxxxx.

When automatically configuring virtual devices with the workstation gateway server, the workstation gateway server does not delete virtual devices, even if the number of devices attached to the virtual controllers that are automatically configured exceeds the QAUTOVRT limit.

If you specify a value less than the number of virtual devices attached to QPACTLnn controllers and you want the extra devices deleted, you must manually delete any virtual devices that exceed the QAUTOVRT limit. If you delete devices to enforce a smaller QAUTOVRT value, begin by deleting the devices from the controller with the highest QPACTLnn value.

Note: The workstation gateway server does not manage the QAUTOVRT values directly. The Telnet application does the managing on behalf of the workstation gateway server.

Changing the Workstation Gateway Configuration

This topic explains each of the parameters that you control to manage the workstation gateway server configuration. Specify CFGTCPWSG to display the configuration (shown in Figure 201).

```
Configure TCP/IP Workstation Gateway                               System:  SYSxxx
Select one of the following:
    1. Change workstation gateway attributes
Related options:
    10. Configure HTTP
    11. Work with autoconfigure virtual devices
    12. Work with limit security officer device access

Selection or command
===> 1

F3=Exit  F4=Prompt  F9=Retrieve  F12=Cancel
```

Figure 201. Configure TCP/IP Workstation Gateway (CFGTCPWSG) Panel

Select option **1**, *Change Workstation Gateway Attributes*. The *Change Workstation Gateway Attributes* (CHGWSGA) command displays the configuration settings you can adjust. Figure 202 on page 324 and Figure 203 on page 324 show you the prompts, which are explained in this topic.

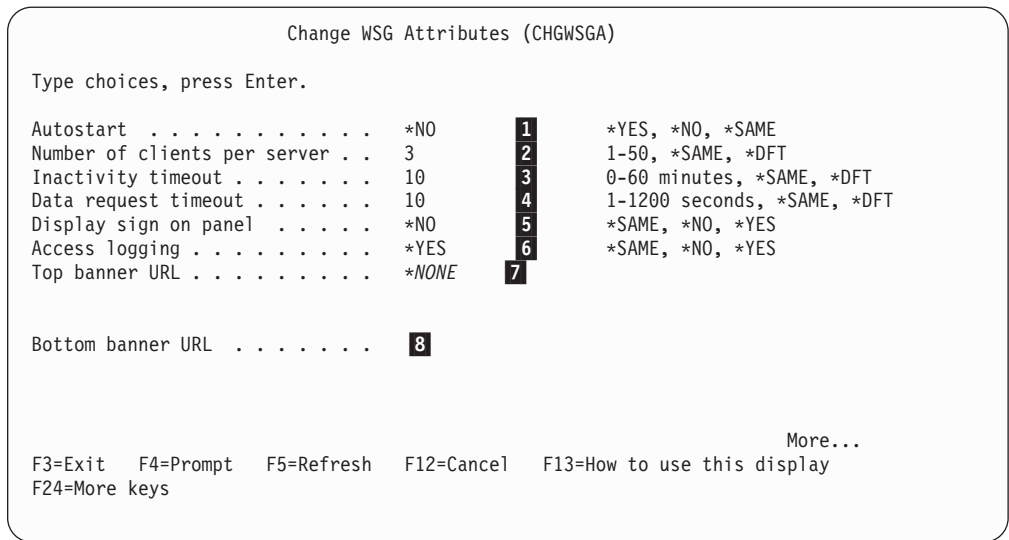


Figure 202. Change Workstation Gateway Attributes — Display 1

Page down to display the remaining prompts:

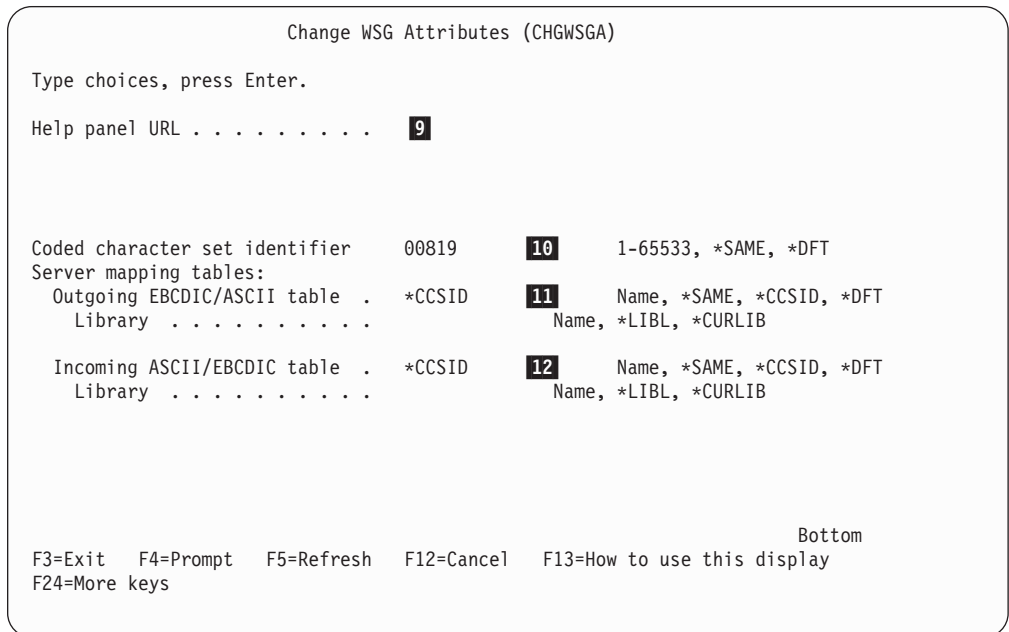


Figure 203. Change Workstation Gateway Attributes — Display 2

Number of Clients per Server (NBRCLT)

2

The AS/400 system manages the number of workstation gateway server jobs to ensure that at least 20 client sessions are always available for use. This number of server jobs either increases or decreases depending on the activity for the workstation gateway server at a given time.

By default, each workstation gateway server job handles 20 client sessions. You can change this value to anything from 1 through 50 clients per server job.

Number of Workstation Gateway Clients Per Server — Example

For example, assume you configured 5 client sessions per server job. The workstation gateway server ensures that four jobs are available for work. The 4 jobs allow 20 clients to use the workstation gateway server concurrently with this configuration (4 server jobs times 5 client sessions per server job allows 20 client sessions). There is also one additional job running for a total of 5 jobs. This additional job monitors the configured or default well-known port and distributes connection requests from clients to the other jobs. If your multiplexing value is lower, less bottlenecking occurs through a server. However, the AS/400 system requires more jobs and resources to support the same number of sessions.

Inactivity Timeout (INACTTIMO)

3

The inactivity timeout specifies how long a workstation gateway server session can be idle before the AS/400 system ends the session. By default, the system allows a workstation gateway server session to remain inactive for 10 minutes. After this time, it ends the session. Possible values are from 0 through 60 minutes. A value of 0 means that there is no timeout.

Note: It can take the system an additional 1 to 300 seconds to end the inactive session (jobs wake up at 5-minute intervals).

Data Request Timeout (DTARQSTIMO)

4

The data request timeout specifies how long the workstation gateway server waits from the time a client initially connects until the workstation gateway server receives all the request data.

Possible values are from 1 through 1200 seconds.

Network Timeout — Hint

Make the timeout value longer if you experience network delays. Make the value shorter to minimize resource usage caused by running more workstation gateway server jobs. If the timeout value becomes too long, too many jobs end up waiting to run. This happens particularly on small AS/400 systems.

Display Sign-on Panel (DSPSGN)

5

Specifies whether to display the AS/400 sign-on panel when a Workstation Gateway request comes in from a World Wide Web (WWW) browser.

Note: The workstation gateway server recognizes only generic, default AS/400 sign-on panels. If you have customized your AS/400 sign-on panel, the workstation gateway server might not recognize the panel. If the workstation gateway server cannot send your sign-on panel, start with the default sign-on panel and try making small changes to adjust your customized sign-on panel to a form that the workstation gateway server recognizes. The default for DSPSGN is *NO. For most everyone, leaving DSPSGN at *NO means that no one can use the workstation gateway server unless the AS/400 system cannot recognize the sign-on

panel. The DSPSGN parameter is not strictly for user exits only, but if you leave DSPSGN at the default, *NO, then only user exits let your users sign on.

To use an exit program, register your user exit program for exit point QIBM_QTMT_WSG, format QAPP0100. Use the *Work with Registration Information* (WRKREGINF) command to determine if the exit point is registered. If you do not use the exit point (meaning you set DSPSGN to *YES), the AS/400 system sends a sign-on panel to any browser that accesses the workstation gateway server. If you do not register an exit program at this exit point, no one can sign on. For more information about registering applications, see "Appendix E. TCP/IP Application Exit Points and Programs" on page 535.

Changing the Sign-on Panel File

To change the format of the sign-on panel, perform the following steps:

1. Create a changed sign-on panel file. You can change a hidden field in the display file named UBUFFER to manage smaller fields. UBUFFER is 128 bytes long and is stated as the last field in the display file. This field can be changed to function as an input/output buffer so the data specified in this field of the display will be available to application programs when the interactive job is started.

You can change the UBUFFER field to contain as many smaller fields as you need if the following requirements are met:

- The new fields must follow all other fields in the display file. The location of the fields on the display does not matter as long as the order in which they are put in the data description specifications (DDS) meets this requirement.
- The length must total 128. If the length of the fields is more than 128, some of the data is not passed.
- All fields must be hidden fields (type *H* in DDS source) or input or output fields (type *B* in DDS source).

Notes:

- a. Do not change the *order* in which the fields in the sign-on panel file are declared. However, you can change the *position* in which they appear on the display.
 - b. Do not change the total size of the input or output buffers. Serious problems can occur if you change the order or size of the buffers.
 - c. Do not use the data descriptions specifications (DDS) help function in the sign-on panel file.
2. Change a subsystem description to use the changed display file instead of the system default of QSYS/QDSIGNON. You can change the subsystem descriptions for subsystems that you want to use the new display.
To change the subsystem description, perform the following steps:
 - a. Use the *Change Subsystem Description* (CHGSBSD) command.
 - b. Specify the new display file on the SGNDSPF parameter.
 - c. Use a test version of a subsystem to verify that the display is valid before attempting to change the controlling subsystem.
 3. Test the change.
 4. Change other subsystem descriptions.

Notes:

1. The buffer length for the display file must be 318. If it is less than 318, the subsystem uses the default sign-on panel, which is QDSIGNON in library QSYS.
2. You cannot delete the copyright line.

Access Logging (ACCLOG)

6

This parameter tells the AS/400 system to create a table that logs all accesses to the workstation gateway server. When you specify *YES to log all accesses to the workstation gateway server, the system places access log records in the physical file QATMTLOG in library QUSRSYS.

For more information about the access log, see “Managing the Access Log” on page 330.

Top Banner URL (TOPBNRURL)

7

Specifies a Universal Resource Locator (URL) for displaying information on the World Wide Web (WWW) browser display, if desired. Unless specified, no banner URLs are displayed.

If you want to send a banner, you must create one. To create a banner, use an editor on a workstation that can create .GIF files. Next, store the file in an integrated file system library, such as in QDLS, QLANSrv, or QOpenSys. Figure 204 shows an example of a banner that workstation gateway server can send with the AS/400 sign-on panel.

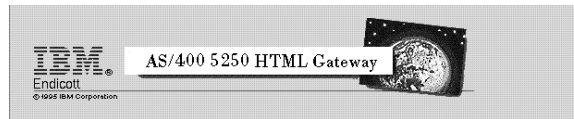


Figure 204. Example of a Banner

Bottom Banner URL (BOTBNRURL)

8

Specifies a banner at the bottom of the panel that you want to send. For more information, see “Top Banner URL (TOPBNRURL)”.

Help Panel URL (HLPPNLURL)

9

Specifies the Universal Resource Locator (URL) to display the online help information for the workstation gateway server. The user accesses the help information by clicking TIPS at the top of the display.

For more information about customizing the online help information for the Web browser clients, see “Granting Access to the Web Browser Online Help Information” on page 329.

Coded Character Set Identifier (CCSID)

10

The HTML specification requires browsers to indicate the content types that they can handle. The browser passes this information by including an ISO (international standard organization) identifier in a MIME header. The workstation gateway server converts this content type identifier into coded-character set identifier (CCSID) information. The AS/400 system uses the CCSID information to interpret the input data and to provide the output data in the proper format for the browser to use for its display. The input can be ASCII or EBCDIC.

If your Web browser has a MIME header, then the system uses the CCSID value from the MIME header to perform ASCII-to-EBCDIC conversion.

The AS/400 system uses the value specified in this parameter in the following cases:

- When your Web browser does not have a MIME header.
- When the MIME header values do not match any values in the CCSID table on the AS/400 system.

The workstation gateway server ignores ISO-xxxx-xx-style MIME headers, starting with the following PTFs:

- SF47241 WSG_41
- SF47242 WSG_32
- SF47243 WSG_37
- SF47244 WSG_42

The CCSID is the ASCII CCSID to which you must convert the EBCDIC data, if possible.

Server Mapping Tables (TBLWSGOUT) and (TBLWSGIN)

11 and **12**

You can use outgoing and incoming mapping tables to manage character conversion. Specify the tables that you want to use and the library in which the tables reside.

Workstation Gateway Server Mapping Tables — Hint

You can use the same conversion tables for workstation gateway server that you use for Telnet or another application. If you decide to do this, ensure that:

- The outgoing table maps from the EBCDIC of the interactive job to any of your ASCII tables.
- The incoming mapping table maps from any ASCII table to the EBCDIC of the interactive job.

For Korea, China, and Taiwan, the AS/400 system does not support MIME standard code pages. This is because of missing MIME standards or CCSIDs for those countries at this time. The system supports the CCSID parameter for each country, as contained in Table 31 on page 329.

For detailed information about CCSIDs, see *National Language Support*, SC41-5101, or *International Application Development*. Table 31 on page 329 contains conversion information.

Table 31. Supported MIME Standard Code Pages

MIME Standard	Name	ASCII CCSID	Character Set	EBCDIC CCSID
US-ASCII	US English	00367	103	00500
ISO-8859-1	Latin-1	00819	697	00500
ISO-8859-2	Latin-2	00912	959	00870
ISO-8859-5	Cyrillic	00915	1150	01025
ISO-8859-7	Greek	00813	925	00875
ISO-8859-8	Hebrew	00916	941	00424
ISO-8859-9	Latin-5	00920	1152	01026
ISO-2022-JP	Japan MBCS	05052	1064,1062,1121,1120	05026
	Taiwan	00950		00937
	China	01381		00935
	Korea	00949		00933

For a list of supported CCSID values that the initial URL can request, see the NLS keyboard (KBD) values, in Appendix C in the *National Language Support* book.

Workstation Gateway Exit Point for Accessing a User Profile Directly

The WSG Application Logon Exit Point (QAPP0100) allows you to bypass the AS/400 sign-on panel and start an application without the client browser sending a user profile or password. This allows you to provide *any* application to client browsers without requiring a sign-on. To do this, your exit program must authenticate the client request and provide sign-on information to the workstation gateway server. The workstation gateway server then uses the output of your exit program as input to the Virtual Terminal APIs and signs on for the client browser.

For information about how to create an exit program, see “Using a WSG exit program to bypass the AS/400 Sign-on Display” on page 571. For a list of supported CCSID values that the initial URL can request, see the NLS keyboard (KBD) values, in Appendix C in the *National Language Support* book.

Granting Access to the Web Browser Online Help Information

To enable the Tips push button on your web browser, perform the following steps:

1. Create a directory for the help information.
2. Copy QTCP/QATMHELP.WSGHELP to the directory.
You can use the WRKMBRPDM command to do this.
3. Change the member type of the help information to HTML.
You can use the WRKMBRPDM command to do this.
4. Enable the new directory to the HTTP server.
Use the HTTP server configuration directives to do this.
5. Grant authority to the HTTP server (default QTMHHTTP user profile) to access the directory and file.
6. Ensure that you have started or restarted the HTTP server.
7. Change the help URL by using the CHGWSGA command.

8. Stop, then restart the workstation gateway server to enable the change.

If you want to serve the Tips push button from a different server, use the instructions provided with that server to enable the online help information. If you serve the help information from another server, then must move the file off the AS/400 system and rename it to the UNIX, DOS, OS/2, or other convention for help (such as wsghelp.html or wsghelp.htm).

The online help information is also available on the Internet at this URL:

<http://www.as400.ibm.com/htmlgate>

Customizing Web Browser Online Help Information

If desired, you can customize the information that your workstation gateway server sends to Web browsers. To do this, first enable the online help information. Next, use a browser editor to change the HTML content in the help file.

Managing the Access Log

The QATMTLOG file is shipped with MAXMBRS(*NOMAX) and SIZE(*NOMAX). Therefore, if you plan to use logging, you can change these file attributes to set limits. If you set limits and the file becomes full for the second time on the same day, the AS/400 system renames the file to a new member sends a message indicating that it has done so. The job log and the QSYSOPR message queues both receive the message. The member name has the format Qcyyymmdd, where c=0 means 19xx and where c=1 means 20xx. The scheme is based on the date, meaning that the rename works only once a day.

If you find that you receive such messages more than once a day, consider enlarging the limit so that the system can record a day's worth of log records in a file. When the system is renaming the log file, it does not log any access records or affect the workstation gateway server function.

The AS/400 system renames this log file only once on any one day. When the log file is full, the system sends a message to the job log and to QSYSOPR stating that the file is full. Logging suspends when the file is full. You must delete or rename the file member yourself to activate logging again.

The QATMTLOG File

QATMTLOG *MBR on AS/400

Figure 205 on page 331 shows a partial display that results when you display the QATMTLOG member using the STRPDM command.

```

                                Specify Members to Work With

Type choices, press Enter.

File . . . . . QATMTLOG      Name, F4 for list
Library . . . . . QTCP       *LIBL, *CURLIB, name

Member:
Name . . . . . *ALL         *ALL, name, *generic*
Type . . . . . *ALL         *ALL, type, *generic*, *BLANK

F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel

```

Figure 205. Using STRPDM to Display the QATMTLOG Member — Display 1

```

                                Work with Members Using PDM                                AS007

File . . . . . QATMTLOG
Library . . . . . QTCP      Position to . . . . .

Type options, press Enter.
3=Copy   4=Delete   5=Display   7=Rename   8=Display description
9=Save   13=Change text  18=Change using DFU  25=Find string ...

Opt Member   Date      Text
5   QATMTLOG  04/16/96  Access log file for CHGWSGA command

Parameters or command                                Bottom
===>
F3=Exit   F4=Prompt   F5=Refresh   F6=Create
F9=Retrieve F10=Command entry F23=More options F24=More keys

```

Figure 206. Using STRPDM to Display the QATMTLOG Member — Display 2

Type a **5** to display the log file member. The AS/400 system displays the contents of the log file as shown in Figure 207 on page 332.

The log contains the following information:

- The date and time the record was logged.
- The IP address of the local system.
- The IP address of the remote system.

- The number of minutes until the connection ended.
- The number of seconds until the connection ended.
- The identifier of the AS/400 display that was sent to the remote system.
- A device type identifier.

The AS/400 system might not always recognize the display identifier (QPADEVnnnnnn) on the sign-on panel. For example, the system not recognize some custom displays. If it does not recognize the display, the device columns are blank.

- A log key.

The *log key* is a variable hexadecimal string that is dynamically generated for each client. It uses timestamp as a seed. 4C4E6F7E in Figure 207 is an example of a log key.

- A description of the activity that was logged.

```

File . . . . . :      Display Physical File Member
Member . . . . : QATMTLOG      Library . . . . . : QUSRSYS
Control . . . . : QATMTLOG      Record . . . . . : 1
Find . . . . . :      Column . . . . . : 1
*...+...1...+...2...+...3...+...4...+...5...+...6...+...7...+...8...+...9...+...0...+...1...+...2...+...3
04/29/96 8:32:13 AM 9.130.42.106 9.130.69.47 0 0 QPADEV0042 4C4E6F7E: 3179-2 Session started.
04/29/96 8:32:37 AM 9.130.42.106 9.130.69.47 0 29 QPADEV0042 4C4E6F7E: 3179-2 Session ended.
04/29/96 8:35:51 AM 9.130.42.106 9.130.69.47 0 0 00000000: Client tried to access active session owned
by another client.
04/29/96 10:05:02 AM 9.130.42.106 9.130.69.47 0 0 00000000: 3179-2 Request for WSG session has been
refused.
04/29/96 10:06:45 AM 9.130.42.106 9.130.69.47 0 0 QPADEV0027 B82608E2: 3179-2 Sign-on panels disabled.
04/29/96 11:33:59 AM 9.130.42.106 9.130.69.47 0 0 QPADEV0027 E9213AFA: 3179-2 Sign-on panels disabled.
04/29/96 11:34:46 AM 9.130.42.106 9.130.69.47 0 0 QPADEV0027 0A24B605: 3179-2 Sign-on panels disabled.
04/29/96 11:35:47 AM 9.130.42.106 9.130.69.47 0 0 00000000: 3179-2 Request for WSG session has been
refused.
04/29/96 11:37:49 AM 9.130.42.106 9.130.69.48 0 0 QPADEV0027 DFBF88BE: 3179-2 Session started.
04/29/96 11:38:28 AM 9.130.42.106 9.130.69.48 0 39 QPADEV0027 DFBF88BE: 3179-2 Session ended.
04/29/96 11:45:04 AM 9.130.42.106 9.130.69.47 0 0 QPADEV0027 C280FC9B: 3179-2 Session started.
04/29/96 11:46:53 AM 9.130.42.106 9.130.69.47 0 0 QPADEV0027 C280FC9B: 3179-2 Session ended.
04/29/96 12:03:23 PM 9.130.42.106 9.130.69.47 0 0 QPADEV0027 F1082E61: 3179-2 Session started.
04/29/96 12:04:02 PM 9.130.42.106 9.130.69.47 0 42 QPADEV0027 F1082E61: 3179-2 Session ended.
04/29/96 12:26:06 PM 9.130.42.106 9.130.69.47 0 0 QPADEV0027 82CAF856: 3179-2 Sign-on panels disabled.
04/29/96 12:26:32 PM 9.130.42.106 9.130.69.48 0 0 QPADEV0027 33709128: 3179-2 Sign-on restricted through exit point
QIBM_QTMT_WSG format QAPP0100.
04/29/96 1:12:20 PM 9.130.42.106 9.130.69.47 0 0 QPADEV0027 0D8A0DF6: 3179-2 Session started.
04/29/96 1:12:23 PM 9.130.42.106 9.130.69.47 0 5 QPADEV0027 0D8A0DF6: 3179-2 Session ended.
04/29/96 1:15:30 PM 9.130.42.106 9.130.69.47 0 0 QPADEV0027 F2F0B75A: 3179-2 Session started.
04/29/96 1:17:33 PM 9.130.42.106 9.130.69.47 2 3 QPADEV0027 F2F0B75A: 3179-2 Session ended.
04/29/96 1:34:25 PM 9.130.42.106 9.130.69.47 0 0 QPADEV0027 7E6AB936: 3179-2 Session started.
04/29/96 1:35:15 PM 9.130.42.106 9.130.69.47 0 51 QPADEV0027 7E6AB936: 3179-2 Session ended.
04/29/96 2:05:46 PM 9.130.42.106 9.130.69.47 0 0 QPADEV0028 A69EBE2B: 3179-2 Session started.
04/29/96 2:05:51 PM 9.130.42.106 9.130.69.47 0 6 QPADEV0028 A69EBE2B: 3179-2 Session ended.
04/29/96 2:19:00 PM 9.130.42.106 9.130.69.47 0 0 00000000: Exit point QIBM_QTMT_WSG format QAPP0100
not active.
***** END OF DATA *****
Bottom
F3=Exit F12=Cancel F19=Left F20=Right F24=More keys

```

Figure 207. The QATMTLOG Member — Example

Accessing the Workstation Gateway from a Web Browser

To access the workstation gateway server from your Web browser, you must specify the Universal Resource Locator (URL). The URL identifies the following attributes:

- The protocol that your browser uses when contacting the server.
- The location of the server and of the requested object.

To start a session from a Web browser to the workstation gateway server, specify the URL of your host AS/400 system, as in the following example:

```
http://hostname:port/WSG
```

For other supported URL formats, see “Other Supported URL Formats” on page 334.

If the workstation gateway server has an exit program that requires information to be passed to it, use the following form:


```
http://hostname:port  
/WSG/QAPP0100?op_specific_parm
```

http:

Identifies the protocol. The workstation gateway server uses the HTTP protocol.

hostname

Identifies the system to which you are sending a request, such as
xxx.xxx.xxx.xxx.

Note: To use the host name, you must have either a domain name server or a local host table.

To reach a host for which you do not have a name server or a host table, use the dotted decimal address of the host.

:port

Port **5061** is the default port for the workstation gateway server. You can change this port by using the WRKSVRTBLE command. Keep the following in mind if you choose a different port:

- Use wsg for the name of the service.
- Use tcp for the protocol.
- Use the dotted decimal or the alias form for the host name.

Either is accepted. Therefore, you can use the workstation gateway server whether or not you have configured a host table.

If you do not specify a port on the URL, the request goes to the AS/400 HTTP server at well-known port 80. Because you specified WSG, HTTP redirects your request to the workstation gateway server.

Note: You must configure the HTTP server to allow it to redirect traffic. To do this, specify the *redirect* directive:

```
Redirect /WSG http://--address--:5061/WSG
```

WSG

The WSG request keyword.

Note: Specify WSG in upper case.

/QAPP0100?

The prefix that indicates the exit point information follows. (The '?' is just a separator character between QAPP0100 and any arguments that the system passes.)

op_specific_parm

Information about what operations the workstation gateway server must perform.

A signature for a workstation gateway server session is added to your address after your initial connection. This signature identifies your session to the AS/400 system.

Notes:

1. The workstation gateway server port is different from the HTTP server port. This is because workstation gateway server is a new type of server for which no well-known port exists.
2. When using the workstation gateway, you might find that the view of the AS/400 display varies between browsers. If there are alignment problems with the fields

on the display with the browser you are using, try another browser. The workstation gateway server does not cause this problem.

URL Request Form for NLS

Translation is usually from the EBCDIC CCSID of your AS/400 system to the ASCII CCSID specified on the CCSID parameter of the CHGWSGA command. When the value of this parameter changes, the EBCDIC CCSID default also changes because the workstation gateway server uses the *best fit* EBCDIC CCSID for the ASCII CCSID.

Individual users can override this default by using the URL request form for NLS.

The format of the URL request for NLS is as follows:

```
http://hostname:port/WSG-XXX
```

where XXX is any three-character keyboard string listed in Appendix C of the *National Language Support* book.

This URL allows you to specify a keyboard, code page, and character set for the client display session.

To request the keyboard for the Belgian Multinational Character Set (BLI), for example, use the following URL request:

```
http://hostname:port/WSG-BLI
```

With the Workstation Gateway User Exit, the format is as follows:

```
http://hostname:port/WSG-BLI
```

See *National Language Support* for a table of supported keyboard strings that you can use.

Other Supported URL Formats

In addition to the URL formats shown in “Accessing the Workstation Gateway from a Web Browser” on page 332 and in “URL Request Form for NLS”, the following are also supported:

```
http://hostname:port/WSG  
/QAPP0100?any_QAPP0100_info
```

```
http://hostname:port/WSG-XXX  
/QAPP0100?any_QAPP0100_info
```

For information about how to create an exit program, see “Using a WSG exit program to bypass the AS/400 Sign-on Display” on page 571.

Security

The following topics consist of tips for how to control the security of your AS/400 system.

Taking Steps to Ensure Workstation Gateway Security

Stop the Server: The easiest way to secure your AS/400 system from Web browsers is to stop the workstation gateway server.

Control User Profiles That Can Access AS/400: To run the server while controlling which user profile gets sent to a Web browser, you can write an exit program. For more information on writing exit programs, see “Adding Your Exit Program to the Registration Facility” on page 537.

Things to Avoid to Ensure Workstation Gateway Security

Do Not Store Passwords in HTML Documents: Secured Sockets Layer (SSL), or Secure HTTP, has been put into effect on AS/400 systems beginning with Version 4 Release 1. This helps to ensure that data being sent from a secure server to a secure browser is encrypted. In general, however, do not store passwords in HTML documents.

Workstation Gateway — Requirements

Some applications continually display panel after panel without requiring keyboard interaction or intervention. Because the workstation gateway server processes one panel at a time, the workstation gateway server cancels applications like this similar to a SYSREQ panel, Option 2. Do not use screen refresh applications or features with workstation gateway server.

How the 5250 Display is Formatted for the Workstation Gateway

The workstation gateway server formats the 5250 display as follows:

1. The display is presented as 80-column, pre-formatted text.
2. Blank lines are omitted to reduce the vertical size of the display image in the browser window.
3. 5250 input fields are translated into HTML INPUT tags with the same length as the 5250 field.
 - If an input field is longer than 160 characters, the field is translated into an HTML text area that is 80 columns long and has enough lines to contain all the needed input text.
 - If an input field is between 60 and 160 characters long, it is presented as an input field with 60 visible characters and a maximum length of the 5250 field size.
4. Function key tags recognized in the function key area of the display are added to the *menu* items at the top of the document.

For example, Figure 208 on page 336 shows a 5250 display *without* workstation gateway.

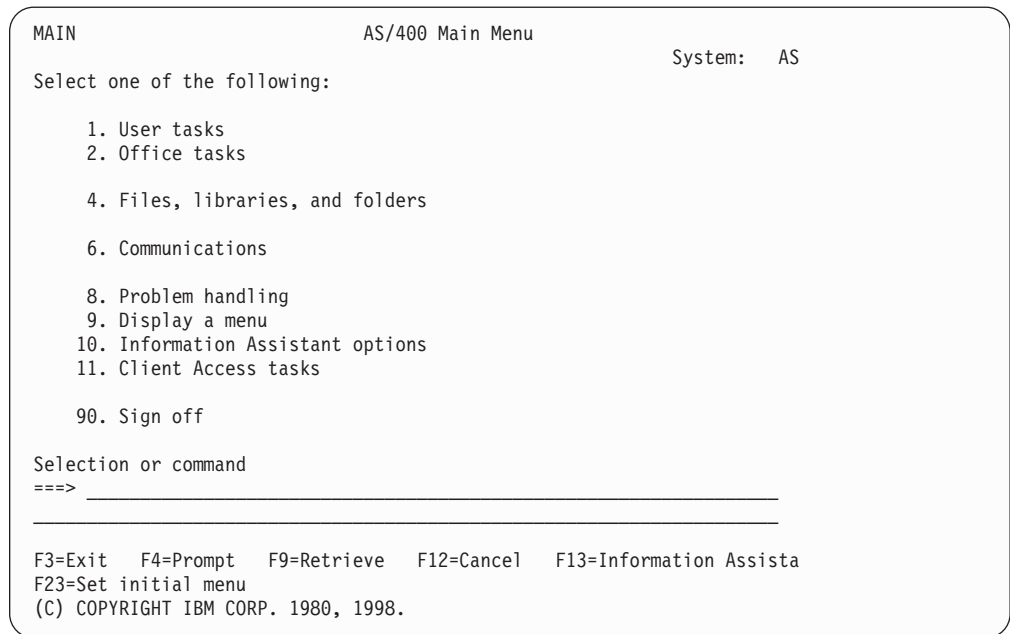


Figure 208. AS/400 Main Menu Display Without Workstation Gateway

Figure 209 shows the same display using workstation gateway (with IBM WebExplorer for Windows 95).

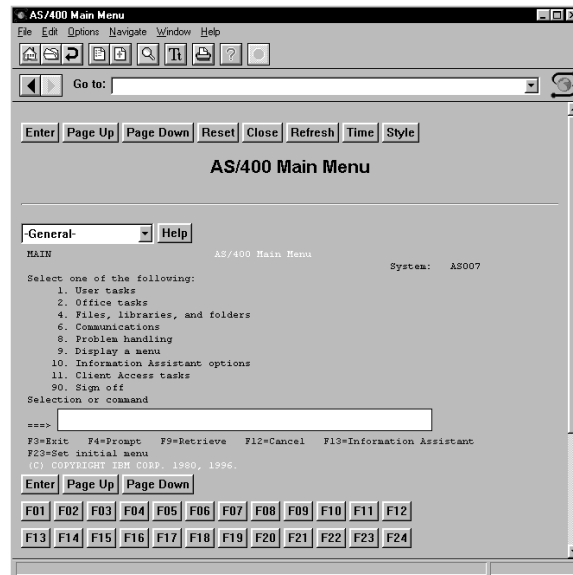


Figure 209. AS/400 Main Menu Display With Workstation Gateway

Converting DDS Applications to HTML

You can change your application display programs to enable them for the Internet by inserting HTML to take advantage of the graphics capabilities of Web browsers. For more information and examples, see *Application Display Programming*, SC41-5715-00.

Note: The DDS keyword HTML is currently not supported in a sub-file record.

Configuration Examples

This topic contains several examples for changing the workstation gateway server configuration.

Starting the Server Automatically when TCP/IP Starts — Example

To automatically start the workstation gateway server when the Start TCP/IP (STRTCP) CL command runs, change the Workstation Gateway attributes as follows:

```
CHGWGSA  AUTOSTART(*YES)
```

The next time the STRTCP command runs, the WSG server is also started.

Changing the Number of Client Sessions per Server Job — Example

This command indicates that the next time you start the workstation gateway server, it handles up to 30 client sessions. If you have a small system, specify a larger number to reduce the amount of resource dedicated to the server. To reduce bottlenecking of workstation gateway server sessions through a server, specify a smaller number. Bottlenecking is more likely to occur if your AS/400 system is a smaller model.

```
CHGWGSA  NBRCLT(30)
```

Using Server Mapping Tables — Example

You can use any tables that you already created for Telnet. However, the AS/400 system supports only SBCS tables and not DBCS tables this way. For more information about mapping tables, see Appendix C. Mapping Tables Associated with TCP/IP Function.

This command indicates that the next time you start the workstation gateway server, it has the following characteristics:

- The ASCII-to-EBCDIC and EBCDIC-to-ASCII conversion is not done with a CCSID value, but instead the outgoing and incoming mapping tables are used.
- A copy of the information found in the table object named TSTWSGO is used for mapping outgoing data in workstation gateway server. The table object is found in one of the libraries in the library list.
- A copy of the information found in the table object named TSTWSGI is used for mapping incoming data in workstation gateway server. The table object is found in one of the libraries in the library list.

```
CHGWGSA  TBLWSGOUT(*LIBL/TSTWSGO)
          TBLWSGIN(*LIBL/TSTWSGI)
```

Using a CCSID When MIME Code Page Not Available—Example

The following example specifies the ASCII CCSID for China, taken from Table 31 on page 329. Because a MIME standard does not yet support China, Korea, and Taiwan, the AS/400 system does not automatically recognize the CCSID values for those countries. Therefore, these countries must specify their CCSID to run the workstation gateway server.

```
CHGWGSA  CCSID(01381)
```

Resolving Network Delays — Example

If you have a slower modem or a small model AS/400 system, you might experience delays on the network. In this case, you can change the data request timeout value to a higher number. Choose a number somewhat higher than the default, and experiment with that value for a while. You must balance network delays with resource usage that can cause bottlenecks on the system. The following command changes the data request timeout value to 60 seconds:

```
CHGWSGA DTARQSTIMO(60)
```

Online Help Information

The following information is what you see when you select the **Tips** button when using the workstation gateway server:

- A Word About Browsers...
- Signing on to the AS/400 workstation gateway server
- Quick Tips for a Fast Start
- Using the Buttons
- Using the Menu Boxes
- FAQs (Frequently Asked Questions)
- Working with AS/400 Menus
- AS/400 Menu Parts

A Word About Browsers...

As you read through this document, remember that Web browsers interpret HTML differently. Your Web browser affects the way that the AS/400 menu and list windows look on your display. For example, the *General* menu might look like a spin box in the IBM WebExplorer, but in Netscape or Mosaic, it might look like a drop-down menu.

It is impractical to provide information for every interface variation caused by a browser. As a result, the information in this document might not accurately describe the interface that you are using.

Also, because of the nature of today's browsers, you might be unable to view this help document and the AS/400 window in which you are working at the same time. You can print this document for your convenience now and in the future.

Signing on to the AS/400 Workstation Gateway Server

All of the parts you are used to seeing on a typical AS/400 sign-on panel are on the AS/400 workstation gateway server display, too, like the following:

- Menu title.
- System and subsystem name.
- Sign-on area for specifying your user ID and password.

To sign on to your AS/400 system:

1. Type your user ID in the *User* box.
2. Type your password in the *Password* box.
3. If you want to call a specific program or procedure, go to a specific menu (or specify a library), and type the name in the appropriate box.

4. Click *Enter*.

Quick Tips for a Fast Start

The AS/400 workstation gateway server provides the same capabilities that you find on a typical AS/400 system. How you access those capabilities, however, is different from what you are used to doing. This section provides some quick tips to help you get started.

Using the buttons: Along the top of most AS/400 windows, you find a series of buttons. These include the following:

Enter button

Clicking **Enter** is the same as pressing the Enter key.

Page Up

Clicking **Page Up** moves you up (back) one display.

Page Down

Clicking **Page Down** moves you down (forward) one display.

Reset button

Clicking **Reset** returns all fields back to their default state. This button might look different from the other buttons in the action bar because it is a browser function and not a server function.

Close button

Clicking **Close** disconnects from the workstation gateway server. Remember that you can sign off from the AS/400 system and still be running a workstation gateway serversession. After clicking **Close**, you must start a new session to get to a sign-on panel again.

Refresh button

Clicking **Refresh** updates your AS/400 workstation gateway serverconnection. This button is useful in the following three different ways:

- It helps you find the *active* AS/400 panel. If you get lost in the history list on your WWW browser and lose track of which panel is active, clicking **Refresh** on any recently used panel brings up the active panel again. This saves you from having to hunt through old, inactive panels.
- It is the mechanism for returning to the active panel from the **Time** button.
- It is essential to anyone attempting to Telnet to a 370 system, as 3270 data streams (VM systems) differ significantly from 5250 data streams (AS/400 systems). An AS/400 system gives the workstation gateway server browser a clear indication when it receives all the data, while VM systems have no such indicator. This means that the AS/400 must estimate when it has all the display information from a VM system. Sometimes this estimate is inaccurate, which means you need the **Refresh** button to receive the additional display information.

Time button

Clicking **Time** shows you how long your session has been connected to the AS/400 workstation gateway server. It also shows you how long your session can remain idle before it is disconnected.

Style button

Clicking the **Style** changes the general format of the display in terms of function key placement and usability. Specifically, by default, a selection of function keys (F1-F24) is available across the lower area of all displays.

Clicking **Style** displays a *Function* pull-down menu instead of the function keys array. This mode of operation continues through the remainder of your session or until you click **Style** again.

Notes:

1. An exception is the sign-on panel. The function keys are not available on this panel, and you must use the *General* pull-down menu and the keys that are supplied there.
2. On the other panels, all of the function keys are posted, although some of them do not have actions associated with them.

Help *General Help* and *Field Context-Sensitive Help* are the two types of help that are available to the user.

- You can request General Help through the Help pull-down menu or button. To do this, select the Help option or click the Help button, respectively. This has the effect of first moving the cursor to row 1 and column 1 on the display before displaying the help. Since row 1 and column 1 are likely to be context insensitive, the General Help panel should appear.
- To access the Field Context-Sensitive help, place a ? character in position 1 of the input field. Only in the first position is the ? character interpreted as a command to move the cursor to this input field. Because the ? is an escape command, this means the ? is *not* left in the input field on return from the function key button action. In essence, the ? means to move the cursor *here*, then whatever other button action you request acts after the move occurs.

Along the bottom of most windows is another series of buttons. These include the following:

Enter button

Clicking **Enter** is the same as pressing the Enter key.

Page Up button

Clicking **Page Up** scrolls up (back) one display.

Page Down button

Clicking **Page Down** scrolls down (forward) one display.

Function key buttons (F1-F24)

Depending on which style you are currently using, function key buttons might not be displayed at the bottom of the display. The default for a session is to display the function key buttons at the bottom of the display. If you press the **Style** button, however, these buttons are moved to a *Functions* menu pull-down instead of being shown at the bottom of the display. This saves display space in addition to being displayed faster.

Using the menu boxes: A series of *menu boxes* is located along the top of most AS/400 windows. The type of boxes available varies with the type of AS/400 window you are displaying. The sign-on panel displays the *General* and *Help* menus but not the *Functions* menu. If a menu box has only one selectable item in it, it is shown as a button instead of as a menu box. You might see menu boxes like those in Figure 210 on page 341.

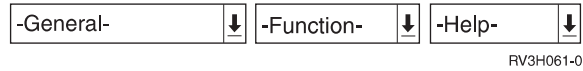


Figure 210. Bottom Action Bar Buttons

There is also a chance that you might see meny boxes like those in Figure 211 .



Figure 211. Function Key Buttons

Also, depending on your browser, these menu boxes might look like *spin dials* or *drop-down menus*. You can activate them by clicking an arrow or just a shadowed area in the box, depending on your browser.

A spin dial has both an *up* and a *down* arrow in the box. A drop-down menu has just a down arrow, or a shadowed area, in the box.

- To use a spin dial, follow these instructions:
 1. Click the arrows to rotate (or spin) through the choices in the box.
 2. When the choice you want appears in the box, click **Enter**.
- To use a drop-down menu, follow these instructions:
 1. Click the down arrow to display a menu of choices.
 2. Select the choice you want.
 3. Click **Enter**.

Here are the possible menu boxes and their associated choices:

- General box
- Functions box
- Help box or button

General Box: The **General** box displays various actions you can perform to exit or to go to other windows. These include the following actions:

- *Page Up*
- *Page Down*
- *Clear*
- *Record Back*
- *PA1*
- *PA2*
- *PA3*
- *Print Screen*
- *Test Request*
- *Host print screen* to print the information currently shown in the window.
- *Attention program* to call a user-defined attention-key-handling program.
- *System requests* to go to the System Request menu.
- *Sign off* to sign off the AS/400 system.
- *F12*
- *F3*

Functions Box: All of the function keys are still available, although you cannot press a function key on your keyboard to access them. Instead, look for them in the *Functions* box. Please note that the *Functions* box is available only if the function key buttons are not displayed at the bottom of the browser window (as controlled by the **Style** button). The specific functions include the following:

- F1
- F2
- F3
- F4
- F5
- F6
- F7
- F8
- F9
- F10
- F11
- F12
- F13
- F14
- F15
- F16
- F17
- F18
- F19
- F20
- F21
- F22
- F23
- F24

Help Box (or button): The Help box calls general help for the current AS/400 window. This might be shown as a button instead of a menu box if there is only one selectable item.

Frequently Asked Questions

If you accustomed to using your AS/400 in a *green screen* environment, the following questions explain how to perform a green screen task in the gateway environment.

1. Where are my function keys?

Those familiar functions keys found at the bottom of a typical AS/400 display are available through the *Functions* box near the top of the window.

Function keys are accessible in two different ways. As controlled by the **Style** button, the 5250 function keys are represented by buttons either at the bottom of the screen or under the functions box. If they are shown at the bottom, then the *Functions* box is not present.

Instead of pressing a function key, perform the following steps:

- a. Click the *Functions* box near the top of the window.
- b. Select the function you want.
- c. Click **Enter**.

2. How do I use my mouse to make a selection?

Use your mouse to move the cursor to the area in which you want to work and then click the left mouse button to make your selections.

3. How do I use my keyboard to make a selection?

- To select a push button using your keyboard, use the **Tab** key to move the cursor to the first push button in a group of push buttons. Next, use the arrow keys to put the cursor focus on the desired push button. Once the cursor focus is on the push button you want to select, press **Enter**. This is the same as clicking on the push button with your mouse.
- To select a menu box item using your keyboard, use the **Tab** key to move to the menu box in which you want to work. Next, use the arrow keys to select your choice. When the desired choice appears in the menu box, press **Enter**.

Note: If you select multiple menu box items, the results are unpredictable. Normally, only the last (farthest to the right) selection is used. Thus, if you select **PA1** from the *General* menu and then **F4** from the *Functions* menu, then **F4** is more likely to be done over **PA1**.

4. Why can't I type into an input field?

If the input field has *any* characters in it, it is probably filled up with data already. You must delete these characters to make room to type in new characters. Unfortunately, browsers do not provide overtyping capability in input fields.

5. **What do I do when my Web Browser refers to 5250 keys?**

You can either press a button that is marked with the key name or use the *General* or *Function* boxes to select a key.

6. **How do I press *Enter* on a AS/400 Workstation Gateway screen?**

Click **Enter**. It performs the same function as the Enter key on a Telnet 5250 screen.

7. **How do I get AS/400-specific help?**

Use the *Help* box or button at the top of the display. Click the arrow, select the type of help you want, and then click **Enter**.

8. **Can I surf to other places on the Web while I am signed on to the AS/400 Workstation Gateway ?**

Yes, you can. But remember, if your gateway connection remains idle for too long, your session is disconnected. To determine how long your connection can remain idle, click **Time**.

9. **Is it safe to store my AS/400 password in an HTML document?**

No, it is not. It is possible for other Web users to *read* your document as it is transmitted over the Web.

Working with AS/400 Menus

AS/400 menus provide easy access to a variety of AS/400 functions. To select an option on an AS/400 menu, specify the menu choice number on the command line and click **Enter**.

List windows (also called *Work with* screens) display a list of objects with which you can work by using a variety of options. List windows usually have a command line below the list, in which you can perform the same action on multiple objects at the same time. To do this, select one or more items in the list, enter command parameters on the command line, and click **Enter**.

AS/400 Menu Parts

This section describes the parts of an AS/400 menu.

Top Banner: If configured, this displays the contents of the top banner.

Top Action Bar: The menu bar is described in the section “Using the buttons” on page 339.

Title: The title shows the name of the window. This is what is saved in your browser’s history list.

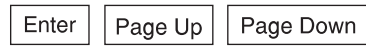
Menu Boxes: The menu boxes for a list window provide access to a variety of actions you can perform on list items. The menu boxes for a list window include the following:

- General
- Functions
- Help

For a definition of these boxes, see "Using the menu boxes" on page 340.

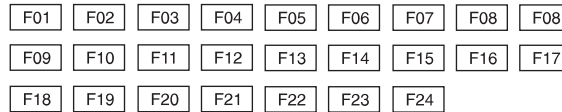
Window Area: This is the actual AS/400 window and any command line area. Large input fields are displayed as text areas so that you can see all the input being typed. This area also includes any description of function key actions and AS/400 messages.

Bottom Action Bar: The bottom menu bar is a subset of the top menu bar. It consists of the following buttons:



RV3H062-0

Function Key Area: Use the **Style** button to make the following function key buttons toggle between being displayed or hidden in the *Functions* menu pull-down:



RV3H063-0

Bottom Banner: If configured, this displays the contents of the bottom banner.

Chapter 11. Line Printer Requester (LPR)

You can request to have your spooled files sent to any system in your TCP/IP network. The term that UNIX TCP/IP software uses to describe this support is **line printer requester (LPR)**. LPR is the sending, or client, portion of a spooled file transfer. On the AS/400 system, the *Send TCP/IP Spooled File* (SNDTCPSPLF) command provides this function by allowing you to specify on which system you want the spooled file printed. It also allows you to specify how you want it printed.

The printing facilities of the destination system handle the printing of the file. On the AS/400 system, the **line printer daemon (LPD)** is the process on the destination system that receives the file sent by the SNDTCPSPLF command. For more information on LPD, see “Chapter 12. Line Printer Daemon (LPD)” on page 363.

LPR Command

The LPR command provides the same parameters and function as the SNDTCPSPLF command.

The LPR command sends a control file to the LPD server. This control file contains attributes and options that the LPR/LPD protocol recognizes. These attributes and options are not like the attributes that an AS/400 holds for a spooled file. Instead, they refer to information such as the names of the sending system, the sending user, and the name for the request. Some limited information about how the you need to print the file, such as the number of copies and information about the separator page, is included.

The LPR command sends these attributes and options to the destination system in a control file, which is always in ASCII format. This control file is separate from the file containing the data stream that it prints.

Client (LPR) and Server (LPD) Relationship

Figure 212 shows the client and server relationship between LPR and LPD.

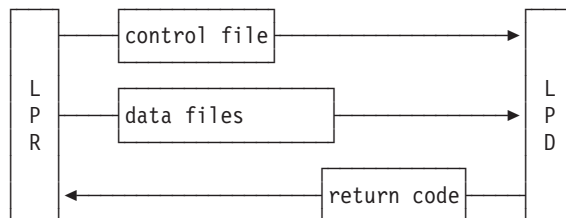


Figure 212. The Client (LPR) and Server (LPD) Relationship

Configuration Requirements for LPR

LPR puts a host name in the control file that it sends in addition to the data file. The receiving system uses this information to determine if the print request came from an authorized host. LPR uses the local domain and host name if you have configured them (Option **12** on the CFGTCP menu). Otherwise, it looks for the host name in the local host table or the name server that you have set up with Options **10** and **12**, respectively, on the CFGTCP menu. If LPR does not find a name, it uses IP followed by the dotted Internet address of the AS/400 system.

To configure the Local Domain and Host Name, use Option **12** (Change TCP/IP Domain Information) from the CFGTCP menu.

Sending a Spooled File (LPR)

The *Send TCP/IP Spooled File* (SNDTCPSPLF) command provides support for the remote printing of spooled files on a specified host and print queue using TCP/IP. You can send files to other AS/400 systems and non-AS/400 systems. Regardless of the type of destination system (AS/400 system or non-AS/400 system), you need to specify a remote system name or an Internet address. You must also specify the name of the spooled file that you are sending (FILE parameter). The values for the other parameters depend on the destination system. For some common parameter settings, see Table 32 on page 351.

When sending spooled files with the SNDTCPSPLF command from the client (local) AS/400 to an LPD server (remote) AS/400, perform the following steps:

1. Locate the spooled file that you want to send.
2. Start the spooled file transfer with the SNDTCPSPLF command.

Step 1 — Locate the Spooled File that you Want to Send

1. Specify the WRKSPLF command to see your spooled files:

```
Work with All Spooled Files

Type options, press Enter.
 1=Send  2=Change  3=Hold  4=Delete  5=Display  6=Release  7=Messages
 8=Attributes  9=Work with printing status

Opt  File      User      Device or      Total   Cur
     QPJOBLOG  HANS     QEZJOBLOG  QPADEV0004  RDY    10   Page  Copy
     QPJOBLOG  HANS     QEZJOBLOG  P23XYG41F   RDY     3     1
     QPCSMPT  HANS     PFEIFFER   RDY     4     1
     QPDCLINE  HANS     PFEIFFER   RDY     1     1
     QPJOBLOG  HANS     QEZJOBLOG  P23XYG41    RDY     1     1
     QPJOBLOG  HANS     QEZJOBLOG  P23XYG41F   RDY     7     1
     QPCSMPT  HANS     PFEIFFER   RDY     4     1
     QPCSMPT  HANS     PFEIFFER   RDY     4     1
     QPCSMPT  HANS     PFEIFFER   RDY     4     1
     QPCSMPT  HANS     PFEIFFER   RDY     4     1
     QPCSMPT  HANS     PFEIFFER   RDY     4     1
                                           Bottom

Parameters for options 1, 2, 3 or command
===>
F3=Exit  F10=View 3  F11=View 2  F12=Cancel  F22=Printers  F24=More keys
```

Figure 213. Work with Spooled Files Display

2. Press **F10** to display the job information for the spooled file.

```

Work with All Spooled Files

Type options, press Enter.
 1=Send  2=Change  3=Hold  4=Delete  5=Display  6=Release  7=Messages
 8=Attributes  9=Work with printing status

Opt  File          File          User      Number  Queue      Library
    Nbr  Job
QPJOBLOG  1  QPADEV0004  HANS      058135  QEZJOBLOG  QUSRSYS
QPJOBLOG  1  P23XYG41F  HANS      058220  QEZJOBLOG  QUSRSYS
QPCSMPT  3  P23XYG41F  HANS      058406  PFEIFFER   PFEIFFER
QPDCLINE  4  P23XYG41F  HANS      058406  PFEIFFER   PFEIFFER
QPJOBLOG  1  P23XYG41  HANS      058405  QEZJOBLOG  QUSRSYS
QPJOBLOG  5  P23XYG41F  HANS      058406  QEZJOBLOG  QUSRSYS
QPCSMPT  1  P23XYG41F  HANS      058480  PFEIFFER   PFEIFFER
QPCSMPT  2  P23XYG41F  HANS      058480  PFEIFFER   PFEIFFER
QPCSMPT  3  P23XYG41F  HANS      058480  PFEIFFER   PFEIFFER
QPCSMPT  4  P23XYG41F  HANS      058480  PFEIFFER   PFEIFFER
Bottom

Parameters for options 1, 2, 3 or command
===>
F3=Exit  F10=View 2  F11=View 1  F12=Cancel  F22=Printers  F24=More keys

```

Figure 214. Spooled Files Job Information

3. Locate the spooled file that you want to send to another AS/400 system and make note of the file, job, user name, and job number.

Note: Option 1 (Send) on the *Work with Spooled Files* display applies to sending spooled files in an SNA network (using the SNDNETSPLF command). This option does not apply to sending spooled files in a TCP/IP network.

Step 2 — Start the Spooled File Transfer

Specify SNDTCPSPLF and press **F4** (Prompt). Complete the highlighted fields of Figure 215 on page 348.

Specify the host name or the special value *INTERNETADR to prompt for the Internet address to which the spooled file is transferred. In the printer queue parameter, specify the qualified output queue (library/outqueue) of the remote system to receive that file.

In addition, specify the information from Figure 214 for the specific spooled file that you want to send.

When transferring spooled files from an AS/400 to an AS/400, transformation of the data stream is not necessary.

```

Send TCP/IP Spooled File (LPR)

Type choices, press Enter.

Remote system . . . . . > sysnam123

Printer queue . . . . . > pfeiffer/pfeiffer

Spooled file . . . . . > qpcsmprt      Name
Job name . . . . . > p23xyg41f      Name, *
User . . . . . > hans                Name
Number . . . . . > 058480           000000-999999
Spooled file number . . . . . > 2    1-9999, *ONLY, *LAST
Destination type . . . . . > *as400  *AS400, *PSF2, *OTHER
Transform SCS to ASCII . . . . . > *no *YES, *NO

F3=Exit  F4=Prompt  F5=Refresh  F10=Additional parameters  F12=Cancel
F13=How to use this display  F24=More keys
Bottom

```

Figure 215. Spooled File Job Information in LPR Display

Note: If you create the spooled file with the current interactive job, specify an asterisk (*) for the job name parameter. Leave the user and job number blank.

Note: Use the command line at the bottom of Figure 214 on page 347 to enter the SNDTCPSPLF or LPR command. Specify the spooled file job information as follows:

```
LPR FILE(File) Job(Number/User/Job) SPLNBR(FileNbr)
```

Press **F4** (Prompt). The system fills in the job information in the matching fields of Figure 215.

Sending Spooled Files to an AS/400 at V2R3 or V3R0M5

Using LPR to send spooled files to a V2R3 system or a V3R0M5 system requires that you apply the following PTFs:

- V2R3 – PTF SF16482 or higher
- V3R0M5 – PTF SF16483 or higher

The phrase *or higher* means any PTFs that take the place of these PTFs.

How the System Sends a Spooled File from an AS/400 System to Another AS/400 System

When you use TCP/IP to send a spooled file to another AS/400 system and the file is not transformed to ASCII, implementation-specific extensions to LPR are available to retain the spooled file attributes. LPR still sends the control file because it is part of the protocol, and the receiving AS/400 checks for the control file extended print attribute. From the extended print attribute, the receiving AS/400 determines that another AS/400 system sent the spooled file.

In this case, the system sends all of the spooled file attributes in the data file. It does not use the option and attribute information in the control file. Except for Job ID and output queue, all of the attributes of the spooled file are the same on both the sending and receiving systems. The system sends the job user ID that ran the

SNDTCPSPLF command in the control file. The receiving system creates the spooled file under this user profile with a job name of QPRTJOB. If the user profile does not exist, the system uses the default user profile QTMLPD. You can prevent user IDs that do not exist on the receiving system from sending with LPR by setting the *PUBLIC access of QTMLPD to *EXCLUDE.

When the destination is an AS/400 system, the specified print queue value (PRTQ parameter) can be the name of any defined output queue on the destination AS/400 system. You need to specify the full library name and output queue name when sending to another AS/400 system. If you do not specify a library name, the system searches the library list that is associated with the user ID. If it does not find the output queue or if the user ID is not authorized to it, the system uses the default output queue of QPRINT in library QGPL.

If the file is transformed to ASCII, the extended print attribute is not sent in the control file. The transform program uses the spooled file attributes to produce an ASCII print data stream that is specific to the specified printer model parameter. The spooled file is created on the receiving AS/400 as type *USERASCII with default attributes. Also, the spooled file name is changed to LPDxxxx where x represents any valid hex character. This forms a unique signature that identifies the LPR client that sent the file.

How the System Sends a Spooled File from an AS/400 System to a Non-AS/400 System

When using SNDTCPSPLF or LPR to send a spooled file to a non-AS/400 system, it is sometimes necessary to refer to the LPD documentation for that implementation to determine the print queue value (PRTQ parameter). For example, the print queue value for the OS/2 licensed program is the physical name of the destination printer object on the desktop.

Note: On some systems, the name of the destination printer queue is case sensitive. If the name of the destination printer queue is lowercase or mixed case, enclose the name in apostrophes (' ') to prevent the AS/400 system from making the name uppercase. This also applies to the destination-dependent options.

When sending to non-AS/400 systems, the LPD server reads the control file that contains a number of printing options and attributes, such as width of output and number of copies. Only attributes that the LPR/LPD protocol supports are sent. In particular, the page range-to-print attribute is not supported. The LPD server determines the implementation of the control file attributes. This control file is part of the LPR/LPD protocol, and LPR builds it automatically.

Number of copies

For the number-of-copies attribute to function properly, you must send it in a way that the LPD server recognizes. The default method puts a print command in the control file for each copy requested. A single copy of the data and control files are sent. Some print server LPDs require this method, but most LAN-attached printers ignore the multiple print commands.

The alternate method puts a single print command in the control file. The data and control files are then sent multiple times. To select this method, press **F10** to display

additional parameters and specify the destination-dependent option DESTOPT(XAIX) on the SNDTCPSPLF command. Note that the option is capitalized.

Verify that multiple copies are sent correctly by using the SNDTCPSPLF command with a test spooled file that has the COPIES attribute set to two or more. Sending with the default method when LPD requires the XAIX option prints only a single copy with no error messages. Sending with DESTOPT(XAIX) specified when LPD requires the default method results in the following error message:

```
TCP3701 Send request failed for spooled file ...
```

You can still print a copy after receiving this error message.

Destination-dependent options

You also use this field for specifying attributes of the LPR/LPD protocol. The AS/400 LPR command supports *filters*, which indicate how to print the file (for example, whether the first column is carriage control characters). You can change the default jobname for the separator page by specifying J immediately followed by the desired job name. For a list of the attributes that are sent in the control file, see “LPR Command” on page 345.

The SEPPAGE parameter specifies whether a separator page is requested when the spooled file is printed on the remote system. The default value is *YES. The LPD implementation determines the printing of the separator page. Some LPD implementations ignore this request and print (or do not print) a separator page by default.

Some LPD systems understand additional implementation-specific control file lines. When sending to one of these systems, you can append additional lines to the control file by specifying them on the (DESTOPT) parameter. Separate the options with spaces to send them in the control file exactly as typed. The options must be described in the LPD documentation for that system.

Because other systems might not support the attributes that are associated with a spooled file, the spooled file that is created on the destination system might not be identical to the one that you sent.

Transformation of Spooled Files

If the spooled file has a printer device type of Systems Network Architecture character string (*SCS) or Advanced Function Printing Data Stream (*AFPDS), you can convert the data stream to ASCII using the host print transform (HPT) on the AS/400 prior to sending. Use the TRANSFORM parameter on the SNDTCPSPLF command to convert the data stream to ASCII. For information on host print transform, see the *Printer Device Programming* book. You cannot transform an Intelligent Printer Data Stream (*IPDS). If the printer device type is *IPDS, no conversion takes place, the file is not sent, and an error message is issued.

You also cannot convert files of device type *SCS that contain IPDS* to ASCII. If you attempt to send a spooled file containing IPDS data using the SNDTCPSPLF command, the send request is canceled. An example is some spooled files that you create with the OfficeVision licensed program. These spooled files are intended to print to an IPDS printer and contain IPDS data. Therefore, you cannot convert the spooled files to ASCII.

Although you can send transformed files to another AS/400 (as type *USERASCII), this is more used when the destination system is a non-AS/400 system, using the options DESTTYP(*OTHER) and TRANSFORM(*YES) (assuming that the printer device type of the spooled file is *SCS or *AFPDS).

Most non-AS/400 LPD receivers expect the data in ASCII format. Unless the printer device type attribute is already *USERASCII, you need to transform the file.

If you know that the destination system supports the printer device type of the spooled file, send it without transforming it. For example, you can send spooled files with a printer device type of AFPDS to Print Services Facility/2 (DESTTYP(*PSF2)). When sending *AFPDS files, you must consider any external resources of that file. These resources, such as fonts and overlays, must also reside on the destination system to allow the file to print correctly.

In addition to AFPDS data, you can send *SCS data to PSF/2. When you send *SCS data, you must specify TRANSFORM(*YES) and MFRTYPMDL(*IBM5202). These values result in the best print fidelity.

When specifying that a file must be transformed from *SCS or *AFPDS to *ASCII, you need to specify the type of ASCII printer (MFRTYPMDL parameter). You can also specify a workstation customizing object that affects how the transform is done (MFRTYPMDL(*WSCST)). For more information about the relationship between the host print transform function and the workstation customizing object, see *Workstation Customization Programming*.

You can also use your own program to handle transformation of the data. User data transform programs must be written to the *Writer Transform Exit Program* interface. For more information on this interface, see *Writer Transform Exit Program* in the *System API Reference*. The name and library of this program are entered in the USRDTATFM parameter. This parameter is prompted only when you select TRANSFORM(*NO) to not use the AS/400 host print transform.

Table 32. Common Parameter Settings When Using SNDTCPSPLF

Destination System Type	Data Stream Type	Transform Value	Manufacturer Type and Model
*AS400	*SCS or *AFPDS	*YES	Specify type of ASCII printer
*AS400	Any	*NO	Leave blank
*PSF2	*AFPDS	*NO	Leave blank
*PSF2	*SCS	*YES	*IBM5202 ¹
*PSF2	*USERASCII ²	*NO	Leave blank
*OTHER	*SCS or *AFPDS	*YES	Specify type of ASCII printer
*OTHER	*USERASCII ²	*NO	Leave blank

Notes:

1. Choose the highest function printer supported by PSF/2.
2. *USERASCII does not necessarily mean that the data stream for the spooled file is ASCII. It means that the data was spooled without being examined or validated by an AS/400 business computing system.

LPR Support of PostScript Printers

The HPT on the AS/400 converts an SCS or AFPDS spooled file to ASCII format. The *Image Print Transform* added in Version 4 Release 2 converts image or PostScript *USERASCII spooled files into various ASCII formats and non-ASCII

formats. Setting TRANSFORM(*YES) enables either the Image or Host Print Transform, depending on the spooled file type.

When you convert PostScript spooled files, text data is converted into a bit-mapped image. Conversely, only image spooled files are converted into PostScript. Image or PostScript spooled files are also converted into Printer Control Language (PCL) for Hewlett-Packard compatible printers and Advanced Function Printing (AFP) data stream.

Image transformation requires additional image configuration information beyond that of the manufacturing type and model parameter, or workstation customizing object. When using the LPR or SNDTCPSPLF command, you must ensure that this information is in the user-defined data attribute of the spooled file. When using printer pass-through, you can specify this with an image configuration object in the IMGCFG parameter of the remote output queue. For detailed information on specifying Image and Host Print Transform parameters, see *Printer Device Programming*, SC41-5713-03.

Determining the Printer Device Type

To determine the printer device type of a spooled file, display the spooled file attributes. The printer device type (DEVTYPE) parameter on the printer file sets this attribute when the spooled file is created. For more information, see *Printer Device Programming*.

Printer Requirements when Sending with Host Print Transform Function

You need to know the printer, the manufacturer, type, and model (MFRTYPMDL parameter) of the destination printer if you are using the host print transform function. Many printers sold by the following companies are supported:

- Epson America Incorporated
- Hewlett-Packard Company
- IBM Corporation
- NEC Corporation
- Okidata Corporation
- Panasonic Corporation
- Xerox Corporation

Authority for Sending Spooled Files

To send a spooled file, you must have one of the following authorities to the file or to the output queue that the file is on.

- Be the owner of the spooled file.
- Have spool control authority (*SPLCTL).
- Have job control (*JOBCTL) special authority on an operator-controlled (OPRCTL(*YES)) output queue.
- Be the owner of the output queue.
- Have add, delete, and read authority to an output queue created with AUTCHK(*DTAAUT).
- Have read authority to output queue created with DSPDTA(*YES).

If you are using LPR through the Remote Writer, then the Remote Writer changes its job attributes to run under the user profile that created the spooled file. The writer does not change its job attributes to those of a system profile. Instead, it uses the default user profile of QSPLJOB. In any case, the Remote Writer always passes the authority test.

Sending Spooled File — Tips

You can send only printer spooled files. You cannot send diskette spooled files.

You cannot send a spooled file that is in an open status.

Successful sending of a spooled file only acknowledges a successful transmission. At that point, the TCP/IP connection is closed. If any errors occur while the LPD server processes the file, messages cannot be sent back. You must not delete any files without verifying that the spooled file was created or printed successfully.

Sending Large Spooled Files

Depending on system load, how your printer is configured, and other variables, you might need to disable or increase the idle timeout value on the printer to print large spooled files. This value is set differently, depending on what kind of printer you are using. The procedure for changing the timeout and range of settings is device dependent. For more information, see the documentation provided with the printer. On some printers, this is set on the control panel. On others, it is set using a Telnet command. You need a large timeout value because the printer connection is opened before the file transforms to ASCII. Otherwise, the system resources used in transforming are wasted if the printer is in use by another job when the transform completes. You cannot send the file until the transform is complete because the LPR/LPD protocol requires the system to send the total byte count before any data.

You can also transform the spooled file to ASCII before sending to the printer by sending it to the AS/400 LPD server. Specify `DESTTYP(*AS400)` `TRANSFORM(*YES)` on the `SNDTCPSPLF` command along with the desired ASCII printer model and options. Specify the local host name for the remote system and an output queue on the AS/400. Use a second `SNDTCPSPLF` command to send the transformed spooled file to the desired printer. This significantly reduces the wait time seen by the printer. You can also use this with printer pass-through (below) by creating two output queues.

You can accomplish this more efficiently with printer pass-through (below) by adding the destination option `XAUTOQ` with the `CHGOUTQ` or `CRTOUTQ` commands. Note the option must be capitalized. The file is transformed and sent to the remote system as specified by the other remote output queue parameters unless the idle timeout value is exceeded during transformation. In this case, the transformed spooled file is routed instead to the AS/400 LPD server and sent back to the same output queue. From there it is automatically sent again to the remote system. The Host Print Transform recognizes that the spooled file is already transformed and sends the spooled file without that delay.

Printer Pass-Through

Printer pass-through enables you to route printer files automatically to another system that supports TCP/IP LPD.

It is possible to do printer pass-through over both SNA and TCP/IP. This section deals purely with TCP/IP considerations.

Setup

To enable printer pass-through, you need to create an output queue with CRTOUTQ and specify a RMTSYS parameter other than *NONE. Specify either the remote system (host) name or *INTNETADR, in which case the system prompts you for the Internet address of the remote system.

The following parameters on the CRTOUTQ command are relevant to using printer pass-through with LPR.

- To use printer pass-through with TCP/IP, you need to specify CNNTYPE of *IP.
- Using the RMTprtQ parameter, specify a specific print queue other than *USER or *SYSTEM on the remote system. If you send to another AS/400, specify only a print queue name or libname/outqname. In this case, the system expects to find the remote output queue in QUSRSYS.
- Specify a DESTTYPE of *OS400 (for OS/400 V3 or V2R3 with TCP/IP), *PSF2 (for a PC with PSF/2), or *OTHER. OS/400 TCP/IP prior to V2R3 does not support LPD.
- Check the TRANSFORM parameter. By default, this is *YES, to use *Host Print Transform*. If you want the system to prompt you for the user transform program parameters, specify *NO. For sending to another AS/400, use *NO with user transfer program *NONE.
- You can send additional control file options to the remote system by specifying them on the DESTOPT parameter. CRTOUTQ (Create Output Queue) also supports the special value *USRDFNTXT. This takes the value of the DESTOPT parameter from the user-defined text of the user profile when the system creates that spooled file. The separator page parameter of the SNDTCPSPLF command is also supported.

```

                                Create Output Queue (CRTOUTQ)

Type choices, press Enter.

Output queue . . . . . OUTQ          > SYSNAM123
Library . . . . .                   *CURLIB
Maximum spooled file size:          MAXPAGES
Number of pages . . . . .           *NONE
Starting time . . . . .
Ending time . . . . .
                                + for more values
Order of files on queue . . . . . SEQ    *FIFO
Remote system . . . . . RMTSYS        > SYSNAM123

Remote printer queue . . . . . RMTPRQ    SYSNAM456

Writers to autostart . . . . . AUTOSTRWTR *NONE

F3=Exit   F4=Prompt   F5=Refresh   F10=Additional parameters   F12=Cancel
F13=How to use this display           F24=More keys
More...

```

Figure 216. Create Remote Output Queue—Display 1 of 2

```

                                Create Output Queue (CRTOUTQ)

Type choices, press Enter.

Queue for writer messages . . . . . MSGQ    QSYSOPR
Library . . . . .                   *LIBL
Connection type . . . . . CNNTYPE        > *IP
Destination type . . . . . DESTTYPE     *OS400
Transform SCS to ASCII . . . . . TRANSFORM *NO

```

Figure 217. Create Remote Output Queue—Display 2 of 2

You now need to ensure that the remote system's print queue exists before starting the pass-through process.

This completes the setup processing.

Starting Printer Pass-Through

To start printer pass-through, perform the following steps:

1. Use the *Start Remote Writer* (STRRMTWTR) command.
2. Specify the queue name on the local (LPR) system that you created in the previous section. This starts a writer in the QSPL subsystem that, when using TCP/IP, uses LPR to send printer files to the remote (LPD) system.
3. Ensure that LPD is operational on the remote system and, in the case of the AS/400, that the remote printer writer is active.

In the event of problems with this processing, the writer's job log is often a useful source of problem analysis material. To access this while the job is running, specify the WRKACTJOB SBS(QSPL) command and select Option 5 against the writer job.

This shows the *Work with Writers* display, from which it is necessary to press **F17** to see the job log. This sequence is different from the usual one for viewing batch and interactive job logs while the jobs are running.

Configuring for a RISC System/6000 System — Scenario

This section describes the following topics:

- How to configure LPD on the RS/6000 system to allow an AS/400 to send a spooled file with LPR.
- How to configure device and virtual printers for AIX print output.
- How to configure a PSF/6000 printer for PSF/6000 print output.

Setting Up for LPD on the RISC System/6000 System — Scenario

Use the following series of commands to follow through the SMIT menus on the RS/6000 system.

At the AIX command line prompt, specify the following:

```
smit
  Spooler (Print Jobs)
    Manage Remote Printer Subsystem
      Server Services
        Host Access for Printing
          Add a Remote Host
```

Specify the host name of the remote host that you want to send spooled files to the local LPD as shown in Figure 218. In this case, it is SYSNAM890. This name must also be in the local host table of the RS/6000 system (*/etc/hosts*).

```
                                Add a Remote Host

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

* Name of Remote HOST                                [Entry Fields]
                                                    sysnam890

F1=Help      F2=Refresh    F3=Cancel    F4=List
Esc+5=Reset  F6=Command    F7=Edit      F8=Image
F9=Shell     F10=Exit     Enter=Do
```

Figure 218. Add a Remote Host

Configuring Device and Virtual Printer for AIX Printing

Two steps are necessary for the following configurations on the RS/6000 system.

Configuring the AIX Printer Device

To configure a non-IPDS printer so that the system recognizes it as an AIX printer, specify the following at the AIX command line prompt (in this case, an IBM 4029 printer is attached in parallel):

```
smit printer
  Printer/Plotter Devices
    Add a Printer/Plotter
```



```

4029      IBM 4029 LaserPrinter (select printer, press Enter)
parallel (select Interface, press Enter)
ppa0 Available 00-00-0P Standard I/O Parallel Port Adapter
p        (select PORT number)

```

Figure 219 shows the final display. Make any changes if necessary. If all the fields are correct, press the Enter key.

```

                                Add a Printer/Plotter

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

[TOP]                                [Entry Fields]
Printer/Plotter type                 4029
Printer/Plotter interface            parallel
Description                          IBM 4029 LaserPrinter
Parent adapter                       ppa0
* PORT number                        [p] +
Type of PARALLEL INTERFACE          [standard] +
Printer TIME OUT period              [600] +#

STATE to be configured at boot time  [available] +
The following attributes have meaning only
when the Printer/Plotter is not used with
a Print Queue:

[MORE...13]                          F2=Refresh          F3=Cancel

Es=Help          F2=Refresh    ext F3=Cancel    F4=List
F9=Shell        F10=Exit       Enter=Do

```

Figure 219. Printer Specifications

Configuring a Virtual Printer

To configure a virtual printer, specify the following at the AIX command line prompt:

```

smit printer
  Manage Local Printer Subsystem
    Virtual Printers
      Add a Virtual Printer

```

```

No.  Description
1    Printer or Plotter Attached to Host
2    Printer or Plotter Attached to Xstation
3    Printer or Plotter Attached to ASCII Terminal
4    Network Printer (Hewlett-Packard JetDirect)

Enter number from list above (press Enter to terminate): -> 1

```

Figure 220. Configuring a Virtual Printer

1. Select Option 1 (Printer or plotter attached to host).

```

Name          Description
lp0           IBM 4029 LaserPrinter

Enter device name (or, ! to exit): -> lp0

```

Figure 221. Enter the Printer Device Name Used and Press Enter

2. Specify the printer device name used and press the Enter key.

For the next series of displays, answer the questions. When in doubt, take the defaults.

Specify the print queue name to use for printing with AIX. In this case, the name is asc.

```

                IBM 4029 LaserPrinter
Header pages wanted? (n=none; a=each file; g=each job): -> (n)
Trailer pages wanted? (n=none; a=each file; g=each job): -> (n)
NOTE:   The 4029 printer supports multiple print data streams.
        Each of the data streams will now be configured individually.
----- PostScript -----
Enter print queue name (or, ! to bypass configuration): -> (ps)  !
----- HP LaserJet II Emulation -----
Enter print queue name (or, ! to bypass configuration): -> (pcl)  !
----- Plotter Emulation -----
Enter print queue name (or, ! to bypass configuration): -> (gl)  !
----- IBM ASCII -----
Enter print queue name (or, ! to bypass configuration): -> (asc)  asc
Should this queue be the default queue? -> (y)  n
4029 (IBM ASCII) configured for print queue asc

Press Enter to continue

```

Figure 222. Virtual Printer Queue Configuration

Verifying LPD Started on the RISC System/6000 System

Verify that LPD is started on the RS/6000 system. Check its status by issuing the following command:

```
'lssrc -s lpd'
```

If LPD is inactive, you need to start the daemon. If you restart the system itself, ensure that LPD starts again.

To use SMIT to start LPD either now or at system restart time, specify the following at the AIX command line prompt:

```

smit
  Communications Applications and Services
  TCP/IP
  Further Configuration
  Remote Printer Subsystem
  Server Services
  lpd Remote Printer Subsystem
  Start Using lpd Subsystem
  Start BOTH Now and at System Restart

```

Verifying Your Configuration on the RISC System/6000 System

1. Verify the printer configuration by printing an RS/6000 file directly to the attached printer. Specify the following:

```
cat /etc/qconfig > /dev/lp0
```
2. Test that the virtual printer works correctly by specifying the following:

```
enq -P asc /etc/qconfig
```
3. Use LPR on the AS/400 system to send a spooled file to the RS/6000 system to be printed using the AIX print queue asc. Find a spooled file that you want to send and write down the file, job, and spool number information. Next, complete the LPR command as shown in the following example, using the file, job, and spool number of your spooled file.

Note: It is important to enclose the PRTQ parameter asc within apostrophes. This is because the RS/6000 system is case sensitive. Send the print queue name asc as it appears in lowercase.

```
LPR RMTSYS(RCHRS001) PRTQ('asc')
FILE(FTPBATCT)
JOB(059016/HANS/FTPBATCT)
SPLNBR(1) DESTTYP(*OTHER)
TRANSFORM(*YES) MFRTYPMDL(*IBM42011)
```

Printer not Active — Symptom

When you specify `enq -A` at the RS/6000 system to display the status of the print queue, the printouts to be printed are queuing in that print queue. If the printer has been offline for some time, the status of that queue could become DOWN.

The following are possible causes for the DOWN status:

- Printer is offline.
- Printer status is DOWN.
- Printer paper jam.

To correct the possible causes, perform the following steps:

1. Press the `online` button at the printer.
2. If the print queue status shows DOWN, specify the following RS/6000 command to get the print queue to READY status again:

```
qadm -U asc
```

where asc is the print queue name.

3. To clear the paper jam and get the printer ready for printing again, perform the following steps:

- a. Clear the paper in the printer.
- b. Display the status of the print queue using the following command:

```
enq -A
```
- c. To clear any outstanding print jobs in the print queue, if necessary, specify the following command:

```
qcan -xnn
```

where nn is the job number in the status screen.

- d. If the status of the printer is DOWN, specify the following command:

```
qadm -U asc
```

- where asc is the print queue name.
- e. Check the status of the print queue to ensure that the status is READY and that the printer is online.

Print Services Facility/6000 Function

The PSF/6000 licensed program provides an intelligent print driver that accepts *input* data stream files, including Advanced Function Printing data stream (AFPDS) and ASCII. The program also produces different *output* data streams like Intelligent Printer Data Stream (IPDS), IBM personal printer data stream (PPDS), or Hewlett-Packard printer control language (PCL).

The PSF/6000 function allows you to perform the following tasks:

- Integrate workstation and host printing under a common architecture.
- Enable LAN print serving capability supporting multiple workstations, operating systems, data streams, and print types.
- Improve systems management of printing by using Network File System (NFS).

Configuring PSF/6000 Function

This scenario configures for PPDS. To configure for other scenarios, refer to the *IBM AIX PSF/6000: Print Services for Users of AIX, G544-3814-01*.

To use the PSF/6000 function, configure a PPDS printer (or an IPDS printer or a PCL printer). Specify the following:

```
smit psfcfg
  Add a Printer or PSF/6000 Queue
    AIX-Defined (Parallel, Serial, or LAN)
```

The system prompts you to choose a data stream type (PPDS, PCL4, PCL5, or PCL5C).

Following this, specify the information as shown in Figure 223.

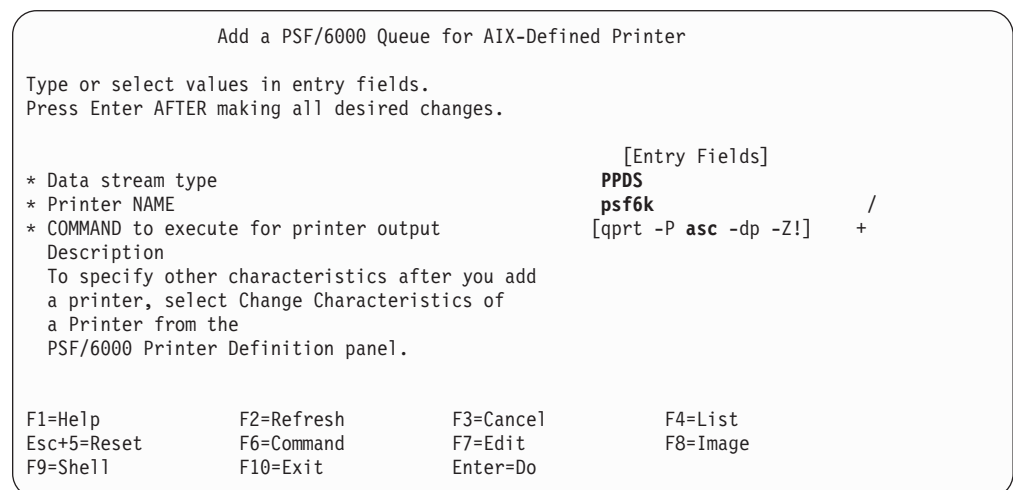


Figure 223. PPDS Printer and Queue Definition

Specify the name of the PPDS printer. This name is equal to the PPDS queue that you want to create.

For the COMMAND to run for the printer output field, specify the name of the AIX output queue asc, not the name of the PSF queue.

Verifying Your Configuration of PSF/6000

To test that the path between the RS/6000 system, PSF/6000 queue, and the PSF/6000 printer is correct, specify the following:

```
enq -P psf6k /etc/qconfig
```

where psf6k is the name of the PSF/6000 queue.

If the file prints correctly, you can send and print AS/400 spooled files with AFP and SCS data stream on the RS/6000 printer.

When sending spooled files to PSF/6000, always specify *OTHER for the LPR destination type parameter.

For AFPDS, specify *N0 for the transform SCS to ASCII parameter. We used the AFP Utilities licensed program with the *Start AFP Utilities (STRAFPU)* command on the AS/400 system to create an overlay that is named OVERLAY. It is this AFPDS spooled file that you sent to the RS/6000 RCHRS001 in Figure 224. PSF/6000 converts the AFPDS data stream into PPDS automatically. For more information about the AFP Utilities licensed program, see *AFP Utilities/400 User's Guide and Reference*, SH18-2416.

```
Send TCP/IP Spooled File (LPR)

Type choices, press Enter.

Remote system . . . . . rchrs001

Printer queue . . . . . 'psf6k'

Spooled file . . . . . overlay      Name
Job name . . . . . p23xyg41f      Name, *
User . . . . . hans                Name
Number . . . . . 060045           000000-999999
Spooled file number . . . . . *ONLY 1-9999, *ONLY, *LAST
Destination type . . . . . *OTHER  *AS400, *PSF2, *OTHER
Transform SCS to ASCII . . . . . *no  *YES, *NO

Bottom

F3=Exit  F4=Prompt  F5=Refresh  F10=Additional parameters  F12=Cancel
F13=How to use this display  F24=More keys
```

Figure 224. Sending AFPDS to PSF/6000

For the SCS data stream, you must specify *YES for the Transform SCS to ASCII parameter. The AS/400 converts the SCS data stream from EBCDIC to ASCII (by

host print transform), which is then sent to PSF/6000 and properly printed. See Figure 225 for an example.

```
Send TCP/IP Spooled File (LPR)

Type choices, press Enter.

Remote system . . . . . > RCHRS001

Printer queue . . . . . > 'psf6k'

Spooled file . . . . . > ftpbatcht      Name
Job name . . . . . > ftpbatcht         Name, *
  User . . . . . > HANS                 Name
  Number . . . . . > 059016            000000-999999
Spooled file number . . . . . *ONLY     1-9999, *ONLY, *LAST
Destination type . . . . . *OTHER       *AS400, *PSF2, *OTHER
Transform SCS to ASCII . . . . . > *yes *YES, *NO

Bottom
F3=Exit  F4=Prompt  F5=Refresh  F10=Additional parameters  F12=Cancel
F13=How to use this display  F24=More keys
```

Figure 225. Sending SCS to PSF/6000

Chapter 12. Line Printer Daemon (LPD)

You can request to have your spooled files printed on any system in your TCP/IP network.

The printing facilities of the destination system handle the printing of the file. The destination system must be running TCP/IP. On the AS/400 system, the **line printer daemon (LPD)** is the process on the destination system that receives the file that the SNDTCPSPLF command sends. The LPD process places the spooled file on a local printer queue. To print the spooled file, place it on a printer queue that is already started to an active printer writer. Another method of printing a spooled file is to start a writer to that printer queue.

Configuring for Line Printer Daemon (LPD)

LPD does not require any special setup when receiving printer files from another AS/400 system. However, with the *Change LPD Attributes (CHGLPDA)* command, you can specify the number of LPD servers that you want initially started. You can also specify whether you want these servers to start whenever you issue the *Start TCP/IP (STRTCP)* command (Figure 226). Access the LPD attributes with either the *Configure TCP/IP LPD (CFGTCPLPD)* command or the CHGLPDA command.

```
Change LPD Attributes (CHGLPDA)

Type choices, press Enter.

Autostart servers . . . . . *YES          *YES, *NO, *SAME
Number of initial servers . . . 2          1-20, *SAME, *DFT

F3=Exit  F4=Prompt  F5=Refresh  F12=Cancel  F13=How to use this display
F24=More keys

Bottom
```

Figure 226. Change LPD Attributes (CHGLPDA) Display

How the Destination System Receives a Spooled File

The LPD server jobs run in subsystem QSYSWRK. Display these jobs using the *Work with Active Jobs (WRKACTJOB)* command. Identify them by their job names in the format QTLPDnnnnn.

```

                                Work with Active Jobs                                ENDAS011
                                                                                   05/11/94 07:26:02
CPU %:   4.5   Elapsed time: 00:02:02   Active jobs: 51

Type options, press Enter.
  2=Change  3=Hold  4=End  5=Work with  6=Release  7=Display message
  8=Work with spooled files 13=Disconnect ...

  Opt Subsystem/Job  User      Type  CPU %  Function
  Status
      QTGTELNETS   QTCP      BCH    .0      DEQW
      QTMSBRCL    QTCP      BCH    .0      PGM-QTMSBRDG  DEQW
      QTMSBRSR    QTCP      BCH    .0      PGM-QTMSBRSR  TIMW
      QTMSNMP     QTCP      BCH    .0      PGM-QTOSMAIN  DEQW
      QTMSNMPRCV  QTCP      BCH    .0      PGM-QTOSRCVR  TIMW
      QTLPD66887  QTCP      BCH    .0      TIMW
      QTLPD67093  QTCP      BCH    .0      DEQW

                                                                                   More...

Parameters or command
====>
F3=Exit      F5=Refresh  F10=Restart statistics  F11=Display elapsed data
F12=Cancel   F23=More options  F24=More keys

```

Figure 227. Viewing the LPD Servers from the Work with Active Jobs Display

Only one LPD server job at a time listens for a valid LPR request from a remote system. When an LPR request comes in, that job handles the request, and attempts to pass the *listening* socket to another waiting LPD server job.

If LPD passes the socket successfully, it submits a replacement LPD server job using the SBMJOB command and ends when the current job is complete. If LPD cannot pass the listening socket (if only one server is running, for example), it runs in single server mode.

In single server mode, additional LPR requesters must wait for any currently running requests to reach a checkpoint stage or complete processing before their request is handled. Because the system cannot pass the listening socket to a waiting LPD server, the current LPD server continues to run instead of ending, and no replacement job is submitted.

To avoid LPR request failures while a job is running, start LPD with a minimum of two servers configured. The wait for a checkpoint stage or processing to complete is often significant, especially if you are receiving a large print job.

How an AS/400 System Receives a Spooled File from Another AS/400 System

If another AS/400 system sends the spooled file, the first file to arrive is the control file. This is because AS/400 LPD uses the newer RCFF (Receive Control File First) and RDFUL (Receive Data File of Unspecified Length) protocols that RFC 1179 describes. If you send the spooled file without transformation from SCS to ASCII, the control file contains the extended print attributes. The AS/400 LPD process checks the extended print option flag and determines that the spooled file attributes are being sent with the data.

In this case, the control file printing command flags are ignored, and the spooled file is created on the destination AS/400 using the spooled APIs. For information on spooled file and print APIs, see *System API Reference*.

If the original spooled file is in SCS format and the TRANSFORM parameter on the SNDTCPSPLF command is set to *YES, the system transforms the file to ASCII. It accomplishes this by using the host print transform function on the client (LPR) system before it sends the file. The system uses the manufacturer, type, and model information given on the MFRTYPMDL parameter to determine which printer attributes to use in the ASCII data stream. The destination system receives the ASCII data stream without change. Because the file is received in *USERASCII format, the destination AS/400 cannot read it. You need to print the spooled file to see the contents.

Spooled file APIs are used to receive files from remote AS/400s. When these APIs create copies of a spooled file, they require access to the original printer file, which might not exist on the target system.

If the access to the printer file is restricted from the user or the default user QTMPLPD, then errors might occur. Because this is an API limitation, it is currently considered an LPD restriction.

In addition, this also makes *Send TCP/IP Spooled File* (SNDTCPSPLF) work consistently in the same manner as SENDNETSPLF, which also requires the original printer file.

How an AS/400 System Receives a Spooled File from a Non-AS/400 System

If the spooled file is being sent from any system other than an AS/400, AS/400 LPD accepts the request using either the newer RCFF and RDFUL protocol, or the older RDF (Receive Data File) and RCF (Receive Control File) protocol. Some LPR clients do not support the new protocol, but the AS/400 LPD server accepts either format. From the control file information, the AS/400 can establish that the file is not being sent from another AS/400. In this case, the AS/400 assumes that no spooled file attributes are being sent with the data.

Because the AS/400 requires that each spooled file have attributes associated with it, default attributes are given to the file using a default printer file, QTMPLPD, in the library QTCP. The printer file QTMPLPD in the library QTCP contains the default installation values. There is another version of the printer file in library QUSRSYS that should be used if you want to customize the printer file. Using the version that exists in QUSRSYS preserves the default installation values.

For non-AS/400 LPR requesters, you obtain best print results by understanding that minimal formatting of the data file occurs, other than what is provided by the destination printing system. The AS/400 LPD server does provide support for the print filters *v*, *l*, and *f*, as described in RFC 1179. If the *f* filter is received, line feeds (LF) are replaced with carriage return/line feeds (CRLF). With the *f* filter, all printer control characters are deleted from the data stream, with the exception of HT, CR, FF, LF, BS, and ESC characters. Printer control characters are any characters below hexadecimal 20. Since a parameter usually follows the ESC character, the system also keeps the character immediately following the ESC.

The default filter is *l*. There is no difference between the *l* and *v* filters. The system treats them both the same.

LPD does not support control file support for banner pages because the AS/400 has its own separator page function.

If the non-AS/400 LPR requests multiple copies, the system changes the attributes of one spooled file to show multiple copies.

How Spooled Files are Named on the Destination AS/400

Prior to V3R1M0, if someone used AS/400 LPR with the DESTTYP(*OTHER) and TRANSFORM(*YES) parameters, the name of the created spooled file became that of the PRTF file, QPTMPLPD. The original spooled file name was placed into the user data field of the created file. Likewise, for non-AS/400 LPR clients, the system names the newly created file QPTMPLPD. The original file name was placed into the user data field.

Beginning with V3R1M0, spooled files that LPD receives have file names in the form LPDxxxx. The *x* represents any valid hex character. These hex characters are the result of cyclic redundancy checking that is performed on client information to identify the LPR client for LPRM support. Spooled files that are named in this manner are unique to each client. The system uses the name as a signature. All files from a single client have the same signature. A client must generate a matching signature to use LPRM to delete any LPR spooled file.

Clients can use LPQ (line printer queue) and LPRM (line printer removal) commands to query and remove LPR spooled files, as AS/400 LPD supports both functions. However, the AS/400 system that acts as a client cannot issue these commands.

If the LPR client is another AS/400 using DESTTYP(*AS400) and TRANSFORM(*NO) parameters, the spooled file has exactly the same attributes on the receiving queue that it had on the sending queue. The spooled file name is not converted to LPDxxxx form and is left unchanged.

The LPQ command sent to an AS/400 system requires a job list parameter, or, more specifically, a user profile under which the query is performed. The following is an example on OS/2:

```
lpq -pmylib/myoutq -sas400.endicott.ibm.com
  ProfileName
```

Starting an LPD Server Job

You must start the server job for the LPD application in the QSYSWRK subsystem. The *Start TCP/IP Server* (STRTCPSVR) command starts the LPD server that is shipped with the TCP/IP Utilities licensed program. You can also start the LPD server job with the AUTOSTART parameter of the *Change LPD Attributes* (CHGLPDA) command. However, the STRTCPSVR command overrides or ignores the AUTOSTART parameter on the CHGLPDA command.

The *Start TCP/IP* (STRTCP) command starts *all* of the servers that you specify in the CHGLPDA command. The STRTCPSVR command starts the number of servers configured in the CHGLPDA command. After you start the minimum number of LPD servers, the STRTCPSVR command starts only *one* additional server at a time.

LPD works most efficiently when two or more servers are running. It is possible to run only one server, but LPD cannot receive any jobs while a current job is running. If a large print job is running, LPR clients have to wait until LPD is ready to accept any new LPR requests.

You can use the *Configure TCP/IP LPD* (CFGTCPLPD) command or the CHGLPDA command to work with the LPD attributes.

Ending an LPD Server Job

To end server jobs, use the *End TCP/IP Server* (ENDTCPSVR SERVER(*LPD)) command. This command also ends any active LPD jobs that are still processing an LPR request.

Attributes of the Received Spooled File

On the AS/400 system all files that are printed have spooled file attributes associated with the file. These attributes specify such things as lines per inch, page widths, number of copies, and which fonts to use. If any file is received that does not have attributes associated with it, default attributes are given to the file. Set these attributes by using the first QTMPLPD printer file that you find in the library list (*LIBL). Files sent from non-AS/400 systems or files sent using the SNDTCPSPLF command and that were transformed do not have attributes associated with them.

The QTMPLPD printer file can have copies in the QUSRSYS library or other libraries. The copy of the printer file in QTCP is the default if there are no others in the library list (*LIBL). This is the case even if the QTCP library is *not* in the library list (*LIBL). The LPD processing adds the QTCP library to the library list if it is not already in the list. An exception exists where a user has 25 libraries in the user portion of the library list. In this situation, the system replaces the twenty-fifth library with the QTCP library to ensure that it finds a copy of the QTMPLPD file.

The QTMPLPD printer file has a printer device type of *USERASCII because LPD expects that data from non-AS/400 systems is ASCII data. If LPD receives data that is not ASCII data from a non-AS/400 system, LPD spools the data as *USERASCII. However, the data might not print correctly.

For the control file options and attributes that the AS/400 LPD function supports, see Table 36 on page 369.

User Profile Library Lists

All user profiles have a library list associated with them. When LPD creates a spooled file on behalf of a user profile, it searches for the output queue requested using the library list for that profile. Therefore, if the requested output queue does not have a library qualifier (library/queuname), the library list is searched for in the queue. If LPD does not find the queue in the library list, it uses the default output queue QPRINT in library QGPL.

Important!:

The library list of a user profile is the library list that is in effect at sign-on time. Any libraries added by initialization programs or other methods are *not* considered in the library list or associated with that user profile.

Changing the QPTMPLPD Printer File Default Values

To change the default values in the QPTMPLPD printer file, use the *Change Printer File* (CHGPRTF) command. For example, to change the number of lines per page for all files received from non-AS/400 systems, specify the following:

```
CHGPRTF FILE(QUSRSYS/QPTMPLPD) PAGESIZE(60)
```

The change takes effect immediately and remains in effect until other changes are made or until the printer file is created again. LPD overrides the following CHGPRTF parameters, which you cannot change with the CHGPRTF command:

- FILE
- TOFILE
- PAGESIZE(*N n) (width option only is overridden)
- SPLFNAME
- OUTQ
- COPIES
- USRDTA

Note: The installation programs copy the QPTMPLPD printer file from the QTCP library into the QUSRSYS library. That means two versions of this printer file exist, with one in the QTCP library and one in the QUSRSYS library. To preserve the installation values, change the QUSRSYS version.

LPD-Supported Commands on the AS/400

Table 33 identifies what LPD commands that the AS/400 system supports.

Table 33. LPD Commands on the AS/400

Command	LPD	Hex Code
Print Any Waiting Jobs	X	X'01'
Receive a Printer Job	X	X'02'
Send Queue State (Short)	X	X'03'
Send Queue State (Long)	X	X'04'
Remove Jobs	X	X'05'

If the system receives any other commands, LPD ignores them and sends a return code of 1, meaning failure, to the requesting system. The connection is closed immediately. LPD logs an error that indicates unsupported function.

Table 34 identifies which LPD subcommands the AS/400 system supports.

Table 34. LPD Subcommands

Subcommand	LPD	Hex Code
Abort Job		X'01'
Receive Control File	X	X'02'
Receive Data File	X	X'03'

Table 34. LPD Subcommands (continued)

Subcommand	LPD	Hex Code
Receive Control File First	X	X'04'
Receive Data File with Unspecified Length	X	X'05'

LPD-Supported Commands for Control File Printing

Table 35 identifies the LPD control file printing commands that the AS/400 system supports. A **control file** is used to pass all of the printing options and attributes that were selected on the SNDTCPSPLF command to the LPD process on the destination system. You cannot update the control file.

Table 35. Control File Printing Commands

Command	LPD	Letter Code
Plot CIF (CalTech Intermediate file) File		c
Print DVI (device-independent) File		d
Print Formatted File	X	f
Print Plot File		g
Reserved		k
Print File Leaving Control Characters	X	l
Print ditroff (device-independent troff) Output File		n
Print File with PR Command Format		p
Print File with FORTRAN Carriage Control		r
Print troff Output File		t
Print Verbatim File	X	v
Extended Print Command		x

LPD-Supported Control File Options and Attributes

Table 36 identifies the LPD control file options and attributes that the AS/400 system supports. The attributes that the AS/400 implementation of LPD does not support are ignored if sent with the spooled file.

The control file and attribute information are not used when sending files from one AS/400 system to another if the file was not transformed before sending. In this case, all the attributes of the spooled file on the receiving system are the same as they were on the sending system.

Table 36. Control File Options and Attributes

Option or Attribute	LPD	Letter Code
Class for banner page		C
Host name	X	H
Indent printing		I
Job name for banner page		J
Print banner page		L
Mail when printed		M
Name of source file ¹	X	N

Table 36. Control File Options and Attributes (continued)

Option or Attribute	LPD	Letter Code
User identification	X	P
Symbolic link data		S
Title for pr command		T
Unlink data file		U
Extended print option ²		X
Width of output	X	W
troff R font (Times Roman type style)		1
troff I font (Times Roman type style)		2
troff B font (Times Roman type style)		3
troff S font (Times Roman type style)		4
Notes:		
1. Only used to fill the user data (USRDATA) field of the spooled file created.		
2. This command is only used when the destination system is an AS/400 system.		

How the Ownership of Spooled Files Is Determined

If the user ID on the sending system exists on a destination AS/400 system, the spooled file is created under that user profile. However, if the user profile does not exist on the destination system, then the spooled file is created under the QTMPLPD user profile.

When sending from a non-AS/400 system to an AS/400 system, the file is always created using the QTMPLPD printer file. This is also true when sending from an AS/400 to an AS/400 with the TRANSFORM(*YES) parameter. In these cases, spooled files received by the LPD server are placed under special jobs.

For example, if user ID JOHN exists, the file is placed under job 999999/JOHN/QPRTJOB. If user ID JOHN does not exist, the file is placed under 999999/QTMPLPD/QPRTJOB. For more information on QPRTJOB, see the *Printer Device Programming*, SC41-5713.

How an LPD Server Selects an Output Queue for a File

Figure 228 on page 371 shows how the AS/400 LPD server selects on which output queue a file is placed.

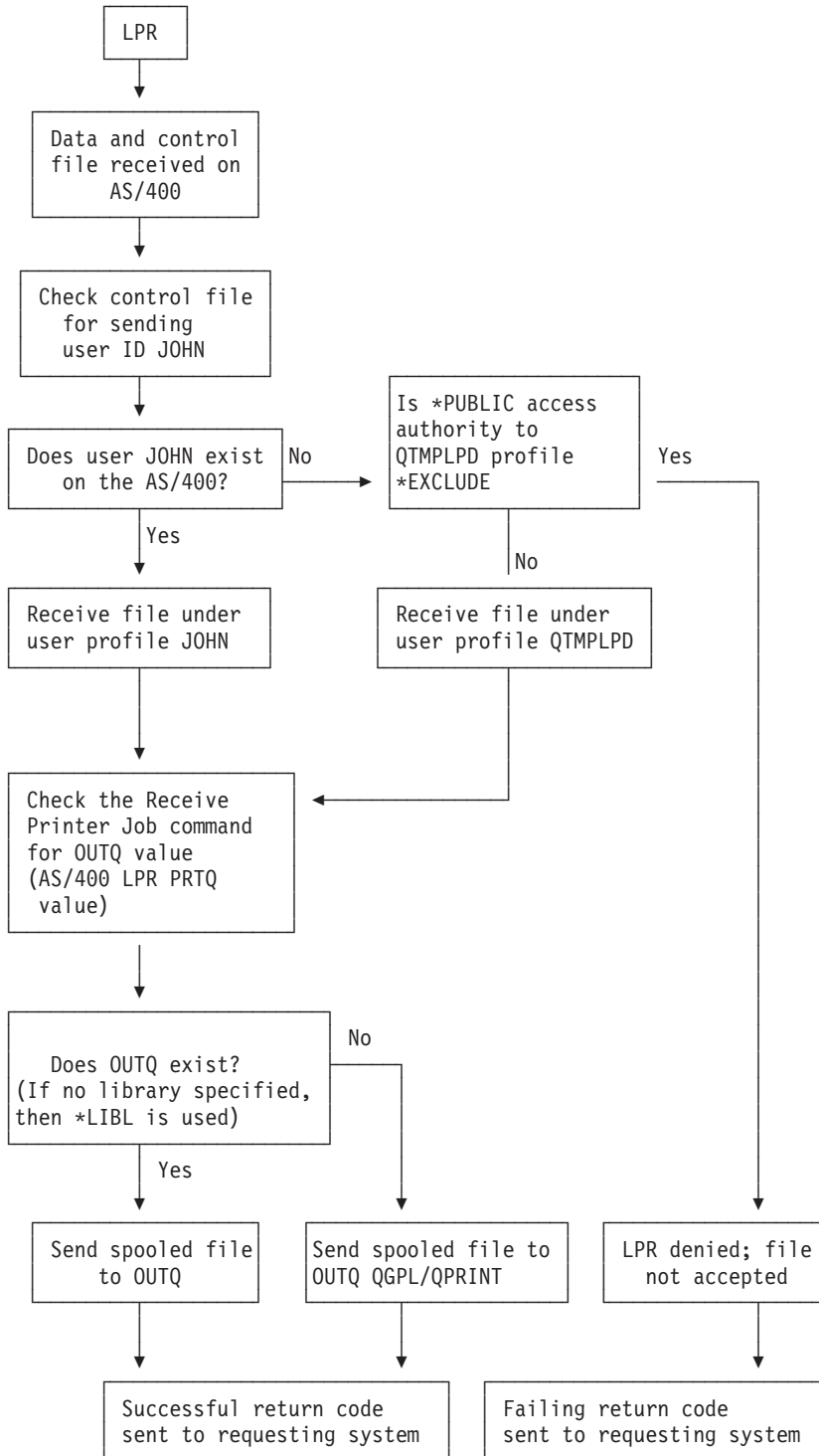


Figure 228. Flow for Determining Output Queue for a Spooled File

- If the user ID on the sending system exists on the destination system, an attempt is made to create the spooled file on the output queue that is defined on the *Receive a Printer Job* command under that user ID.
 - If the requested output queue exists, the spooled file is placed in that output queue. The user ID must have access to that output queue, or the OUTQ

- must have the OPRCTL parameter set to *YES. This means that anyone with *JOBCTL authority, like LPD, can access the OUTQ.
- If the output queue name does not contain a library name, the *LIBL library is searched.
 - If the specified output queue cannot be found, the spooled file is sent to the output queue QPRINT in the library QGPL.
 - If the specified user does not exist on the destination AS/400 system, the *PUBLIC access authority for the user QTMPLPD is checked. The *LIBL library associated with the profile QTMPLPD is searched for the requested output queue.
 - If the authority is not *EXCLUDE, the *Receive Printer Job* command is checked for the name of the requested output queue.
 - If the output queue is found, the file is sent to that output queue, provided the user profile QTMPLPD has access to that output queue or the OUTQ has the OPRCTL parameter set to *YES. This means that anyone with *JOBCTL authority, like LPD, can access the OUTQ.
 - If the specified output queue cannot be found, the spooled file is sent to the output queue QPRINT in the library QGPL.

In all of the previous situations, the file is considered successfully sent and received.

If the QTMPLPD profile has public access set to *EXCLUDE, access to the output queue is denied, and the file is rejected by the AS/400 destination system. The error message is sent to the message queue for the QTMPLPD default user profile.

Important!:

There is no process to notify the requesting system that an authority error was detected because the TCP connection is usually closed by the time this is determined. Any success messages posted by the LPR client application mean the file was temporarily received but not necessarily kept. Do not delete any files until you have verified that you have proper authority and that your files were received successfully on the destination system.

How Authority for Putting Spooled Files on Output Queue is Determined

When LPD creates spooled files, it checks to ensure that the requester has the proper authority to place spooled files on the output queue. It checks to ensure that the following are true:

1. The user has *READ authority to the output queue.
2. The user has *SPLCTL special authority.
3. The user has *JOBCTL special authority.
4. The output queue is OPRCTL(*YES).

If any of these conditions are true, then the user has authority to place the file on the requested output queue. If none of these conditions are true, the file goes to the default output queue QPRINT in library QGPL.

Using LPD to Print ASCII Files

A possible use of LPD is to print ASCII files received from systems running LPR-to-ASCII workstation printers that are attached to the AS/400 in one of the following ways:

- ASCII workstation controllers.
- NWS (Nonprogrammable workstation) printer ports.
- PWS (Programmable Workstation) serial ports.

This is useful for situations where a programmable workstation does not have its own printer attached. You can also use it from any ASCII system that is running LPR.

OS/2 TCP/IP—Example

The following is the syntax of the LPR command on OS/2 TCP/IP that prints file CONFIG.SYS directly to printer WSPRT9 on AS/400 RCHASM03:

```
lpr config.sys -p WSPRT9 -s RCHASM03
```

Additionally, an LPD server must be started on RCHASM03, and WSPRT9 must be a printer of the type described previously. The file arrives with a sending user ID of PC-USER and is processed according to Figure 228 on page 371.

Note: Because the spooled file is still an ASCII file, you cannot display the file on the AS/400 system while it is on the print queue.

Using LPD to Print ASCII Files Converted to EBCDIC

Another use of LPD is to receive printer files from ASCII hosts running LPR and print them in EBCDIC. There are some limitations on character conversion, however. The ASCII hosts must be able to convert the printer data stream to EBCDIC before sending it using LPR.

To do this, you need to customize the LPD printer file on the AS/400. Full details are given in the commentary to the code in the example shown in Figure 229 on page 375.

Do not change the original printer file in QTCP. Instead, make a copy of it, change the copy, and include it higher up the user library list portion of the *LIBL for the user profile receiving the file.

There are two options for changing the LPD printer file:

1. Change the QUSRSYS/QPTMPLPD working copy from *USERASCII to *SCS. This affects all LPR clients because it resides in the *SYSLIBL path and is found first by all user profiles. This means that all LPR clients must send EBCDIC data streams (or at least only send EBCDIC streams until the QUSRSYS/QPTMPLPD printer file is restored to *USERASCII type).
2. Change the QUSRSYS/QPTMPLPD working copy from *USERASCII to *SCS, and move it to another library that is your *CURLIB or somewhere in your *USRLIBL path. Ensure that your target library is found ahead of library QTCP, which has the installation copy of the printer file. You must also ensure that the target library is unique to your user profile, *LIBL, in order to avoid affecting other users.

Your *LIBL is considered to be one that exists when your user profile is signed on. You cannot run any programs when you sign on to set the library list. Change to your current interactive session has no effect upon the *LIBL used. This is because the user profile *LIBL is checking occurs in an LPD server batch job and not your interactive job.

To receive EBCDIC files, the LPR client must be any user with an AS/400 user profile that has the *SCS working copy in its *LIBL path. For example, if someone moves the printer file to library JOHNDOE, any user profile with JOHNDOE as its *CURLIB or in its *USRLIBL uses the *SCS printer file.

Attention

Removing the working copy from library QUSRSYS forces every user profile that does not have library JOHNDOE in its *LIBL to find the installation copy in QTCP. If library QTCP is not already in the *LIBL, then the system adds it as the last library in the *LIBL for all user profiles. This insures that the system finds a printer file. Do not change the installation copy in QTCP. This insures that other user profiles, including the default profile QTMPLPD, still receive ASCII data streams as usual.

System/370 Example

For System/370 systems, source files already exist in EBCDIC form. Therefore, when you send System/370 files to an AS/400 using LPR, specify the *binary* option to keep the system from translating the files from EBCDIC into ASCII.

The following example shows the LPR command to send to an AS/400 that has DEVTYPE(*SCS) for the QTMPLPD printer file:

```
LPR fname ftype fmode (HOST AS400.ENDICOTT.IBM.COM
  PRINTER some1ib/someoutq COPIES 1 NOHEADER
  WIDTH 132 FILTER f LINECOUNT 66 BINARY
```

This example explicitly specifies the width of the source file. It is best if the source file uses fixed-width records instead of variable-length records to ensure that it pads all records to this width.

The BINARY flag indicates that the LPR platform must not do any EBCDIC to ASCII conversion. This means that the LPD server receives an EBCDIC data stream into an *SCS spooled file. However, this spooled file prints only on EBCDIC printers.

RISC System/6000 Example

This example uses the AIX C shell. It has not been tested on other UNIX systems.

In this example, you automatically pipe any program arguments passed into the **dd** command and then pipe the output of the **dd** command to the **lpr** program.

In this example, as400 is the name of the AS/400 printer to which you are sending the file.

```

#!/bin/csh -f
#-
# Convert ASCII file to EBCDIC and send it to AS/400
# to be received as *SCS file
# the AS/400, it is required that the "working" LPD
# PRTF located in QUSRSYS be changed with the following
# command:
#
#   CHGPRTF FILE(QUSRSYS/QPTMPLPD) DEVTYPE(*SCS)
#
# When you are finished, restore the original settings
# with:
#
#   CHGPRTF FILE(QUSRSYS/QPTMPLPD) DEVTYPE(*USERASCII)
#
# Caveats:
#
# - Square brackets will not convert properly. Other
#   special characters may not convert properly.
#
# - If you customize the QUSRSYS/QPTMPLPD, you will
#   affect all users of LPD services who may not want
#   *SCS files.
#
# - It is strongly recommended that QTCP/QPTMPLPD be
#   left alone. This is the working installation
#   default version. Copy or customize it to another
#   library that will be found ahead of QTCP.
#
# - If you erase the copy of QPTMPLPD in QUSRSYS, your
#   *LIBL is searched, and if no QPTMPLPD is found, the
#   version in QTCP is used. Therefore, you may copy
#   the QPTMPLPD printer file to a private library in
#   your *USRLIBL or *CURLIB and change it to be *SCS
#   without affecting other users, provided your copy
#   is found ahead of any other versions (namely, the
#   one in QTCP).
#-----
set nm=$0
if ("$1" == "-h" || "$1" == "") then
    echo " "
    echo "Usage:" "$nm:t" "[-h] file(s)"
    echo " "
    echo "Will convert file to EBCDIC and LPR to AS/400 printer queue"
    echo "using the following string:"
    echo " "
    echo "   dd conv=ebcdic cbs=132 < $* | lpr -P as400 -1"
    echo " "
    exit
endif

echo "dd conv=ebcdic cbs=132 < $* | lpr -P as400 -1"
dd conv=ebcdic cbs=132 < $* | lpr -P as400 -1

exit;

```

Figure 229. Sample AIX C Shell for Printing AIX File on AS/400

Authority Required to Receive Spooled Files

The destination system administrator restricts output queue access for users without user IDs on the destination system. This is done by restricting the access authorities of the QTMPLPD user ID, which is the default profile used for any user ID that is not found. However, this restriction does not affect user IDs that are found on the destination system.

If you set the *PUBLIC authority of the user profile QTMPLPD to *EXCLUDE, only users with user IDs that are the same on both the sending and receiving AS/400 systems receive spooled files on the destination system. The QTMPLPD user profile ships with an authority of *OBJOPR. If you do not have a user ID on the destination AS/400 system, you are still able to send spooled files to the destination system under the QTMPLPD user profile.

Receiving Spooled Files — Benefits

There are advantages to having a user ID defined on the receiving AS/400 system. If you have the same user ID on both systems, you are the owner of the spooled file on the receiving system. It is then easier to find and access the spooled file on the destination system. You can find the spooled file with the *Work with Spooled Files* (WRKSPLF) command.

If you do not have a user ID on the receiving system, then QTMPLPD owns the file. Because you do not own the spooled file, you might have limited access to the spooled file. Your authority to the output queue on which the spooled file is located determines your access to the spooled file.

If you are using a security level of 50, the system value QALWUSRDMN (allow user domain objects in libraries) must contain the library name QTEMP to enable LPR service. It must also contain the library name QUSRSYS to enable LPD service.

Creating a Default User Profile

A user profile, QTMPLPD, is necessary to allow remote LPR requests that do not have an AS/400 user ID to access printer files on an AS/400 system running LPD.

The QTMPLPD user profile ships with the TCP/IP licensed program. If you damage or delete the QTMPLPD user profile, specify the following to create the user profile:

```
CRTUSRPRF USRPRF(QTMPLPD)
          PASSWORD(*NONE)
          USRCLS(*USER)
          MSGQ(QTCP/QTMPLPD)
          AUT(*EXCLUDE)
          PWDEXP(*NO)
          PWDEXPITV(*NOMAX)
```

```
RVKOBJAUT OBJ(QTMPLPD)
          OBJTYPE(*USRPRF)
          USER(*PUBLIC)
          AUT(*ALL)
```

```
GRTOBJAUT OBJ(QTMPLPD)
          OBJTYPE(*USRPRF)
          USER(*PUBLIC)
          AUT(*OBJOPR)
```

Chapter 13. BOOTP Server

Bootstrap Protocol, or BOOTP, provides a dynamic method for associating workstations with servers. It also provides a dynamic method for assigning workstation IP addresses and initial program load (IPL) sources. Together, BOOTP and Trivial File Transfer Protocol, or TFTP, provide support for the IBM Network Station for the AS/400 system. They also provide support for other clients that use the BOOTP and TFTP protocols. For more information on TFTP, see “Chapter 14. TFTP Server” on page 383.

BOOTP is a TCP/IP protocol. It allows a client to find its IP address and the name of a load file from a server on the network. A client uses BOOTP to find this information without intervention from the user of the client.

The BOOTP server listens on the well-known BOOTP server port 67, which DHCP also uses. Because of this, BOOTP and DHCP cannot operate at the same time on the same system. When the server receives a client request, it looks up the IP address that is defined for the client and returns a reply to that client. This reply contains both the client’s IP address and the name of the load file. The client then initiates a TFTP request to the server for the load file.

The mapping between the client hardware address and IP address is kept in the BOOTP table, which the AS/400 system administrator maintains. For information about working with the BOOTP table, see “Working with the BOOTP Table” on page 379.

Accessing BOOTP Functions through Operations Navigator

You can access BOOTP server functions from a command line interface or from Operations Navigator. Not all BOOTP functions are available on both interfaces.

This chapter discusses how to start, stop, and configure the BOOTP server from the command line interface. The only Operations Navigator function that this chapter documents is how to add Network Stations to existing BOOTP environments. For more information, see “Adding IBM Network Stations to an Existing BOOTP Environment” on page 379.

To access BOOTP server functions through Operations Navigator, perform the following steps:

1. Double-click your AS/400 server in the main tree view of Operations Navigator.
2. Double-click **Network**.
3. Double-click **Servers**.
4. Double-click **TCP/IP**.
5. Right-click **BOOTP** to open a context menu of BOOTP server functions.

Note: Although some of the functionality of the command line interface and the Operations Navigator is the same, the actual menu commands and parameters are not necessarily the same.

Starting the BOOTP Server

Specify the following *Start TCP/IP Server* (STRTCPSVR) command with the SERVER parameter set to *BOOTP:

```
STRTCPSVR SERVER(*BOOTP)
```

Automatically Starting the BOOTP Server

Set the AUTOSTART parameter to *YES on the CHGBPA command. It has no effect on the STRTCPSVR command because the STRTCPSVR command ignores the AUTOSTART parameter value. If you run the STRTCPSVR SERVER (*BOOTP) command while the BOOTP server is running, you receive a diagnostic message.

Ending the BOOTP Server

Specify the following *End TCP/IP Server* (ENDTCPSVR) command with the SERVER parameter set to *BOOTP:

```
ENDTCPSVR SERVER(*BOOTP)
```

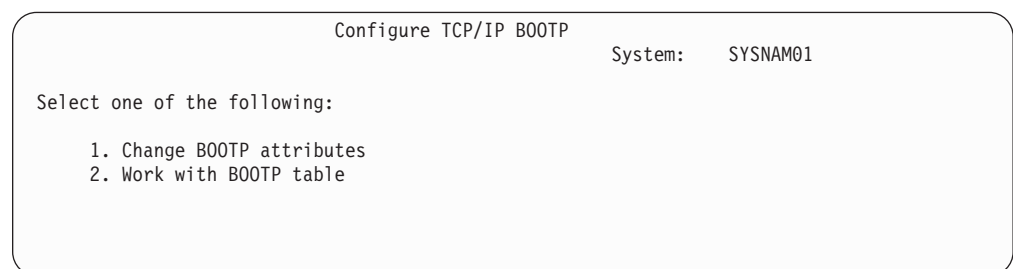
Configuring the BOOTP Server

Use the *Configure TCP/IP BOOTP* (CFGTCPBP) command to configure the BOOTP server. The following are two different ways to get to this command prompt:

- Specify the *Configure TCP/IP BOOTP* (CFGTCPBP) command.
- Specify the *Configure TCP/IP Applications* (CFGTCPAPP) command from the command line and select option **4**, *Configure BOOTP*.

After you specify the command, the following display appears:

Figure 230. *Configure TCP/IP BOOTP*



The following two AS/400 commands control the BOOTP server:

- The *Change BOOTP Attributes* (CHGBPA) command allows an administrator to set the configurable attributes for the BOOTP server.
- The *Work with BOOTP Table* (WRKBPTBL) command allows an administrator to work with the BOOTP table.

Changing BOOTP Attributes

Select option **1**, *Change BOOTP Attributes*, of the *Configure TCP/IP BOOTP* display (or simply type CHGBPA and press F4) to display the *Change BOOTP Attributes* display. The AUTOSTART parameter controls if the BOOTP server is to start automatically when TCP/IP is started with the STRTCP command.

Note: You must have *IOSYSCFG special authority to make changes to the BOOTP attributes with the CHGBPA command.

```
Change BOOTP Attributes (CHGBPA)

Type choices, press Enter.

Autostart server . . . . . *YES          *YES, *NO, *SAME
```

Figure 231. Change BOOTP Attributes (CHGBPA)

Working with the BOOTP Table

Select option **2**, *Work with BOOTP Table*, of the *Configure TCP/IP BOOTP* display (or simply specify WRKBPTBL) to display the *Work with BOOTP Table* display.

The administrator uses the *Work with BOOTP Table* display to add, change, remove, or display an entry in the BOOTP table.

For information about working with the BOOTP table, see *IBM Network Station Manager for AS/400*, SC41-0632.

```
Work with BOOTP Table                               System:  SYSNAM01

Type options, press Enter.
 1=Add  2=Change  4=Remove  5=Display

Client
Host    MAC      IP
Opt  Name      Address  Address
-----
act01.ibm.com  02.01.8C.06.34.98  9.130.42.1
```

Figure 232. Work with BOOTP Table (WRKBPTBL)

Adding IBM Network Stations to an Existing BOOTP Environment

This section describes how to add Network Stations to an existing BOOTP environment. You can use either the command line interface or Operations Navigator to add Network Stations.

Note: You might have BOOTP clients that are on a subnet that is different from the one on which the Bootstrap server is located. If this is the case, you need to have a router that shows the server to those clients. For more information, see *Network Station Manager Installation and Use*, SC41-0664-00.

Adding Network Stations with the Command Line Interface

This procedure describes how to use the command line interface to add Network Stations to an existing BOOTP environment.

1. At an AS/400 command prompt, specify the following:
WRKBPTBL
2. In the *Options* field, specify **1** to add a Network Station.
3. Specify the following information:
 - Client host name – The host name identifies the Network Station as a unique destination within a TCP/IP environment. An example of a valid host name is ns1.mycompany.com.
 - MAC address – The Media Access Control (MAC) address is a unique, hardware-specific identifier for each Network Station. The address is located on the box of the Network Station. To find the MAC address without the box, perform the following steps:
 - a. Power on the Network Station.
 - b. After the keyboard controller test, press Enter.
 - c. In the Setup Utility, press **F4**.
 - d. Record the MAC address.
 - IP address – Each Network Station requires a unique IP address. Therefore, you must assign a specific address to each Network Station. You must ensure that the IP address is valid for your organization and that no other device in the network uses it. An example of a valid IP address is 192.168.1.2.
4. Press Enter to exit the *Configure TCP/IP BOOTP* display.

Adding Network Stations with Operations Navigator

This procedure describes how to use Operations Navigator to add Network Stations to an existing BOOTP environment. Operations Navigator requires OS/400 V4R2 or later.

1. Use Operations Navigator to locate the BOOTP server with the following path:
Network object/Servers/OS/400
2. Double-click **BOOTP**.
3. Click **Add**.
4. Specify the following *Network Device* information:
 - Client host name – The host name identifies the Network Station as a unique destination within a TCP/IP environment. An example of a valid host name is ns1.mycompany.com.
 - MAC address – The MAC address is a unique, hardware-specific identifier for each Network Station. The address is located on the box of the Network Station. To find the MAC address without the box, perform the following steps:
 - a. Power on the Network Station.
 - b. After the keyboard controller test, press Enter.

- c. In the Setup Utility, press **F4**.
- d. Record the MAC address.
- IP address – Each Network Station requires a unique IP address. Therefore, you must assign a specific address to each Network Station. You must ensure that the IP address is valid for your organization and that no other device in the network uses it. An example of a valid IP address is 192.168.1.2.
- Hardware type – Your Network Stations can attach to either a Token-ring or Ethernet LAN.
 - Specify a value of **6** for Token-ring or IEEE (802.3) Ethernet networks.
 - Specify a value of **1** for a Version 2 (802.2) Ethernet network.
5. If you do not use Gateway IP addresses for remote LANs, leave this field blank. Otherwise, specify the IP address of the IP router or gateway that your Network Station uses to reach the server.
6. If you do not use a subnet mask for remote LANs, leave this field blank.
7. Verify that the following default values are correct:
 - *Type* is IBM Network Station Manager.
 - *File name and directory* is /QIBM/ProdData/NetworkStation/kernel.
8. Click **OK**.
9. Repeat steps 3 through 8 for each additional Network Station.
10. Click **OK** to update the BOOTP server.

Chapter 14. TFTP Server

Trivial File Transfer Protocol, or TFTP, is a simple protocol that provides basic file transfer function with no user authentication. Together, TFTP and Bootstrap Protocol, or BOOTP, provide support for the IBM Network Station for an AS/400 system. They also provide support for other clients that use the TFTP and BOOTP protocols. For more information on BOOTP, see “Chapter 13. BOOTP Server” on page 377.

TFTP is a generic implementation that allows you to use clients other than IBM Network Stations. For more information, see “Configuring TFTP for Clients other than IBM Network Station” on page 389.

Accessing TFTP Functions through Operations Navigator

You can access TFTP server functions from either a command line interface or Operations Navigator. Not all TFTP functions are available on both interfaces.

This chapter discusses how you start, stop, and configure the TFTP server from the command line interface. This chapter does not document any of the Operations Navigator functions. See the online Help for Operations Navigator for information about using the Operations Navigator for TFTP functions.

To access TFTP server functions through Operations Navigator, perform the following steps:

1. Double-click your AS/400 server in the main tree view of Operations Navigator.
2. Double-click **Network**.
3. Double-click **Servers**.
4. Double-click **TCP/IP**.
5. Right-click **TFTP** to open a context menu of TFTP server functions.

Note: Although some of the functionality of the command line interface and the Operations Navigator is the same, the actual menu commands and parameters are not necessarily the same.

Starting the TFTP Server

Specify the following *Start TCP/IP Server* (STRTCPSVR) command with the SERVER parameter set to *TFTP:

```
STRTCPSVR SERVER(*TFTP)
```

The STRTCPSVR command ignores the AUTOSTART parameter value. If you run the STRTCPSVR SERVER (*TFTP) command when the TFTP server is running, you receive a diagnostic message.

Automatically Starting the TFTP Server

Set the AUTOSTART parameter to *YES on the CHGTFTPA command. The AUTOSTART parameter of the CHGTFTPA command affects only the operation of the STRTCP command. It has no effect on the STRTCPSVR command.

Ending the TFTP Server

Specify the following *End TCP/IP Server* (ENDTCPSVR) command with the Server parameter set to *TFTP:

```
ENDTCPSVR SERVER(*TFTP)
```

Changing TFTP Attributes

The *Change TCP/IP TFTP Attributes* (CHGTFTPA) command is used to change the TFTP server attributes. The following are two different ways to get to this command prompt:

- Specify the *Change TCP/IP TFTP Attributes* (CHGTFTPA) command.
- Select option **3** on the *Configure TCP/IP Applications* (CFGTCAPP) display.

Note: You must have *IOSYSCFG special authority to make changes to the TFTP attributes with the CHGTFTPA command.

```
Change TFTP Attributes (CHGTFTPA)

Type choices, press Enter.

Autostart server . . . . . *NO          *YES, *NO, *SAME
Enable subnet broadcast . . . . *YES      *YES, *NO, *SAME
Number of server jobs:
  Minimum . . . . . 2          1-20, *SAME, *DFT
  Maximum . . . . . 6          1-250, *SAME, *DFT
Server inactivity timer . . . . 30        1-1440, *SAME, *DFT
ASCII single byte CCSID:
  Coded character set identifier 00819      1-65532, *SAME, *DFT
Maximum block size . . . . . 1024        512-65464, *SAME, *DFT
Connection response timeout . . 60        1-600, *SAME, *DFT
Allow file writes . . . . . *NONE        *DFT, *NONE, *CREATE...
Alternate source directory . . . '*NONE'

More...
F3=Exit  F4=Prompt  F5=Refresh  F12=Cancel  F13=How to use this display
F24=More keys
```

Figure 233. Change TFTP Attributes (CHGTFTPA) – Display 1

```
Change TFTP Attributes (CHGTFTPA)

Type choices, press Enter.

Alternate target directory . . . '*NONE'
```

Figure 234. Change TFTP Attributes (CHGTFTPA) – Display 2

Server and Client Ports

The TFTP server uses a subnet-directed broadcast address as the destination address. It also uses a well-known port as the port of datagrams sent to clients that have requested the subnet broadcast option. The clients listen for and receive datagrams on the well-known port. The keyword for the well-known port is **subntbcst_ftp**, and its decimal value is **247**.

The TFTP server sends subnet-directed broadcast datagrams to clients that request the subnet broadcast option. The source ports from which the TFTP server sends these datagrams do not have to be unique. They can be arbitrarily allocated.

Some routers filter or block subnet-directed broadcast datagrams. In support of router filters, you can define restricted ports for the QTFTP profile. If you define restricted ports for the QTFTP profile, the TFTP server uses only the defined restricted ports as the source ports for the subnet-directed broadcast datagrams. Network administrators define router filtering rules to allow subnet-directed broadcast datagrams to pass through router filters based on the source port of subnet-directed datagrams being one of the restricted ports defined for the QTFTP profile.

TFTP Extensions

The following sections include information on the TFTP Transfer Size Option and the Subnet Broadcast TFTP.

TFTP Transfer Size Option

The Transfer Size option allows the client to determine how much data is transferred on a read request (RRQ). This is useful for requesting a subnet broadcast of a file. The client finds the size of the buffer it needs in order to store the file in memory. Drawing from this block size, the client determines the number of blocks for the transfer. The number of blocks is helpful information for tracking the blocks that have been received. You can also use it for the last block acknowledgment (ACK), which must be sent to terminate a transfer normally. Without the Transfer Size option, determining the size and the last block of the transfer requires the client to wait for a block to be received that is smaller than the block size of the transfer.

Note: For files transferred in *netascii* mode, this option might not be as useful if you are converting the data during the transfer in a way that changes its size. Also, the server might require additional processing time to determine the transfer size due to conversion of the file to the appropriate CCSID.

TFTP Subnet Broadcast Option

With the increasing popularity of the Network Station, the possibility for *boot storms* also increases. These storms occur when large numbers of clients request their boot code at the same time. When hundreds of stations are involved in booting, the same data must be routed through each hop in the network between each Network Station and the server.

The TFTP Subnet Broadcast option provides a solution to this problem. It allows the server to broadcast the boot code to the Network Stations on a subnet basis. Using

subnet-directed broadcast, Subnet Broadcast data packets are unicast between routers until they reach the subnet on which the Network Stations reside. At this point, the router at the destination subnet broadcasts the data packets to the Network Stations on the subnet. Disinterested hosts on the subnet throw the data packets away. The packets are usually thrown away by the host's IP layer after it determines that no applications are interested in receiving data on the port to which the broadcast was directed. See Figure 235 on page 387 for an illustration of a subnet-directed broadcast. This solution can drastically reduce the network traffic as well as the time that it takes many Network Stations to boot when booting simultaneously.

The TFTP Subnet Broadcast option enables clients to join a broadcasting filegroup. It also allows clients to receive all subsequent blocks for a file until the client becomes the master client. A client becomes the master client when it receives an Option Acknowledge (OACK) packet from the TFTP server that indicates that it is the master client. A client must keep track of blocks that it receives. After a client becomes the master client, it can request the blocks that it has not received. The master client requests blocks by sending ACK packets that include the block number of the block *prior to* the block that the master client requires. For example, if the client wants block 5, it sends an ACK with a block number of 4.

When a client receives an OACK packet that indicates that it is the master client, the client must send an ACK that requests the first block it requires. From then on, the client must request blocks in ascending but not necessarily consecutive order. A master client continues to send ACK packets to the server to indicate the next block that it requires. When the master client receives all of the blocks it requires, it sends an ACK with the number of the last block on the file being transferred. Once the server receives an ACK with the last block number of the file being transferred, the transfer to the client sending the ACK is considered complete. A client can terminate its transfer at any time by sending an ACK for the last block or by sending an Error (ERR) packet. A client can terminate this transfer regardless of whether it is the master client or not.

Note: This TFTP Subnet Broadcast option is designed to improve simultaneous transfer of large files to multiple clients on a common subnet. This option does not help with files that require only a few blocks to transfer or single client transfers.

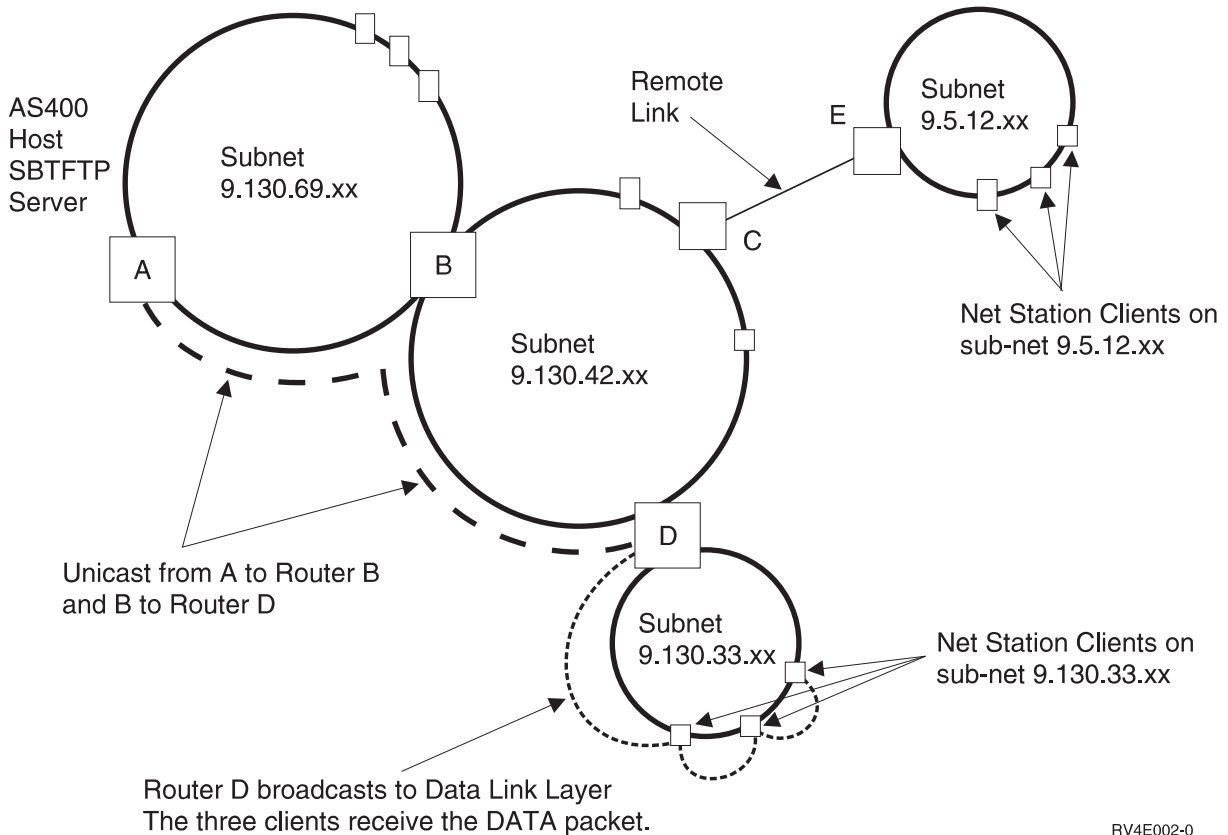


Figure 235. Example of Broadcasting over Subnets

Client to Server TFTP Read Request (RRQ) Options

The information that follows includes the additional TFTP options that are supported and a description of their use. To view the standard TFTP request parameters and their meanings, refer to Internet Request for Comments (RFC) 1350. For more information related to the TFTP options that are described here, see RFCs 1782, 1783, and 1784. Internet RFC 2090 describes the TFTP multicast option, which has some similarities to the Subnet Broadcast option. However, the TFTP Multicast option it is not supported at this time. The TFTP Multicast option RFC is mentioned here as a reference to help understand the Subnet Broadcast option.

The following is a list of supported options and their descriptions:

blksize

Null (0h) terminated keyword *blksize* that is followed by the requested block size and represented as a null-terminated ASCII string. This option requests a block size for the requested file transfer instead of using the default of 512.

sbroadcast

Null-terminated keyword *sbroadcast* that is followed by the subnet mask of the subnet to which the client is connected. This option indicates that the client wants to participate in a subnet-directed broadcast group. The subnet mask that is included with this option is used with the client's IP address to determine the client's subnet address.

tsize

Null-terminated keyword *tsize* that is followed by a null-terminated ASCII representation of 0 (30h). This option is a request for the server to return the file size in an Option Acknowledgment (OACK).

Server to Client TFTP Option Acknowledgment (OACK)

The TFTP server sends an Option Acknowledgment (OACK) to a client in response to either a read request or a write request that includes additional TFTP options as described in “Client to Server TFTP Read Request (RRQ) Options” on page 387. An OACK that the servers sends in response to a transfer request includes only responses to requested options that the server supports. The server can also send an OACK to a client subsequently to the start of a subnet broadcast transfer. This is done to indicate to the client whether it is the master client in a subnet broadcast file group. An OACK packet that the server sends subsequently to the start of a subnet broadcast transfer includes the *sbroadcast* option.

The following is a list of supported options and their descriptions:

blksize

Null (0h) terminated *blksize* keyword that is followed by the block size that is used for this file transfer. It is represented as a null-terminated ASCII string. This is the response to a requested block size, and the value returned here can be less than the requested block size. The server determines the block size for the transfer based on the requested block size, the maximum configured block size, and possibly the subnet broadcast transfers that are already in progress.

sbroadcast

Null-terminated *sbroadcast* keyword that is followed by a null-terminated ASCII string that includes the following fields separated by commas:

port

The ASCII representation of the port to which the subnet-directed broadcast datagrams are broadcast. This is the well-known port that is registered with the Internet Assigned Number Authority (IANA) with the keyword of *subntbcst_tftp* and a decimal value of 247. This field might be empty in OACK packets that the server sends subsequently to the start of a subnet broadcast transfer.

sbid

The ASCII representation of a decimal number that is called the *subnet broadcast identifier*. Possible values are 0 through 4,294,967,295 (FFFFFFFFh). This is used along with the server source port to determine if a subnet-directed broadcast datagram is part of a requested transfer. This field can be empty in OACK packets that the server sends subsequently to the start of a subnet-based broadcast transfer.

mc

This is either an ASCII (31h) **1** or ASCII **0** (32h) to indicate to the client whether it is currently the master client. A value of **1** indicates that the client is the master client, and a value of **0** indicates that the client is not the master client.

In response to an OACK, the master client must send an ACK to the server. The master client sets the block number in this ACK to the number of the block prior to the first block that is required by the master client.

The master client acknowledges subnet broadcast data (BDATA) packets by sending an ACK to the server. The master client sets the block number in this ACK to the block prior to the current block that the master client requires.

Clients that are not indicated as being the master client respond to an OACK packet with an ACK that has the block number set to zero.

Note: The block number in ACK packets is the 2-byte binary representation of the number in network byte order.

tsize

The null-terminated *tsize* keyword that is followed by the null-terminated ASCII representation of the decimal number that represents the file size of the requested file. The client uses this information to ensure that it has enough space to store the file and to determine the last block number of the file.

Note: The client can also determine the file size and last block of a transfer when it receives a block that contains less data than the block size.

Server to Client Broadcast Data (BDATA) Packets

The following is a list of the fields in a Broadcast Data Packet and their descriptions:

block#

2-byte binary number in the network byte order that indicates the number of a particular block of data.

sbid

4-byte binary number in the network byte order that is called the *subnet broadcast identification*. This must be compared with the *sbid* that was returned in the OACK response to a read request (RRQ) with the Subnet Broadcast option. Along with the source port, this uniquely identifies a Subnet Broadcast File Transfer. The source port of the BDATA packet must be compared with the source port of the initial OACK packet that was received for this transfer. Only BDATA packets that match on both the SBID and source ports are considered part of the requested transfer. All other BDATA packets must be ignored.

data

This is the data for this block of the file transfer. With the exception of the last block of the file, the size of the data is equal to the block size for the transfer. The last block of the file must be less than the block size, even if it means that the length of the data in the last block is zero. However, the server might not be done broadcasting blocks after the last block of the file is broadcast. Control can be transferred to another client in the subnet broadcast file group that has not yet received all the blocks in the file.

Configuring TFTP for Clients other than IBM Network Station

To allow other clients to use the TFTP server, you must ensure that the QTFTP profile has authority to access the directories and files that the clients access through the TFTP server. You also need to set the TFTP server attributes to allow the desired client requests.

When configuring TFTP for use by clients other than the IBM Network Station, first determine the directories and files that the clients are using. For this example, the clients use the TFTP server to read files from the directory `/netpc/bin/system`.

1. Use the MKDIR command with an argument of /netpc to create the directory /netpc, as follows:
MKDIR (netpc)
2. Specify the WRKLNK command with an argument of /netpc, as follows:
WRKLNK (netpc)
3. Specify option **9** to display the current authorities.
4. For the *PUBLIC user, specify option **2**, *Change user authority*, and specify *NONE for *New data authorities*. This ensures that the file is not open to the public.
5. To add a user on the *Work with Authority* menu, specify the following on the first line: 1 for *Opt*, QTFTP for *User*, and *RX for *Data Authority*. Press Enter.
6. Press the **PF5** key to refresh the menu. You see the userid *PUBLIC with a data authority of *EXCLUDE, the userid QTFTP with a data authority of *RX, and your own userid with a data authority of *RWX.

Use the MKDIR command to create the following directories:

```
/netpc/bin
/netpc/bin/system
```

Each directory inherits the authority of the parent directory and has the owner added implicitly as a user with *RWX authority. Copy any files that the client requests to the netpc/bin/system subdirectory. You can copy the files in various ways, such as using the COPY command, FTP, or Client Access/400. You must ensure that the QTFTP profile has *R authority to each file that the client requests. To set the authorities for the files, use the WRKLNK command and option **9**, *Work with Authority*.

7. Specify the CHGTFTP command and press the **PF4** key.
8. Change the Alternate source directory to /netpc/bin/system and press Enter. This allows the TFTP server to request any file with the appropriate authority settings, including the directory /netpc/bin/system in its path.
9. To have the changes take effect, stop the TFTP server with ENDTCPVSR *TFTP and restart it by using STRTCPSVR *TFTP.

Chapter 15. Routed Server

The Route Daemon, or RouteD, provides support for the Routing Information Protocol, or RIP, on an AS/400 system. RIP is the most widely used routing protocol today. It is an Interior Gateway Protocol, or IGP, that assists TCP/IP in the routing of IP data packets within an autonomous domain. Dynamic routing protocols allow you to handle networks with multiple routers or automatic switching to redundant routes.

RIP Version 2 is available for Version 4 Release 2 and above. If you have migrated from a previous release, the system automatically migrates your routing configuration files as part of the Version 4 Release 2 installation process. For more information, see "Working with Routed Configuration" on page 393 and "RIP_INTERFACE Statement" on page 394.

Accessing Routed Functions through Operations Navigator

You can access Routed server functions from a command line interface or from Operations Navigator. Not all of the Routed functions are available on both interfaces. However, most Routed functions are available *only* through Operations Navigator.

This chapter discusses how to start, stop, and configure the Routed server from the command line interface. It does not document any of the Operations Navigator functions. See the online Help in Operations Navigator for information about using Operations Navigator for Routed functions.

To access Routed server functions through Operations Navigator, perform the following steps:

1. Double-click your AS/400 server in the main tree view of Operations Navigator.
2. Double-click **Network**.
3. Double-click **Servers**.
4. Double-click **TCP/IP**.
5. Double-click **Routed**.

Note: Although some of the functionality of the command line interface and the Operations Navigator is the same, the actual menu commands and parameters are not necessarily the same.

The online help in Operations Navigator assists you with the following:

- Starting the Routed server.
- Automatically starting the Routed server whenever TCP/IP is started.
- Ending the Routed server.
- Configuring the Routed server.

Starting the Routed Server

To start the Routed server, specify the following *Start TCP/IP Server* (STRTCPSVR) command with the SERVER parameter set to *ROUTED:

```
STRTCPSVR SERVER(*ROUTED)
```

Automatically Starting the Routed Server

The AUTOSTART parameter of the CHGRTDA command affects the operation of the STRTCP command. It has no effect on the STRTCPSVR command, which ignores the AUTOSTART parameter value. If you run the STRTCPSVR SERVER (*ROUTED) command when the Routed server is running, you receive a diagnostic message.

You can use the STRTCPSVR command to restart the Routed server when the Routed server is *not running*. However, the system ignores the restart instruction and simply starts the server.

Ending the Routed Server

To end the Routed server, specify the following *End TCP/IP Server* (ENDTCPSVR) command with the server attribute set to *ROUTED:

```
ENDTCPSVR SERVER(*ROUTED)
```

Another way to end the Routed server (and all other servers) is to specify the ENDTCPSVR command without parameters. If you use the ENDTCPSVR command to end a server that is not active, you receive a diagnostic message.

Configuring the Routed Server

Use the *Configure TCP/IP RouteD* (CFGTCPRTD) command to configure the Routed server. The following are two different ways to get to this command prompt:

- Specify the *Configure TCP/IP RouteD* command, CFGTCPRTD, from the command line.
- Specify the *Configure TCP/IP Applications* command, CFGTCPAPP, from the command line. Select option **2** (Configure RouteD).

After you specify the command, you see the following display:

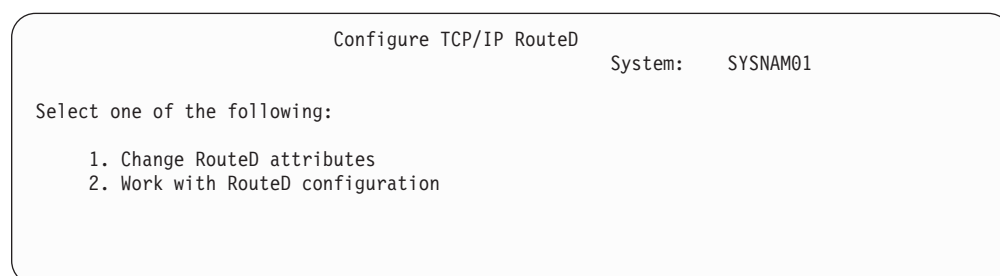


Figure 236. *Configure TCP/IP RouteD*

The following two AS/400 commands control the Routed server:

- The *Change RouteD Attributes* (CHGRTDA) command allows an administrator to set the configurable attributes for the Routed server.
- The *Work with RouteD Configuration* (WRKRTDCFG) command allows an administrator to work with the Routed configuration.

Working with RouteD Configuration

Use the *Work with RouteD Configuration* (WRKRTDCFG) command to change the RouteD server configuration. The following are two different ways to get to this command prompt:

- Specify the *Work with RouteD Configuration* command, WRKRTDCFG.
- Select option **2** on the *Configure TCP/IP RouteD* (CFGTCPRTD) display.

Note: You must have *IOSYSCFG special authority to make changes to the RouteD configuration with the WRKRTDCFG command.

```
Work with RouteD Configuration                               System:  SYSNAM01
Type options, press Enter.
  1=Add  2=Change  3=Copy  4=Remove  5=Display  13=Insert

Sequence
Opt  Number  Entry
-----
00010 # * * * * * >
00020 # RTD DEFAULT CONFIGURATION >
00030 # * * * * * >
00040 # >
00050 # RouteD Interface Definitions
00060 # -----
00070 # TCP/IP will learn about a route to network 9.0.0.0 th >
00080 # means external to RouteD, therefore do not allow Rout >
00090 # route to this network.
00100 #
00110 # RIP_INTERFACE * SUPPLY RIP1 METRIC 1 BLOCK 9.0.0.0 MA >
00120 #
00130 #
More...
F3=Exit  F5=Refresh  F6=Print List  F12=Cancel  F17=Top  F18=Bottom
```

Figure 237. Work with RouteD Configuration

RouteD Configuration Scenario

Figure 238 on page 394 shows how the RouteD configuration entries work in a sample network. The routers know every route within every network, including networks X, Y, Z, A, and W.

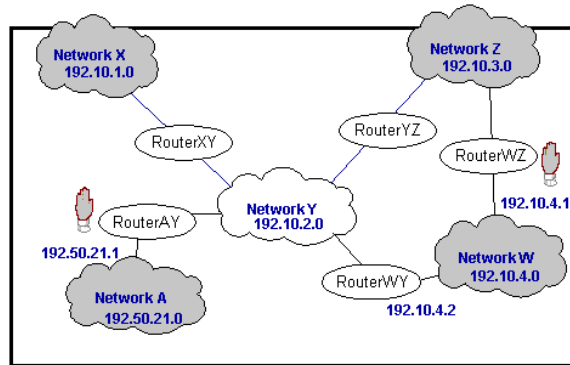


Figure 238. RouteD Configuration Scenario

- **Case 1** – If router AY has an interface of 192.10.2.1, a metric of 1, and a NOFORWARD parameter of 192.50.21.0, then none of the hosts in the networks reach network A.
- **Case 2** – If router WZ has an interface of 192.10.3.1, a metric of 1, and a NOFORWARD parameter of 192.10.4.0, then none of the IP packets go through router WZ to get to network W. IP packets can still reach network W, however, because router WY provides a route to that network.

Note: If you set the parameter option of any interface to Passive, then no routing takes place across the interface.

RIP_INTERFACE Statement

Use the RIP_INTERFACE statement to specify all of the routing options that you configure on a per-interface basis. The RIP_INTERFACE statement now contains the functionality for defining routes and creating static routes. Prior to Version 4 Release 2, this functionality existed in the NET statement and HOST statement.

You can specify multiple interface options on a single entry in the configuration file. You can accomplish this only if one of those options that requires a destination address appears on a given statement. Options include the following: block, forward, forward.cond, and noforward. For example, a statement can use the forward and metric options on a single line. However, the forward and noforward options cannot appear on the same line. It is recommended that only one option appear on a given line. It is also recommended that you use multiple lines to specify multiple options for a given interface.

You can specify interfaces on the AS/400 system in the following methods:

Network

Specified as an IP address and either a *mask* or an IP address and *bit number*. The bit number *n* indicates which bit in the 0 – *n* bits of the IP address (counting left to right) is the last bit of the IP address' network portion. If the MASK and bit number are missing, the system calculates a network by using the subnet mask of the interface specified through the ADDTCPIFC command.

Interface name

Logical Interface name that identifies a PPP interface with an IP address that is assigned dynamically when the PPP connection becomes active.

Hostname

The Host name of the AS/400 system, which is resolvable through DNS.

- * Refers to all of the interfaces on the AS/400 system and is useful for setting default values that apply to all interfaces. You can override these defaults by providing a RIP_INTERFACE statement for a specific interface with different values for selected parameters.

Supply Values

The following is a list of possible values for a RIP_INTERFACE supply value:

PASSIVE

The system does not receive or generate any RIP traffic on the specified interface.

SUPPLY RIP1

Indicates which version of the RIP protocol the system uses to send and receive routing information to and from neighboring routers. For SUPPLY RIP1, the system processes only RIPv1 packets.

SUPPLY RIP2

Indicates which version of the RIP protocol the system uses to send and receive routing information to and from neighboring routers. For SUPPLY RIP2, the system uses the multicast address 224.0.0.9 to process only RIPv2 packets, as specified in the RFC1723 sect.3.5.

SUPPLY OFF

Indicates that the system receives both RIPv1 and RIPv2 on the specified interface. However, the system does not send RIP packets.

Note: The default supply value for interfaces that you do not specify is SUPPLY RIP1. The system does not support RIP Version 1 Compatibility mode.

DIST_ROUTES_IN

DIST_ROUTES_IN controls how Routed redistributes routes that it receives from this RIP_INTERFACE network to Wide Area Networks, or WANs. This parameter does *not* affect redistribution of routes to Local Area Networks, or LANs. The following is a list of values and their definitions:

***CALC**

Routed determines a value of FULL or LIMITED by whether the RIP_INTERFACE network is a LAN or a WAN. If the specified interface is broadcast-capable, it is assumed local, and a value of FULL is given. Otherwise, the system uses a value of LIMITED.

FULL

Indicates that Routed redistributes routes that it receives from the specified interface to all of the other interfaces using normal RIP algorithm. Specify this value only for local networks.

LIMITED

Indicates that the server is not to redistribute routes that it receives from the RIP_INTERFACE network to other LIMITED interfaces. Specify this value only for some type of WAN. You cannot set this value for a LAN.

Metric

This parameter specifies the metric that the system adds to routes that it receives through the specified interface. Possible values are 1 through 15.

Community

This parameter specifies the community name used by this interface for authentication purposes as specified in RFC 1723 section 3.1. It is valid for interfaces with a SUPPLY value of RIP2. The *rip_community_name* is a character string of 1 to 16 characters in length.

If you specify the community option, then the system indicates that authentication is needed for this interface. The community name that is specified with the community option must match the community name sent in all RIP2 message blocks for this interface. If you do not specify the community option, then the system does not indicate any authentication for this interface.

Additional Parameters

You can also encounter the following RIP_INTERFACE parameters:

- BLOCK
- FORWARD
- FORWARD.COND
- NOFORWARD

BLOCK

The BLOCK parameter prevents the *network* route received on the specified interface from being included in the RouteD routes table. Consequently, the *network* is unknown and not forwarded to any other routers. Specify networks that you want to block by one of the following methods:

Network

A network that is specified as an IP address and a *mask* or as an IP address and a *bit number*. The bit number *n* indicates which bit in the 0 – *n* bits of the IP address (counting left to right) is the last bit of the network portion of the IP address. If the MASK and bit number are missing, the system calculates a network by using the subnet mask of the interface specified through the ADDTCPIFC CL command.

PRIVATE

The PRIVATE keyword refers to the sets of IP addresses that are designated for use by the Internet Assigned Number Authority (IANA) only within *private internets*. For more information, see RFC 1918, section 3.

- 10.0.0.0 to 10.255.255.255 (10/8 prefix) – 1 class A network.
- 172.16.0.0 to 172.31.255.255 (172.16/12 prefix) – 16 contiguous class B networks.
- 192.168.0.0 to 192.168.255.255 (192.168/16 prefix) – 256 contiguous class C networks.

When the RouteD server tries to send a route, it processes multiple forward parameters in the supplied order. The first forward parameter that allows the system to send the route over the specified interface ends the processing. The default is not to forward.

FORWARD

Use of the FORWARD keyword forwards the specified *network* route exclusively over the specified interface. If the specified interface is inactive, Routed takes no special action to forward this network.

Specify a *network* as both an IP address and a *mask* or as both an IP address and a *bit number*. The bit number *n* indicates which bit in the 0 – *n* bits of the IP address (counting left to right) is the last bit of the network portion of the IP address. If the MASK and bit number are missing, the system calculates a network by using the subnet mask of the interface specified through the ADDTCPIFC CL command.

FORWARD.COND

Use of the FORWARD.COND keyword forwards the specified *network* route exclusively over the specified interface. If the specified interface is inactive, Routed forwards the network over **all** of the other interfaces.

Specify a *network* as both an IP address and a *mask* or as both an IP address and a *bit number*. The bit number *n* indicates which bit in the 0 – *n* bits of the IP address (counting left to right) is the last bit of the network portion of the IP address. If the MASK and bit number are missing, the system calculates a network by using the subnet mask of the interface specified through the ADDTCPIFC CL command.

NOFORWARD

When you use the NOFORWARD parameter, the system does not send out RIP information about the specified *network* to the specified *interface*. Specify networks in one of the following two methods:

Network

Specify a *network* as both an IP address and a *mask* or as both an IP address and a *bit number*. The bit number *n* indicates which bit in the 0 – *n* bits of the IP address (counting left to right) is the last bit of the network portion of the IP address. If the MASK and bit number are missing, the system calculates a network by using the subnet mask of the interface specified through the ADDTCPIFC CL command.

PRIVATE

The PRIVATE keyword refers to the sets of IP addresses designated for use by the IANA within *private internets*. For more information, see RFC 1918, section 3.

- 10.0.0.0 to 10.255.255.255 (10/8 prefix) – 1 class A network.
- 172.16.0.0 to 172.31.255.255 (172.16/12 prefix) – 16 contiguous class B networks.
- 192.168.0.0 to 192.168.255.255 (192.168/16 prefix) – 256 contiguous class C networks.

Changing Routed Attributes

Use the *Change RouteD Attributes* (CHGRTDA) command to change the Routed server attributes. The following are two different ways to get to this command prompt:

- Specify the *Change RouteD Attributes* command, CHGRTDA.

- Select option **1** on the *Configure TCP/IP RouteD* (CFGTCPRTD) display.

Note: You must have *IOSYSCFG special authority to make changes to the RouteD attributes with the CHGRTDA command.

```
Change RouteD Attributes (CHGRTDA)

Type choices, press type.

Autostart . . . . . *No          *SAME, *YES, *NO
Supply . . . . . *No          *SAME, *YES, *NO
```

Figure 239. Change RouteD Attributes (CHGRTDA)

Chapter 16. REXEC Server

The Remote Execution (REXEC) server is a TCP/IP application that allows a client user to submit system commands to a remote server system. The user identifier, password, and command to be performed are sent from the user's client program to the server. The server validates the user, runs the requested command, and returns the results of the command to the client.

Commands submitted to the AS/400 host fall into three categories:

AS/400 command processor

You run AS/400 command processor commands by specifying QCAPCMD as the target of the client REXEC.

Qshell command interpreter (OS/400 option 30)

You can use the Qshell interpreter by specifying qsh as the target of client REXEC.

"Spawned paths"

You can run any AS/400 program in a "child" (spawned) job by specifying the complete path to the program or shell script as the target of the REXEC command.

Accessing REXEC Functions through Operations Navigator

Although, most REXEC functions are available *only* from a command line interface, you can also access configuration options from Operations Navigator.

To access REXEC server functions through Operations Navigator, perform these steps:

1. Double-click your AS/400 server in the main tree view of Operations Navigator.
2. Double-click **Network**.
3. Double-click **Servers**.
4. Double-click **TCP/IP**.
5. Double-click **REXEC**.

See the online Help in Operations Navigator for information about using Operations Navigator for REXEC functions.

The remainder of this chapter discusses how to start, stop, and configure the REXEC server from the command line interface.

Note: Although the command line interface and Operations Navigator provide much of the same function, the actual menu commands and parameters are not necessarily the same.

Starting the REXEC Server from the Command Line Interface

Specify the *Start TCP/IP Server* (STRTCPSVR) command with the SERVER parameter set to *REXEC, as follows:

```
STRTCPSVR SERVER(*REXEC)
```

Automatically Starting the REXEC Server

The AUTOSTART setting in the REXEC configuration affects the operation of the STRTCP command. It has no effect on the STRTCPSPVR command. The STRTCPSPVR command ignores the AUTOSTART parameter value. If you run the STRTCPSPVR SERVER (*REXEC) command when the REXEC server is running, the system starts one additional REXEC server.

Ending the REXEC Server

Specify the *End TCP/IP Server* (ENDTCPSPVR) command with the server attribute set to *REXEC, as follows:

```
ENDTCPSPVR SERVER(*REXEC)
```

Changing Attributes

The *Change REXEC Attributes* (CHGRXCA) command changes the REXEC server attributes. The following are two ways to get to this command prompt:

- Specify the *Change REXEC Attributes* (CHGRXCA) command.
- Select Option **17** on the *Configure TCP/IP Applications* (CFGTCPAPP) display.

Note: You must have *IOSYSCFG special authority to make changes to the REXEC attributes with the CHGRXCA command.

Figure 240. Change REXEC Attributes (CHGRXCA)

```
Change REXEC Attributes (CHGRXCA)

Type choices, press Enter.

Autostart server . . . . . *YES          *YES, *NO, *SAME
Number of initial servers . . . 2          1-20, *SAME, *DFT
Inactivity timeout . . . . . 300          1-2147483647, *SAME, *DFT
Coded character set identifier 00437      1-65533, *SAME, *DFT
```

REXEC Command Considerations

The REXEC server is restricted to running commands that are allowed in batch jobs. The command must have *BATCH as one of the *Where allowed to run* values.

The maximum length of a command that the REXEC server can process is 4000 bytes. Some REXEC clients limit the command to a smaller length.

For spawned paths, the program that runs in the child process must be either a program object in the QSYS.LIB file system (*PGM object) or a shell script. You must specify the path with the proper syntax for the file system in which the file resides.

For Qshell commands, you can put the same commands that you would enter at an interactive command line into a non-interactive shell script.

Selecting a Command Processor

You can use the REXEC server command processing selection exit program (QIBM_QTMX_SVR_SELECT) to select which command processor the REXEC server uses to run the submitted command. (If you do not use an exit program, the REXEC server uses the AS/400 Control Language (QCAPCMD) processor.) The allowed command processors are:

- AS/400 Control Language (QCAPCMD)
- Qshell interpreter
- Spawned path (a shell script or program object)

Because data conversion is optional for the Qshell and spawn options, the exit program also selects whether the REXEC server performs ASCII-EBCDIC conversions on the stdin, stdout, and stderr streams.

REXEC Connection Usage

The REXEC protocol allows a REXEC client to specify whether to use one or two connections for returning data.

For AS/400 CL command processing

If you choose AS/400 CL command processing and two connections, normal output returns on the first connection, and error output returns on the second connection. The REXEC server returns all spooled data that is written to the default printer file (*PRTF). This includes data that is written to the screen if the command is run in an interactive job. Any messages written to the job log return to the client on the second connection.

If the client specifies that all data is returned on a single connection, the job log messages are returned first, followed by any spooled output.

For Qshell and spawned path command processing

For Qshell or spawned path command processing, the REXEC server by default returns normal output on the first connection and error output on the second connection. (The REXEC stdin, stdout, and stderr streams are mapped to file descriptors 0, 1, and 2 respectively, and the QIBM_USE_DESCRIPTOR_STDIO environment variable is set to Y.) These options allow you to redirect input and output.

Choosing the Qshell command processor sets these environment variables:

- `TERMINAL_TYPE=REMOTE`
- `PATH=/usr/bin`
- `LOGNAME= user`, where *user* is the user profile
- `HOME=homedir`, where *homedir* is the user's home directory

The child job inherits any other environment variables that the exit program sets.

Spawned child processes are batch jobs or prestart jobs. They cannot do interactive I/O. See *ILE C for AS/400 Programmer's Guide*, SC09-2712-01 for complete details on this support.

Spooled Output Considerations

Note: This section applies only to AS/400 CL commands.

The REXEC server overrides the default printer file (*PRTF) to capture spooled output. Any resulting spool files are tagged with the user data field set to REXECSVR. After the REXEC server runs the specified command, each spooled file with this user data tag is retrieved, returned to the client, and then deleted. If more than one spool file is created, the files are processed in the order created, as determined by the spool file number.

If the command or program run through REXEC performs its own print file override and changes the user data, the REXEC server is unable to capture and return the resulting spooled data.

Client Considerations

The AS/400 REXEC client (RUNRMTCMD) uses a single connection for returned data, which is written to a spooled file on the client system. The RUNRMTCMD command is documented in *CL Reference (Abridged)*, SC41-5722.

The UNIX, OS/2, Windows 95, and Windows NT REXEC clients all use two connections, returning the normal output to the stdout stream and the error output to the stderr stream.

The VM REXEC client uses a single connection for the returned data, which is written to the console of the user.

REXEC Server Jobs and Job Names

REXEC server jobs start when you run the STRTCP command and set the REXEC AUTOSTART parameter to *YES. You can also start REXEC server jobs by running the STRTCP SVR command with a SERVER parameter of *REXEC or *ALL. These jobs run in the QSYSWRK subsystem. Their purpose is monitoring and processing requests from REXEC client users. The format for the names of these jobs is QTRXCnnnnn, where nnnn is a 5-digit decimal number.

To work with jobs in the QSYSWRK subsystem, including REXEC server jobs, specify the following command:

```
WRKSBSJOB SBS(QSYSWRK)
```

If you choose to have commands processed by the Qshell command interpreter, you start Qshell by using the spawn() application program interface (API) to create a child job.

If you choose to have commands interpreted as spawn path names, the REXEC server treats command strings as path names and passes them to the spawn() API. Spawned child processes are batch jobs or prestart jobs. Shell scripts are allowed for the child process. If you specify a shell script, the appropriate shell interpreter program is called. The shell script must be a text file and must contain this format on the first line of the file:#!interpreter_path <options>.

Creating REXEC Server Spooled Job Logs

The REXEC server automatically writes a server job log to a spooled file when it ends with an error.

To have a spooled job log produced at the end of each REXEC session and each time the REXEC server ends, use the CHGJOB command, as follows:

```
CHGJOB JOB(QTCP/QTMXRCS) LOG(4 00 *SECLVL)
```

To obtain a spooled job log only when a server ends, use the CHGJOB command, as follows:

```
CHGJOB JOB(QTCP/QTMXRCS) LOG(4 00 *NOLIST)
```

Exit Points for Controlling REXEC Server

Available exit points give you additional control over the REXEC server. The TCP/IP request validation exit point (QIBM_QTMX_SERVER_REQ) provides additional control for restricting an operation. The REXEC server command processing selection exit point (QIBM_QTMX_SVR_SELECT) allows you to specify which command processor the REXEC server uses for interpreting and running your commands. If you add exit programs to both of these exit points, REXEC server first calls the program that you add to QIBM_QTMX_SERVER_REQ. The TCP/IP server logon exit point (QIBM_QTMX_SVR_LOGON) provides additional control over authenticating a user and setting up the user's environment for the REXEC server. For a detailed description of these exit points and how to use them, see "Appendix E. TCP/IP Application Exit Points and Programs" on page 535.

Chapter 17. DHCP Server

Dynamic Host Configuration Protocol, or DHCP, provides a framework for passing configuration information to hosts on a TCP/IP network. DHCP functions either as a DHCP server or as a BOOTP/DHCP Relay Agent. If DHCP functions as a DHCP server, it processes DHCP packets on the local system. If DHCP functions as a BOOTP/DHCP Relay Agent, it relays DHCP and BOOTP packets on the local system but does not process them. The Relay Agent forwards the packets to one or more different server IP addresses. The Relay Agent uses port 67, the same port that the BOOTP or DHCP server uses.

During configuration, you must specify the IP addresses that the Relay Agent uses for forwarding. When the Relay Agent forwards a packet, it inserts its address into the packet. It does this so the DHCP or BOOTP server can return a response to that Relay Agent. For more information on DHCP Relay Agents, see “DHCP Relay Agent” on page 420.

DHCP Overview

DHCP enables client machines to get network configuration information, including an Internet Protocol (IP) address, from a central DHCP server. The administrator sets the DHCP server to assign the network information to each client, which makes the process transparent to the user. DHCP saves time, reduces errors, and insures the consistency of the configurations for mobile and non-mobile users.

What is DHCP?

DHCP is a client/server protocol that enables you to centrally locate and dynamically distribute configuration information, including IP addresses.

DHCP is based on the Bootstrap Protocol, or BOOTP, and adds the capability for automatically allocating reusable network addresses. It also adds the capability for distributing additional host configuration options. DHCP clients and servers use existing BOOTP/DHCP Relay agents.

DHCP defines IP address allocation policies, including the following:

Dynamic

A DHCP server assigns an IP address to a requesting BOOTP or DHCP client from a range of available addresses.

Static

A DHCP server administrator assigns a static, pre-defined address that is reserved for a specific BOOTP or DHCP client.

DHCP provides the following lease policies for IP addresses:

Temporary

An IP address is temporarily *leased* to a DHCP client. To continue using the address, a DHCP client that does not have a permanent lease must periodically request the renewal of its lease on its current IP address. The process of renewing leased IP addresses occurs dynamically as part of the DHCP protocols. It is not generally visible to end users.

Permanent

A BOOTP or DHCP client leases an IP address for an infinite period of time. No process of lease renewal is required.

Planning for DHCP

Before you put DHCP into effect in your network, you need to make the following decisions:

- How many DHCP servers do you need?
- Do you already have BOOTP servers in your network?
- Do you have hosts with special requirements?
- What is a reasonable lease time?

How Many DHCP Servers do you Need?

The number of servers that you need depends largely on the number of subnets you have and the number of DHCP clients that you plan to support. It also depends on the lease time that you choose and whether your routers (or individual computers on the subnets) are enabled with BOOTP Relay. Keep in mind that the DHCP protocols do not currently define server-to-server communication. Therefore, they cannot share information, and one DHCP server cannot perform as a *hot backup* if the other one fails.

DHCP clients send broadcast messages. By design, broadcast messages do not cross subnets. To allow the client's messages to be forwarded outside its subnet, you must configure a BOOTP/DHCP Relay Agent. Otherwise, you must configure a DHCP server on each subnet.

Note: Many routers available today can act as a BOOTP/DHCP Relay Agent. Software is also available for some operating system platforms that enable a regular computer on the network to act as a BOOTP/DHCP Relay Agent. On the AS/400 system, BOOTP/DHCP Relay capability comes standard as part of the DHCP server.

Using a Single DHCP Server: If you choose to use a single DHCP server to serve hosts on a subnet, you must consider the effects if the single server fails. Generally, the failure of a server affects only those DHCP clients that attempt to join the network. DHCP clients already on the network typically continue operating unaffected until their lease expires. However, clients with a short lease time might lose their network access before you can restart the server.

Using Multiple DHCP Servers: To avoid a single point of failure, you can configure two or more DHCP servers to serve the same subnet. If one server fails, the other continues to serve the subnet. Each of the DHCP servers must be accessible either by direct attachment to the subnet or by using a BOOTP/DHCP Relay Agent. Remember that you cannot run more than one DHCP server on any individual system. Multiple DHCP servers require multiple systems.

Because two DHCP servers cannot serve the same addresses, address pools that you define for a subnet must be unique across DHCP servers. Therefore, when using two or more DHCP servers to serve a particular subnet, you must divide the complete list of addresses for that subnet among the servers. Configure one server with an address pool that consists of 70% of the available addresses for the subnet. Configure the other server with an address pool that consists of the remaining 30% of the available addresses.

Using multiple DHCP servers decreases the probability of having a DHCP-related network access failure, but it does not guarantee against it. If a DHCP server for a particular subnet fails, the other DHCP server might be unable to service all of the requests from new clients. This can use every available address in the server's limited pool.

You can bias which DHCP server uses all of its addresses first. DHCP clients tend to select the DHCP server that is offering more options. To bias service toward the DHCP server with 70% of the available addresses, offer fewer DHCP options from the server that holds 30% of the available addresses for the subnet.

Do you Already have BOOTP Servers in your Network?

If you already have BOOTP clients and servers in your network, consider replacing your BOOTP servers with DHCP servers. DHCP servers can optionally serve BOOTP clients the same IP configuration information as current BOOTP servers.

If you cannot replace your BOOTP servers with DHCP servers or want both to serve your network, perform the following steps:

1. Turn off BOOTP support in your DHCP server.
2. Make sure that your BOOTP servers and DHCP servers do not give out the same IP addresses.
3. Configure any BOOTP/DHCP Relay Agents to forward the BOOTP/DHCP broadcasts to both the appropriate BOOTP and DHCP servers.

You also need to consider that BOOTP and DHCP servers cannot both run on any individual system. To continue running your BOOTP servers in addition to your DHCP servers, you need to run them on separate systems.

A DHCP server allocates a permanent IP address to a BOOTP client. If subnets are renumbered in such a way that a BOOTP-assigned address is unusable, you must restart the BOOTP client and obtain a new IP address.

Do you have Hosts with Special Requirements?

You might have hosts with individual or special administrative needs, such as the following:

Permanent lease

Assign permanent leases to designated hosts by specifying an infinite lease time. Additionally, the DHCP server allocates a permanent lease to BOOTP clients that explicitly request it, as long as support for BOOTP clients is enabled. The DHCP server also allocates a permanent lease to DHCP hosts that explicitly request it.

Specific IP address

Reserve a specific address and configuration parameters for a specific DHCP (or BOOTP) client host on a particular subnet.

Specific configuration parameters

Allocate specific configuration information to a client regardless of its subnet.

Manually defined workstations

Explicitly exclude addresses from DHCP subnets for existing hosts that do not use DHCP or BOOTP for configuring their IP network access.

DHCP servers and clients automatically check to see if an IP address is in use before allocating or using it. However, they are unable to detect addresses of

manually defined hosts that are either turned off or temporarily off the network. In that case, duplicate address problems might occur when a manually defined host re-accesses the network and its IP address is not explicitly excluded.

What is a Reasonable Lease Time?

The default lease time is 24 hours. The lease time that you choose depends largely on your needs, including the following:

- The number of hosts to support compared to the number of available addresses.
If you have more hosts than addresses, choose a short lease time of one to two hours. This ensures that unused addresses are returned to the pool as soon as possible. Keep in mind that the DHCP lease time affects your network operation and performance.
 - Shorter lease times increase the amount of network traffic due to DHCP lease renewal requests. For example, if you set a lease time of 5 minutes, each client sends a renewal request about every 2.5 minutes.
 - Longer lease times limit your ability to reuse IP addresses. Extremely long lease times also delay configuration changes that occur when a client restarts or renews a lease.
- The time available to make network changes.
Hosts receive changes to configuration information when they are restarted or when they renew their lease. Allow a timely and adequate window to make these changes. For example, if you usually make changes overnight, you might assign a lease time of 12 hours.
- The number of available DHCP servers.
If you have only a few DHCP servers for a large network, choose a longer lease time to minimize the impact of server downtime.

Setting Up a DHCP Network

The IBM DHCP server provides configuration information to clients. This information is based both on statements contained within the server's configuration file and on information provided by the client. The server's configuration file defines the policy for allocating IP addresses and other configuration parameters. The file is a *map* that the server uses to determine what information to provide to the requesting client.

Before you start the DHCP server, use Operations Navigator to create or change the DHCP server configuration file. Figure 241 on page 409 represents the DHCP Server Configuration window.

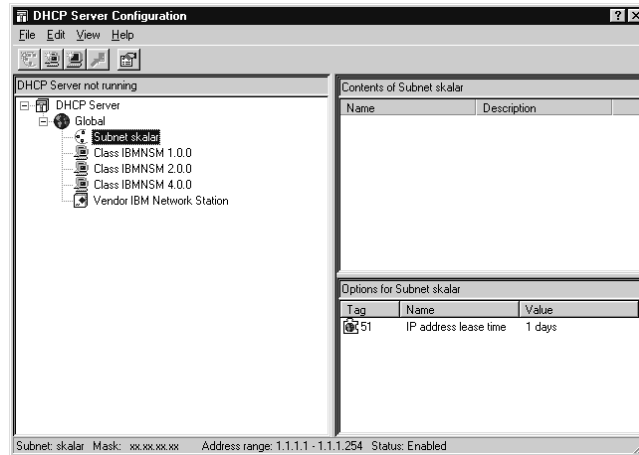


Figure 241. DHCP Server Configuration

Once the DHCP server is running, you can also make dynamic changes to the configuration by using Operations Navigator to modify the configuration file and re-initialize the DHCP server.

Creating a Scoped Network

Create a hierarchy of configuration parameters for a DHCP network. Specify some configuration values to serve globally to all clients, and specify other configuration values to serve only to certain clients. Serving different configuration information to clients is often based on network location, equipment vendor, or user characteristics.

Depending on your configuration, specify subnets, classes, vendors, and clients to provide configuration information to different groups of clients, as follows:

- When defined globally, client, vendor, or class options are available to DHCP clients regardless of their network location.
Parameters specified for a subnet, class, or client are considered local to the subnet, class, or client. A client defined within a subnet inherits both the global options and the options defined for that subnet. If a parameter is specified in more than one level in the network hierarchy, the lowest level (which is the most specific) is used.
- Use the **subnet** to specify configuration parameters for one subnet for a specific location in your network or enterprise.
- Use the **class** to configure DHCP classes to provide unique configuration information from the server to clients that identify themselves as belonging to that class. For example, a group of clients can all use a shared printer.
- Use a **vendor** to provide unique configuration information to clients that identify themselves as using a specific vendor's equipment or software. Specially defined options are served to these clients.
- Use a **client** in the DHCP server configuration file to serve specified options to a specific client or to exclude that client from service. You can also use a client to exclude IP addresses from service.

Defining Scoped Statements

The concept of scoped statements in a DHCP server configuration file is shown in Figure 242.

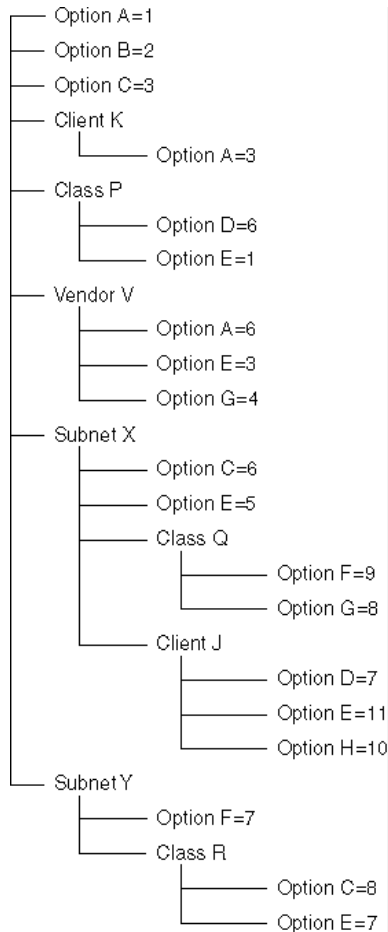


Figure 242. DHCP Hierarchy

In this example, you see the following:

- Options A, B, and C are global. They are inherited by all clients in the network unless overridden by a value for the same option at a lower level in the network.
- DHCP options that you define at a global level are overridden by options defined in a globally defined client for that client only.
- DHCP options that you define at the subnet level override options defined at the global level for clients falling in that subnet.
- Clients that you have not specifically defined automatically fall into a specific location in the hierarchy based on their current network location.
- A client that requests a class must fall within a subnet in which the class is defined, or you must globally define the class, before the client receives options from the class.
- Options defined for a client override options in a class requested by the client, provided the class is legal for that client.
- Vendor options are always defined globally so that every client that requests a vendor is served the same options.

The following examples show requesting clients and the options and values that they are served:

- Client K, if located on subnet Y and requesting class R, is served the following options and values:

A 3

B 2

C 8

E 7

F 7

- Client K, if located on subnet X, is served the following options and values:

A 3

B 2

C 6

E 5

- Client J, if located in subnet X and requesting class P and vendor V, is served the following options and values:

A 1

B 2

C 6

D 7

E 11

F Not served

H 10

Vendor option A

6

Vendor option E

3

Vendor option G

4

- Client J, if located in subnet X and requesting class Q and vendor V, is served the following options and values:

A 1

B 2

C 6

D 7

E 11

F 9

G 8

H 10

Vendor option A

6

Vendor option E

3

Vendor option G

4

- Client J, if located in subnet X and requesting class R, is served the following options and values:

A 1

B 2

C 6

D 7

E 11

H 10

- Any client other than J or K that is not located in either subnet X or subnet Y is not served any options or values.
- Any client other than J or K that is located in subnet X and requesting class Q and Vendor V receives the following options or values:

A

B

C

E

F

G

Vendor option A

Vendor option E

Vendor option G

Specifying DHCP Options

DHCP allows you to specify *options*, also known as BOOTP vendor extensions, to provide additional configuration information to the client. Internet Engineering Task Force (IETF) Request for Comment (RFC) 2132 defines the options that you can use.

Each option is identified by a numeric code. Examples of options that DHCP servers pass to DHCP (and in many cases to BOOTP) clients are as follows:

- 1 Subnet Mask
- 3 Gateway/Router Address
- 6 Domain Name Server Addresses
- 12 Host Name
- 15 Domain Name

Note: You cannot find a list of all the possible DHCP options and their descriptions in this document. Reference the actual IETF RFC documentation for a more thorough explanation of how DHCP options work and for descriptions of the

actual DHCP option numbers themselves. See “Request for Comment and Internet Draft Documents” for information on what IETF RFCs are and how to view them yourself.

Architected DHCP Options

Architected options 0 through 127 and option 255 are reserved for definition by RFCs. The DHCP server, the DHCP client, or both the server and client use options in this set. The administrator can change most architected options, but some options are reserved for the exclusive use of the client and server programs themselves.

Examples of architected options that the administrator must not configure at the DHCP server include the following:

- 52** Option overload
- 53** DHCP message type
- 54** Server identifier
- 55** Parameter request list
- 56** Message
- 57** Maximum DHCP message size
- 60** Class identifier
- 77** User class

User-defined DHCP Options

Options 128 through 254 represent user-defined options that administrators define to pass information to the DHCP client to accomplish site-specific configuration parameters.

The format of user-defined options is always a string of hexadecimal bytes, and the server passes the specified value to the client. In order for it to be of any use, however, the client must have some special, application-specific program or command file to process the value.

Request for Comment and Internet Draft Documents

The Internet is governed by protocols that are defined in Internet Engineering Task Force (IETF) Request for Comment (RFC) documents. RFCs outline existing protocols, suggest new protocols, and establish standards for the Internet protocol suite. Internet drafts are proposals, techniques, and mechanisms that document IETF work in progress. Online copies of RFCs and Internet Drafts are available from IETF.

To access RFCs or Internet Drafts, point your Web browser at the following URL, which links to the Internet Documentation and IETF Information home page:

<http://www.ietf.org>

For a better understanding of DHCP, see the following RFC documents:

RFC 2131

Dynamic Host Configuration Protocol

RFC 2132

DHCP Options and BOOTP Vendor Extensions

RFC 951

Bootstrap Protocol

RFC 1542

Clarifications and Extensions to the Bootstrap Protocol

Internet Draft titled

The User Class Option for DHCP

Accessing DHCP Functions through Operations Navigator

You can access DHCP server functions from either a command line interface or Operations Navigator. Not all DHCP functions are available on both interfaces. However, most DHCP functions are available *only* through Operations Navigator.

This chapter discusses how you start, stop, and configure the DHCP server from the command line interface. This chapter does not document any of the Operations Navigator functions. See the online help for Operations Navigator for information about using the Operations Navigator for DHCP functions.

To access DHCP server functions through Operations Navigator, perform the following steps:

1. Double-click your AS/400 server in the main tree view of Operations Navigator.
2. Double-click **Network**.
3. Double-click **Servers**.
4. Double-click **TCP/IP**.
5. Double-click **DHCP**. Figure 243 represents the DHCP Server Configuration window.

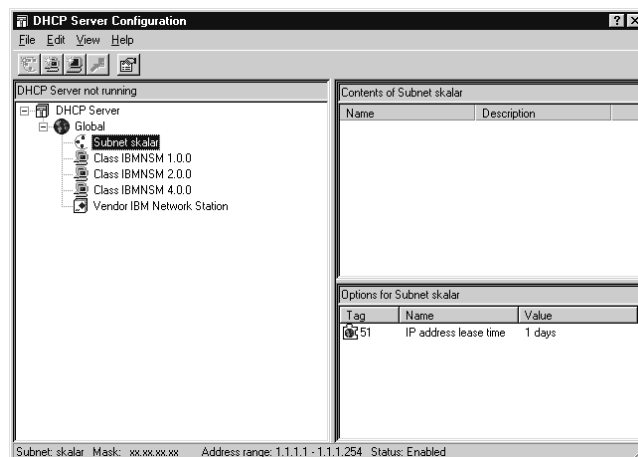


Figure 243. DHCP Server Configuration

The online help in Operations Navigator assists you with the following:

- Adding a new subnet to an existing DHCP configuration.
- Changing an existing subnet.
- Configuring a DHCP Relay Agent.
- Configuring DHCP to assign a permanent IP address.

- Creating a new DHCP configuration.
- Ending the DHCP server.
- Getting authority to configure the DHCP server.
- Making IP addresses available for IP address management.
- Migrating BOOTP to a new DHCP configuration.
- Migrating BOOTP to an existing DHCP configuration.
- Setting options with DHCP configuration.
- Starting the DHCP server.
- Starting the DHCP server automatically when TCP/IP starts.

Note: Although some of the functionality of the command line interface and the Operations Navigator is the same, the actual menu commands and parameters are not necessarily the same.

Starting and Ending the DHCP Server from the Command Line Interface

The instructions in this section apply only to the command line interface. For information on starting and ending the DHCP server with Operations Navigator, see the online help in Operations Navigator.

Starting the DHCP Server

To start the DHCP server, specify the *Start TCP Server* (STRTCP) command. This command has no parameters, and it attempts to start all of the TCP/IP servers that have been designated for autostart, as follows:

```
STRTCP
```

The DHCP server starts with this command if the AUTOSTART parameter in the DHCP Server attributes is set to *YES.

Note: The AUTOSTART parameter is a part of the CHGDHCPA command.

You can use the STRTCP SVR command to start the DHCP server outside of the full-TCP start. STRTCP SVR has a SERVER parameter where you can specify which TCP servers you want to start, as follows:

```
STRTCP SVR SERVER(*DHCP)
```

STRTCP SVR also has the following RESTART parameter:

```
RESTART(*DHCP)
```

Automatically Starting the DHCP Server

The AUTOSTART parameter of the CHGDHCPA command affects the operation of the STRTCP command. It does not have an effect on how the STRTCP SVR command works if DHCP is explicitly requested with SERVER(*DHCP). However, it does have an effect if STRTCP SVR SERVER (*ALL) is issued. If you run the STRTCP SVR SERVER (*DHCP) command when the DHCP server is running, you receive a diagnostic message. In this case, the STRTCP SVR command ignores the AUTOSTART parameter value.

If you use the STRTCPSVR command to restart the DHCP server when the DHCP server is not running, the server ignores the restart instruction and simply starts.

Notes:

1. If you have installed both DHCP and BOOTP servers, only one of the servers can run. Therefore, an AUTOSTART parameter value of *YES is not allowed for both BOOTP and DHCP. Because of this, STRTCP never attempts to start them both.
2. Use the *Change DHCP Attributes* (CHGDHCPA) command to change the startup attributes of the DHCP server. The changes take effect the next time you start the DHCP server by either the *Start TCP/IP* (STRTCP) command or the *Start TCP/IP Server* (STRTCPSVR) command. You must have *IOSYSCFG special authority to use this command.

The AUTOSTART parameter specifies whether to start the DHCP server automatically when TCP/IP is started by the STRTCP command. When the DHCP server is started by the STRTCPSVR command, the DHCP server ignores the AUTOSTART parameter and starts regardless of the value of this parameter.

There is an exception. If the STRTCPSVR *ALL command is issued, all TCP/IP servers that have been configured are started. However, a BOOTP and DHCP server cannot both run on the same machine at the same time. If you issue the STRTCPSVR *ALL command, the system first checks to see if both a BOOTP and DHCP server job are configured. If both are configured, the system checks the AUTOSTART attribute for each server.

If the AUTOSTART attribute for one of the servers is set to *YES and the other is set to *NO, the system starts the server with the AUTOSTART attribute set to *YES. If both the BOOTP and DHCP AUTOSTART attributes are set to *NO, the DHCP server starts.

3. If you want the BOOTP server to function as the default, specify that change in the *Change BOOTP Attributes* (CHGBPA) command, not in the DHCP configurations.

Ending the DHCP Server

To end the DHCP server, specify the following *End TCP/IP Server* (ENDTCPSVR) command with the server attribute set to *DHCP:

```
ENDTCPSVR SERVER(*DHCP)
```

Changing DHCP Attributes

The *Change DHCP Attributes* (CHGDHCPA) command changes the startup attributes of the DHCP server. The change takes effect the next time you start the DHCP server from either the *Start TCP/IP* (STRTCP) command or the *Start TCP/IP Server* (STRTCPSVR) command.

Note: You must have *IOSYSCFG special authority to make changes to the DHCP attributes with the CHGDHCPA command.

Exit Points for a DHCP Server

The DHCP processing server (*not* the BOOTP/Relay Agent) provides three exit points. These exit points are as follows:

- QIBM_QTOD_DHCP_REQ - DHCP Request Packet Validation
- QIBM_QTOD_DHCP_ABND - DHCP Address Binding Notification
- QIBM_QTOD_DHCP_ARLS - DHCP Address Release Notification

See the *System API Reference*, SC41-5801-03 for a detailed description of these exit points and how to use them.

Examples of DHCP Configurations

Configuring DHCP for a Local Area Network

Figure 244 illustrates a Local Area Network (LAN) with a server and two of the network's client machines. In a network using DHCP, the server automatically assigns configuration information to each client that includes IP address information. Using DHCP in even a simple LAN, such as the one in the illustration, saves administrative time and trouble. The automatic configuration assignment is transparent to the user.

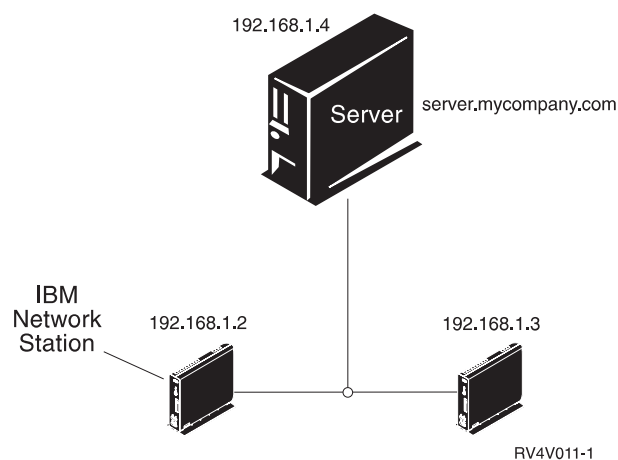


Figure 244. LAN with Server and Two Clients

Address Range

In order for the server to automatically assign IP addresses, you must define an address pool from which the server chooses the IP addresses. The starting address for this example is 192.168.1.2, and the ending address is 192.168.1.4. Both the starting address and the ending address fall within the range, and the server considers them candidates for assignment to a client machine.

Configuring DHCP for a Local Area Network with a Router

Figure 245 on page 418 illustrates a LAN with a server, a router, and five of the network's client machines. In a network that uses DHCP, the server automatically assigns configuration information that includes IP address information. The DHCP

server automatically assigns the network information to each client, which makes the process transparent to the user.

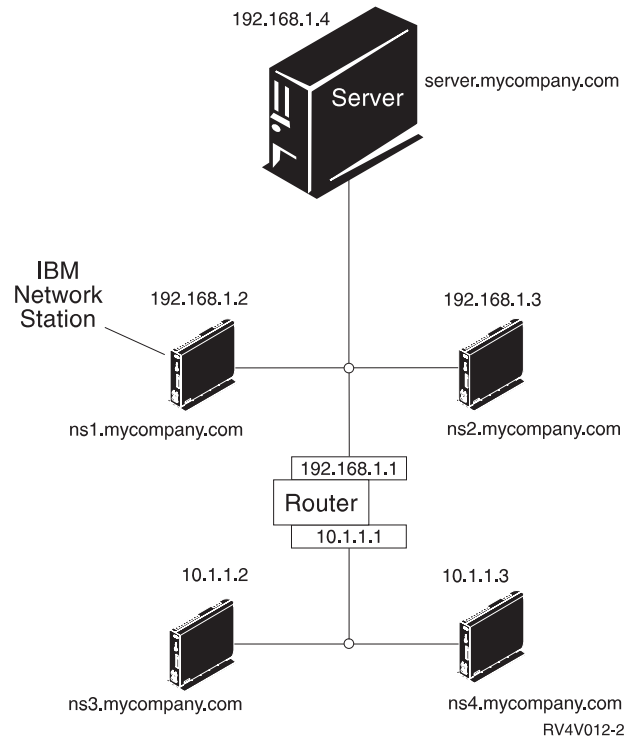


Figure 245. LAN with Server, Router and Five Clients

Subnet Options

The example shows two subnets. The starting address for the address range for one subnet is 192.168.1.1, and the ending address is 192.168.1.4. The starting address for the address range in the second subnet is 10.1.1.1, and the ending address is 10.1.1.3.

Using DHCP to Configure Clients Attached to a Twinax Workstation Controller

Figure 246 on page 419 illustrates a twinaxial network with a server, three of the network's client machines, and a workstation controller. In a network that uses DHCP, the server automatically assigns configuration information that includes IP address information.

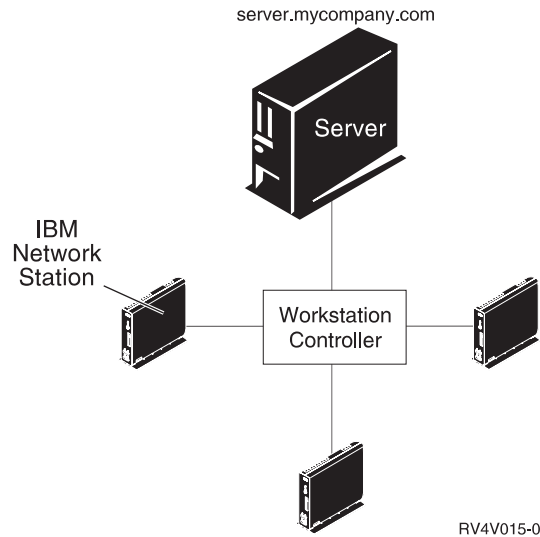


Figure 246. Twinax Network with Server, Three Clients and Workstation Controller

For more information on Twinax networks, see *Network Station Manager Installation and Use*, SC41-0664-00.

Migrating an Existing BOOTP Configuration

Migrating to DHCP means that BOOTP client data on the server becomes available to the DHCP server, and DHCP can then service those clients. BOOTP and DHCP cannot operate at the same time on the same system. You can disable the BOOTP server at any time.

Unless your system is new, it might have an existing BOOTP configuration. In this instance, you can choose to migrate client data now and disable the BOOTP server at a later time. Be aware, however, that a client data migration to DHCP represents a point-in-time. This means that the client data as migrated to DHCP might not necessarily match the client data as it is known by BOOTP at the time you disable it. Consider migrating BOOTP client data and then turning off the BOOTP server at a time when clients are not on the system.

In Operations Navigator, you can migrate a BOOTP configuration to DHCP by using either the New DHCP Configuration wizard or the **Migrate BOOTP** dialog. Whenever you use the New DHCP Configuration wizard to create a new configuration, the wizard checks for an existing BOOTP configuration. If one is detected, the wizard asks if you want to migrate the configuration to the DHCP server configuration.

For instructions on migrating BOOTP to a new or existing DHCP configuration, see the online help in Operations Navigator.

DHCP Relay Agent

The DHCP server functions as either a DHCP server or a BOOTP/DHCP Relay Agent. If the server functions as a DHCP server, it processes DHCP packets on the local system. If the DHCP server functions as a BOOTP/DHCP Relay Agent, it relays DHCP and BOOTP packets on the local system but does not process them. The Relay Agent forwards the packets to one or more different server IP addresses.

The BOOTP/DHCP Relay Agent handles both BOOTP and DHCP client requests for IP addresses and forwards those requests to remote destination addresses.

The BOOTP/DHCP Relay Agent runs on the local subnet. The Relay Agent intercepts client BOOTP or DHCP packet broadcasts. It then forwards them on to the non-local DHCP server, which sends the response back to the Relay Agent. The Relay Agent then forwards the response to the client on the local subnet. It uses the port 67, the same port that the BOOTP or DHCP server uses.

Specify during configuration the IP addresses that the Relay Agent uses for forwarding.

When the Relay Agent forwards a packet, it inserts its address into the packet so the DHCP server can return a response to that Relay Agent.

For instructions on configuring a BOOTP/DHCP Relay Agent, see the online help in Operations Navigator.

Adding Network Stations

To add Network Stations to your DHCP environment, run the Network Station Setup Assistant. For more information on the Network Station Setup Assistant, see *Network Station Manager Installation and Use, SC41-0664-00*.

Chapter 18. AS/400 Domain Name System (DNS)

The Domain Name System (DNS) is an advanced system for managing the host names that are associated with Internet Protocol (IP) addresses on TCP/IP networks.

Material on DNS is covered in the **AS/400e Information Center** under the **TCP/IP** topic. For more information see "TCP/IP Topics in the Information Center" on page xv.

On AS/400, DNS configuration is only available through AS/400 Operations Navigator. Operations Navigator is a graphical user interface, and a part of Client Access for Windows 95/NT.

For a list of recommended DNS publications, see "Additional DNS documentation" on page 422.

How DNS works

Material on DNS is covered in the **AS/400e Information Center** under the **TCP/IP** topic. For more information see "TCP/IP Topics in the Information Center" on page xv. See "Additional DNS documentation" on page 422 for a list of DNS publications that contain more detailed information.

The Domain Name System (DNS) was developed to eliminate the problems and limitations of using host tables only to resolve host names on large TCP/IP networks. DNS is the naming service of intranets and the Internet. Virtually all TCP/IP software, including electronic mail, now uses DNS. Applications such as Telnet, File Transfer Protocol (FTP), and HyperText Transfer Protocol (HTTP) browsers use DNS as well.

What does DNS do?

The primary job of DNS is to translate TCP/IP host names to Internet Protocol (IP) addresses. Host names are easy for people to remember, but IP addresses are what TCP/IP uses to make connections between hosts across the network.

Name resolution and mapping

DNS translates or **resolves** host names to IP addresses. It can also resolve an IP address to a host name. When DNS associates an IP address to a host name, the IP address is said to be **mapped** to the host name.

What is DNS?

DNS is a system; not one thing, but many. It is a method of logically dividing TCP/IP networks into manageable units that are called **domains**. It organizes these units or domains into a **hierarchy** whose structure is similar to the roots of a tree. It is a method for naming both the domains in the hierarchy and the hosts in the domains, so that no two names are identical.

DNS is a distributed database

DNS is designed so that the host name and IP address information of a network can be stored in many different locations around the network. This helps to distribute the name resolution workload, and makes host name management easier. Usually, only the host names and IP addresses of the hosts within a domain are stored in that domain's database. However, DNS servers have the ability to share their domain information with other DNS servers.

DNS servers

DNS servers do most of the DNS work. It is their job to receive requests (called **queries**) for network information, and return answers. If the answer to a query is not in a DNS server's domain, the DNS server has the ability to search out into the DNS hierarchy.

DNS resolvers

If there is a server, there must be a client. In DNS, the client is called a **resolver**. Sometimes DNS resolvers are built into each TCP/IP application. Sometimes a central resolver does all of the client work for all of the TCP/IP applications on a host. In either case, the resolver works invisibly and automatically each time a TCP/IP application is used to communicate with another host.

Additional DNS documentation

You can find additional documentation for the Domain Name System in these sources:

- *DNS and BIND third edition* by Paul Albitz and Cricket Liu. Published by O'Reilly & Associates, Inc., 1998. ISBN number: 1-56592-512-2. This is the most definitive source on DNS.
- *AS/400 TCP/IP Autoconfiguration: DNS and DHCP Support*. This is an IBM Redbook, order number: SG24-5147. It is available on the World Wide Web at the following site: <http://www.redbooks.ibm.com/>
- *AS/400 Tips and Tools for Securing Your AS/400*. This is an IBM White book, order number: SC41-5300. It has important DNS security information. It is available on the World Wide Web at the following URL:
<http://as400bks.rochester.ibm.com/cgi-bin/bookmgr/bookmgr.cmd/DOCNUM/SC41-5300>
- RFC 1034 *Domain names concepts and functions*.
- RFC 1035 *Domain names implementations and specifications*.

RFCs on the World Wide Web

RFCs are posted in many places on the Web. DNS-related RFCs are available at the following address: <http://www.dns.net/dnsrd/docs/rfc.html>

Chapter 19. Client SOCKS Support

AS/400 client SOCKS support enables programs that use (AF_INET SOCK_STREAM) sockets to communicate with server programs that run on systems outside a firewall. In addition, by using client SOCKS support, both AS/400 FTP and AS/400 TELNET client connections can be directed through a firewall. The key advantage to AS/400 client SOCKS support is that it enables client applications to access a SOCKS server transparently without changing any client code. AS/400 client SOCKS support operates with any SOCKS server that supports Version 4 SOCKS protocols.

For information on AS/400 client SOCKS support and SOCKS server concepts, see the *Sockets Programming*, SC41-5422-03 book. For instructions on how to configure your SOCKS server on the AS/400 firewall or for information on firewall concepts, see the *Getting Started with IBM Firewall for AS/400*, SC41-5424-02 book or go to <http://www.as400.ibm.com/firewall>.

Accessing SOCKS Functions through Operations Navigator

For information on how to configure an AS/400 host to use a SOCKS server, use the AS/400 Operations Navigator. Define the client SOCKS configuration entries that you need by using the SOCKS tab found under the Operations Navigator function of Client Access/400 for Windows 95/NT. The SOCKS tab has substantial help for configuring the client system for AS/400 client SOCKS support.

To access SOCKS server functions through Operations Navigator, perform the following steps:

1. Double-click your AS/400 server in the main tree view of Operations Navigator.
2. Double-click **Network**.
3. Double-click **Protocols**.
4. Double-click **TCP/IP**.
5. Click the **SOCKS** tab.

For a complete description of the sockets APIs, see *System API Reference*, SC41-5801-03.

Chapter 20. TCP/IP Performance

The following are performance items that should be considered when using TCP/IP.

*BASE Pool Size

The TCP/IP protocol and application code always runs in the *BASE pool on the AS/400 system. If the *BASE pool is not given enough storage, TCP/IP performance, especially SMTP performance, can be adversely affected.

Although it is possible to run in less than 4000 KB of storage to perform well when running both FTP and SMTP sessions, it is suggested that the *BASE pool be configured to use at least 4000 KB of storage. You can use the WRKSYSSTS to view and change pool sizes on the AS/400 system. Pool 2 is the base pool. Another alternative is to change the pool in which the TCP/IP jobs run.

TCP/IP Jobs

TCP/IP jobs, like other jobs on your system, are created from job descriptions and associated classes. The job descriptions and classes should be adequate in most cases; however, they may be changed to fit your configuration. The TCP/IP job descriptions, classes, and subsystem descriptions can be found in the QTCP or the QSYS library that was loaded in your system when TCP/IP was installed.

Each application has a job description associated with it. This job description has a number of items associated with it that define how the application runs on the AS/400. One of these pieces of information is the routing entry compare value. This value identifies which routing entry in a subsystem description is used when this job is submitted. By changing that routing entry, you can select in which storage pool to run the jobs for a particular application. For information on compare values, see *Work Management*.

Other items that can be changed or selected on a job description include the job priority, the logging level for messages, and the initial library list.

If the storage pool that you select to run the TCP/IP application jobs in is not large enough, excessive paging can occur. This directly affects performance on the AS/400 and the performance of the applications.

TCP/IP Protocol Support Provided by IOP

AS/400 TCP/IP protocol support runs down in the AS/400 System Licensed Internal Code, at the same level as LU 6.2 and APPN*. One of the goals of integrating TCP/IP into the AS/400 System Licensed Internal Code is to provide performance and capacity comparable to APPC.

Further, moving some functions that are normally done by the TCP/IP software into the IOP reduces interactions between the system and the input/output processor (input-output processor (IOP)). These functions may include:

- Checksum calculation of outgoing TCP and UPD datagrams (prior to V4R4)
- Checksum verification of incoming TCP and UPD datagrams (prior to V4R4)
- Outbound batching of TCP and UDP datagrams.

- Fragmentation of TCP and UDP datagrams into segments that match the MTU size.
- Starting with V4R2, AS/400 collects all TCP datagrams in one batch and UDP datagrams in a second batch. Ports and IP addresses are ignored. Releases prior to V4R2 batch together datagrams at the IOP when these conditions are true:
 - The protocol (TCP or UDP) matches
 - The source and destination ports match
 - The source IP address and destination IP address match
 - They arrive consecutively into the IOP

The IOP then passes the datagram batch to IP.

- Handling of IP and ICMP datagrams in error (unless IP NAT, which disables this function, is active)
- Resolving physical addresses using ARP protocol

These functions are called *TCP/IP-assist functions*. Whether these functions are done by the IOP or the System Licensed Internal Code (SLIC), depends on the IOP type, the OS/400 release, and the TCP/IP configuration. For details about specific functions, contact your local service representative. TCP/IP-assist functions are available on these IOPs:

- #2617 Ethernet/IEEE 802.3 adapter/HP
- #2619 16/4 Mbps Token-Ring Network adapter/HP
- #2618 Fiber distributed data interface adapter (FDDI)
- #2665 Shielded distributed data interface adapter (SDDI)
- #2666 High-speed communication adapter that is running frame relay only
- #2668 AS/400 wireless LAN adapter

Note: You can get the same function without using one of the above IOP adapters (done instead at a higher level in the system (SLIC)). When you use the X.25 protocol, you do not gain the advantage of the TCP/IP-assist function.

The *TCP/IP-assist functions* are also available on the following LAN IOAs and ATM IOAs:

- #2723 PCI Ethernet IOA
- #2724 PCI Token-Ring IOA
- #2838 PCI 100/10 Mbps Ethernet IOA
- #6149 16/4 Mbps Token-Ring IOA
- #2811 PCI 25 Mbps UTP ATM IOA
- #2812 PCI 45 Mbps Coax T3/DS3 ATM IOA
- #2813 PCI 155 Mbps MMF ATM IOA
- #2814 PCI 100 Mbps MMF ATM IOA
- #2815 PCI 155 Mbps UTP OC3 ATM IOA
- #2816 PCI 155 Mbps MMF ATM IOA
- #2818 PCI 155 Mbps SMF OC3 IOA
- #2819 PCI 34 Mbps Coax E3 ATM IOA

Note: If you configure your 100 Mbps ethernet line for TCPONLY, all IOP assist functions are disabled.

TCP/IP-assist functions that are available on frame relay IOAs are:

- #2699 Two-Line WAN IOA
- #2720 PCI WAN/Twinaxial IOA
- #2721 PCI Two-Line WAN IOA

Communications restrictions apply if any of the following communication functions are required when using the frame relay IOAs, as listed above:

- X.25, Frame Relay, or IPX Protocol
- SDLC protocol, if used to connect to more than 64 remote sites
- Communications line speeds greater than 64 Kbps and up to 2.048 Mbps for the synchronous data link control (SDLC) or frame relay protocols (bisync is always limited to a maximum of 64 Kbps)
- Communications line speeds greater than 64 Kbps and up to 640Kbps for X.25

Merge Host Table Performance

You can use the following data to help you plan for and anticipate performance when merging host tables. The data represents averages of measurements that are taken. The actual time required on your AS/400 system will be different.

Three cases were measured:

- Small merge—merge a 250-record file into the local host table that currently has 50 entries
- Medium merge—merge a 2000-record file into the local host table that currently has 50 entries
- Large merge—merge a 5000-record file into the local host table that currently has 50 entries.

The results of this test are shown in Table 37.

Table 37. Merge Host Table Performance

Number of records merged	Record format	Elapsed time (min:sec)	CPU percent
250	*AIX	0:42	43.7
2000	*NIC	5:38	49.4
5000	*NIC	13:54	48.6

This data equates to about 6 records per second and about .07-.08 processing unit seconds per record.

Running TCP/IP Only: Performance Considerations

Certain configurations of 2838 - 10/100 Mbps Ethernet cards allow you to run the IOP with only TCP/IP instead of all protocols for better performance. You need a 2838 Ethernet card with either:

- 2810 IOP
- 2809 IOP (the 2838 must be the only input/output adapter (IOA)IOA on the IOP)

If you have one of these configurations, you can use the TCPONLY parameter when you create or change your Ethernet line descriptions. Setting TCPONLY to *YES in other hardware configurations has no effect on the line.

Chapter 21. TCP/IP Problem Analysis

This chapter is intended to be used for determining solutions to problems encountered while using TCP/IP. This chapter contains:

- General information for determining problems regardless of what application you are using. You are given the following information:
 - A chart of potential problems.
 - A cause list for each potential problem in the chart.
- Information for determining problems with PING, SLIP, TELNET, FTP, SMTP, the POP server, the HTTP server, the workstation gateway server, LPR, and LPD. The debugging information for sockets is available in *Sockets Programming*, SC41-5422-03. The debugging information for the SNMP agent is available in *Simple Network Management Protocol (SNMP) Support*, SC41-5412-00. For TELNET, FTP, SMTP, the POP server, LPR, and LPD, you are given one or more of the following:
 - A chart of potential problems.
 - A cause list for each potential problem in the chart.
 - A description for tracing the TCP/IP application.
 - A description of the materials required for reporting TCP/IP problems.

The following steps should be followed when finding causes and solutions for TCP/IP problems:

1. Follow the flow chart in Figure 247 on page 431 to verify the connectivity between the local and remote systems.
2. Once you verify connectivity, determine which section of this document to go to based on what area the problem is occurring in. For example, if the problem is in SMTP, go to the section “Determining Problems for SMTP” on page 457.
3. Read the section associated with the function where the problem is occurring.
4. Use the flow chart provided to isolate the problem, if possible.
5. Find the cause list for that potential problem.
6. Follow the cause list instructions for correcting the problem.
7. Obtain the debugging materials required for reporting the problem if none of the actions in the cause list corrects the problem. This might include a trace of the function when the problem occurs. Instructions for collecting a communications trace are provided in “Collecting a Communications Trace” on page 493.

Note: All reported TCP/IP problems should include a copy of the configuration files used for TCP/IP processing.

To obtain a copy of the TCP/IP configuration files do the following:

1. If you have not created the library IBMLIB or output queue IBMOUTQ, enter the following commands:

```
CRTLIB LIB(IBMLIB)
CRTOUTQ OUTQ(IBMLIB/IBMOUTQ)
```

2. Enter the following commands to add the library IBMLIB to your library list and to change the output queue for your job to output queue IBMOUTQ:

```
ADDLIBLE IBMLIB
CHGJOB * OUTQ(IBMLIB/IBMOUTQ)
```

Enter the following commands to obtain a list of all physical files used for TCP/IP configuration:

```
WRKF FILE(QUSRSYS/QATOC*) FILEATR(PF)
WRKF FILE(QUSRSYS/QATM*) FILEATR(PF)
```

To copy the contents of each of the files, you can use option 3 (Copy from the work with files) or you can enter the following command on the command line for each listed file to copy the contents of each file to a separate spooled file in the IBMOUTQ output queue.

```
CPYF FROMFILE(QUSRSYS/QATOCHOST) TOFILE(*PRINT)
      FROMMBR(*ALL) TOMBR(*FROMMBR)
      MBROPT(*ADD) CRTFILE(*NO) OUTFMT(*HEX)
```

General TCP/IP Problems

This section discusses information that you need to verify regardless of which application you are using.

If you are unable to use your application, you should use the following flow chart to identify the cause. The cause lists that follow identify potential problems.

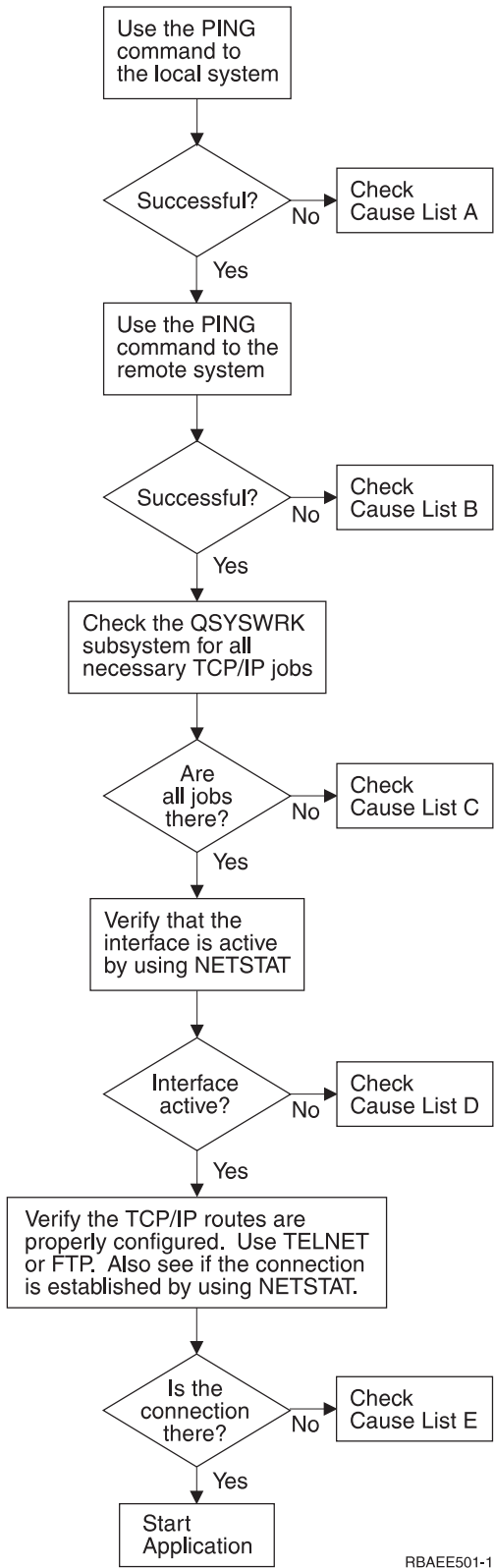


Figure 247. Initial TCP/IP Problem Analysis

Cause List A

1. Verify TCP/IP has been activated on your system.

Enter the STRTCP command to ensure that your TCP/IP stack is active. If it is active, you should receive message TCP1A04, TCP/IP currently active. If TCP/IP is not active, entering the STRTCP command will activate TCP/IP on your AS/400. Verify that no errors occurred while starting TCP/IP on your AS/400.

2. Verify your AS/400 TCP/IP software.

On the AS/400 system, the host name LOOPBACK and the interface with a line description value of *LOOPBACK, are reserved for verifying the AS/400 TCP/IP software. If you specify the LOOPBACK host name, no data is sent out on any of the physical lines. This allows you to quickly determine if TCP/IP software is working correctly on your AS/400 system.

To verify your TCP/IP software:

- a. Ensure the local host table has an entry for a LOOPBACK host name and internet address of 127.0.0.1.
- b. Ensure that the interface associated with the LOOPBACK host is active. The internet address usually associated with the LOOPBACK interface is 127.0.0.1. Ensure that there is an interface with the LOOPBACK host name's IP address configured with a line description of *LOOPBACK. Use the command:

```
NETSTAT OPTION(*IFC)
```

to view the LOOPBACK interface's status. If it is not active, use option 9 to activate it.

- c. After verifying that the LOOPBACK host's interface is active, type:

```
PING RMTSYS(LOOPBACK)
```

The loopback host allows the user to:

- Test FTP, TELNET, LPR or user-written application programs without being attached to a physical line or network.
- Verify that the TCP/IP software is installed and operating correctly.

Note: A similar test can be performed by using the PING command to verify connection with one of your other locally defined IP addresses.

- d. To test the software and hardware (adapter and network connection), specify the internet address of an external host on your network:

```
PING RMTSYS('nnn.nnn.nnn.nnn')
```

Depending on the line type of the local adapter, this command sends data out to the local adapter and is received again by the adapter and then responded to in the same way as the previous command.

Note: Ethernet does not send the data out on the Ethernet backbone if the internet address is for the local adapter.

- e. If you cannot successfully verify your system's connection to the network by specifying your system name or its internet address, check the source service access point (SSAP) of the line description associated with the interface. X'AA' must be specified as an entry in the SSAP (source service access point) list. This occurs by default when a new line description is created if the SSAP parameter is left at its default value of *SYSGEN. If you have an existing line description, use the Change Line Description command to add these values to the list.

Not all line description types must have a SSAP for TCP/IP so please check the source service access point (SSAP) list in the line description associated with the interface as described in Appendix A. Configuring a Physical Line for TCP/IP Communication.

- f. Verify all line description items, particularly the frame size which should be greater than the maximum transmission unit (MTU) of the interface.
- g. If a remote system fails to respond, it may mean that the system, network, gateway, router, or bridge in the network is unavailable or not working. Failure to respond can also mean that the remote system does not put into effect the ICMP echo request protocol. Try verifying the connection to other systems and between those other systems to determine where the failure is most likely located.
- h. Verify that the local interface configuration and the routing configuration, described in “Step 2—Configuring a TCP/IP Interface” on page 30 and “Step 3—Configuring TCP/IP Routes” on page 32, is correct.
- i. Ensure the following two routing entries are configured in the QSYSWRK subsystem description if the TCP/IP interfaces, including LOOPBACK, do not activate or you cannot end or start TCP/IP. If they do not exist, or if they are not correct, then add or correct them and try the request again.

```
ADDRTGE  SBSD(QSYS/QSYSWRK) +  
          SEQNBR(2505) +  
          CMPVAL(TCPIP) +  
          PGM(QSYS/QTOCTCPI) +  
          CLS(QSYS/QSYSCLS20) +  
          MAXACT(*NOMAX) +  
          POOLID(1)
```

```
ADDRTGE  SBSD(QSYS/QSYSWRK) +  
          SEQNBR(2506) +  
          CMPVAL(TCPEND) +  
          PGM(QSYS/QTOCETCT) +  
          CLS(QSYS/QSYSCLS20) +  
          MAXACT(*NOMAX) +  
          POOLID(1)
```

Cause List B

If your VFYTCPCNN or PING commands were successful to the local system, you should verify the possibility of connecting between your system and the other system you want to communicate with. Run the PING command as you did previously, but this time specify the internet address of the remote host. See “Common Error Messages” on page 438.

1. If you can verify the connection using the remote internet address but not the remote system name, then the name or address is not correct in your host table, or the remote name servers may not be available.
2. If your system uses remote name servers, verify that you can reach each remote name server by using the PING command and specifying the internet address of the remote name server.
3. When a remote system appears not to respond, it is possible that the system has responded but the packet was dropped due to a maximum frame size limitation. For example, if a TCP/IP communications line on the AS/400 system was configured with a maximum frame size of 576 bytes, an ICMP echo request sent on that line might preclude receiving an echo reply if the reply exceeded 576 bytes. It may appear to the AS/400 user that the remote host was not responding even though it had responded properly. Use the additional parameters of the PING command to decrease the packet length of the PING.

Almost all IP hosts assume that other IP hosts accept datagrams that are at least 576 bytes long (see RFC 791 for more information on length of datagrams). If any line used by AS/400 TCP/IP is configured to allow less than 576 bytes, any hosts that transmit to the AS/400 system using that line should be configured to send datagrams that are less than or equal to 576 bytes; otherwise, data may be lost.

4. There are additional parameters on the PING command that allow you to specify the packet length, the number of packets to be sent, and the wait time for a response. The default wait time of 1 second allows the remote system enough time to respond in most networks. However, if the remote system is far away or if the network is busy, increasing the wait time parameter can give a successful result.

It is recommended that the parameter values be left at their default values. Be aware that if you do change them, a combination of large packet length and short wait time may not give the network enough time to transmit and receive the response, and time-outs can occur. If the network is not given enough time to transmit and receive the response, it can appear that you do not have connectivity to a system when, in fact, you actually do.

5. If a remote system fails to respond, it may mean that the system, network, gateway, router, or bridge in the network is unavailable or not working. Failure to respond can also mean that the remote system does not put into effect the ICMP echo request protocol. Try verifying the connection to other systems and between those other systems to determine where the failure is most likely located.
6. If a remote system fails to respond when you are using the PING command to verify an interface, which is configured to a line description of Ethernet type, make sure the correct Ethernet standard or *ALL is specified in the Ethernet line description.
7. Failure to get responses from all systems in a network indicates the trouble is somewhere along the path. Verify the connection to the gateway leading into the network in question. If this fails, work back from the remote system you cannot reach until you find the point of failure.
8. Packets are sent using a low-level protocol that does not guarantee delivery. Because an echo request may be lost, do not assume that a network or gateway has failed until several commands fail to get beyond a point in the path.

Cause List C

1. Check the AS/400 QSYSWRK subsystem for all necessary jobs (local or remote). There should be at least the QTCPIP job. It is a control job. There should also be at least one job for each of the applications you are attempting to use as shown in Figure 248 on page 435. It is possible that these jobs may not be named identically to your subsystem jobs for the FTP, LPD, and TELNET jobs. All FTP jobs begin with QTFTP. All LPD jobs begin with QTLPD. All TELNET jobs will be named QTVTELNET and QTVDEVICE. It is possible to have more than one FTP, LPD, or TELNET server jobs. All SMTP jobs begin with QSMTP. SMTP has up to four jobs active in the QSYSWRK subsystem and two jobs active in the QSNADS subsystem. All SNMP jobs begin with QTMSNMP. SNMP can have three jobs active in the QSYSWRK subsystem, QTMSNMP, QTMSNMPRCV, and QSNMPSA.

Use the Work with Active Jobs (WRKACTJOB) command to display these jobs. Type WRKACTJOB SBS(QSYSWRK).

2. End TCP/IP processing using the ENDTCP OPTION(*IMMED) command if all the jobs are not there. Look for all the job logs associated with the jobs.
3. Change the job description message logging level for all the job description objects to 4 0 *SECLVL. See “Working with the Job Log and Message Queues” on page 438 for detailed information on the message logging levels.
4. Start TCP/IP processing again using the STRTCP command
5. Verify that all jobs are active.
6. Check the job logs if the appropriate jobs are not active.

```

Work with Active Jobs                SYSNAM03

                                02/03/99 18:06:32
CPU %:    .8    Elapsed time: 02:21:32    Active jobs: 93

Type options, press Enter.
  2=Change  3=Hold  4=End  5=Work with  6=Release  7=Display message
  8=Work with spooled files 13=Disconnect ...

Opt  Subsystem/Job  User      Type  CPU %  Function      Status
-----
   QSYSWRK        QSYS      SBS    .0     DEQW
   QMSF           QMSF      BCH    .0     DEQW
   QNEOSOEM       QUSER     ASJ    .0     PGM-QNEOSOEM  TIMW
   QNEOSOEM       QUSER     BCH    .0     PGM-QNEOSOEM  TIMW
   QNEOSOEM       QUSER     BCH    .0     PGM-QNEOSOEM  TIMW
   QNPSEVRD       QUSER     BCH    .0     SELW
   QPASVRP        QSYS      BCH    .0     PGM-QPASVRP   DEQW
   QPASVRS        QSYS      BCH    .0     PGM-QPASVRS   TIMW
   QPASVRS        QSYS      BCH    .0     PGM-QPASVRS   TIMW

Parameters or command
===>
F3=Exit  F5=Refresh  F7=Find  F10=Restart statistics
F11=Display elapsed data  F12=Cancel  F23=More options  F24=More keys
More...

```

Figure 248. Work with Active Jobs Display—Display 1

```

Work with Active Jobs                                SYSNAM03
02/03/99 18:06:32
CPU %:      .8   Elapsed time: 02:21:32   Active jobs: 93

Type options, press Enter.
 2=Change  3=Hold  4=End  5=Work with  6=Release  7=Display message
 8=Work with spooled files 13=Disconnect ...

Opt Subsystem/Job User      Type CPU % Function      Status
   QTLPD03516  QTCP      BCH   .0
   QTLPD03580  QTCP      BCH   .0
   QTMSNMP      QTCP      BCH   .0 PGM-QTOSMAIN  DEQW
   QTMSNMPCV   QTCP      BCH   .0 PGM-QTOSRCVR  TIMW
   QTVDEVICE   QTCP      BCH   .0 PGM-QTVDEVGM  TIMW
   QTVTELNET   QTCP      BCH   .0
   QZBSEVTM    QUSER     ASJ   .0 PGM-QZBSEVTM  EVTW
   QZHQRVD     QUSER     BCH   .0
   QZRCSRVD    QUSER     BCH   .0
                                           SELW
                                           More...

Parameters or command
====>
F3=Exit  F5=Refresh  F7=Find  F10=Restart statistics
F11=Display elapsed data  F12=Cancel  F23=More options  F24=More keys

```

Figure 249. Work with Active Jobs Display—Display 2

Cause List D

The network status (NETSTAT) function on the AS/400 system allows you to view the TCP/IP interface status, the TCP/IP route configuration information, and the TCP/IP connection status on your local system. You can use either the WRKTCPPSTS command or the NETSTAT command.

1. Start TCP/IP using the STRTCP command before using the network status function. The Work with TCP/IP Network Status menu is displayed but the options are not functional until TCP/IP has been started.

On the Work with TCP/IP interface status display, if you attempt to start an active interface or end an inactive interface, an appropriate error message is sent. If an inactive interface does not reach the active state after taking the start interface option, there may be a problem with the interface, the line, or the line configuration. See the job log of the QTCPIP job in the QSYSWRK subsystem to see what errors might have occurred when activating the interface. You can also look in the QSYSOPR and the QHST message queues to help determine the status.

Type WRKCFGSTS *LIN to determine if the line description has a problem. See Appendix A. Configuring a Physical Line for TCP/IP Communication for information on line descriptions.

2. Verify that at least one passive listening connection is shown for each of the servers on the Work with TCP/IP Connection Status display, option 3 from the Work with TCP/IP Network Status display.

SNMP

TELNET

Version 4 Release 4 supports SSL Telnet in addition to Telnet. SSL Telnet reflects a listening port of 992 by default and traditional Telnet uses port 23. Restricting Telnet listening ports is the recommended approach to disabling the traditional Telnet server, while allowing the SSL Telnet to be enabled.

FTP

SMTP, if configured
POP
WSG
LPD
REXEC
HTTP

Passive listening connections have an asterisk in the *Remote Address* and *Remote Port* fields. Ending these connections is not recommended. Remote systems cannot use SNMP, FTP or TELNET, send SMTP mail to the local system, or send spooled files using LPR to the local system if the associated passive listening connections have been ended. They can be restarted by ending and starting the servers using the ENDTCPVSR and STRTCPVSR commands and then specifying the server you want ended and started.

3. Ensure that the ports associated with the application you are attempting to use are not restricted. Use option 4 (Work with TCP/IP port restrictions) from the Configure TCP/IP menu to view the current port restrictions.

Cause List E

Verify the configuration data. If everything checks out, go to the flow chart for the particular application that you are using.

PING Command Considerations

The following sections discuss how the AS/400 system concatenates the domain name to the host name and examples are given for some of the most common PING error conditions.

Concatenating the Domain Name to the Host Name

This example illustrates how the AS/400 system concatenates the domain name to the host name if a period is not used at the end of the domain name.

Your AS/400 system name is SYSNAM01.A400SSC.DFW.COMPANY.COM, and you want to verify the connection to a system whose full name is SYSNAM02.DFW.COMPANY.COM. Your remote name server is configured to return the address for SYSNAM02.DFW.COMPANY.COM if you send the name server either the full name (SYSNAM02.DFW.COMPANY.COM) or just the short name (SYSNAM02). You do not have a SYSNAM02 host name in your local host table.

If you type:

```
PING SYSNAM02.DFW.COMPANY.COM
```

the AS/400 system sends SYSNAM02.DFW.COMPANY.COM to the remote name server. If you type:

```
PING SYSNAM02
```

the remote name server reports that the host is unknown. The reason that the remote name server does not recognize SYSNAM02 is because the AS/400 system sends the SYSNAM02.A400SSC.DFW.COMPANY.COM name to the remote name server for resolution. In other words, AS/400 TCP/IP concatenates the local domain name to the host name.

If you type:

```
PING SYSNAM02.
```

then just SYSNAM02 is sent to the remote name server, and the remote name server resolves the name, similar to when the full host name and domain name are specified. The only difference between this name and the previous name is the use of the period at the end of the name.

Common Error Messages

When you use the PING command to verify the connection to another host in the network, TCP/IP could give you an error message. The following are some of the most common error conditions:

- No TCP/IP service available

TCP/IP has not been started yet or has not completed starting. All jobs may not be started in the QSYSWRK subsystem.

Note: Use the NETSTAT command to see if TCP/IP is active. Use the Work with Active Jobs (WRKACTJOB) command to verify that the QSYSWRK subsystem and related jobs are active. If they are not active, look in the job log or system default output queue for any messages.

- Not able to establish connection with remote host system

Check your configured interfaces, their related line descriptions and the TCP/IP routes.

- Remote host did not respond to VFYTCPCNN within 10 seconds for connection verification 1.

Your configuration is probably correct, but you do not get an answer back from the remote system. Ensure that the remote host is able to reach your system. Call the remote system operator and ask them to verify the connection to your system.

Check the host tables or remote name server (if you are using a name server) for both systems, and the TCP/IP interfaces and routes. The remote name server may not be able to serve you for some reason.

If you are using an Ethernet line, make sure you specified the correct Ethernet standard or *ALL.

- VFYTCPCNN: Unknown host, xxxxxx where xxxxxx is the host name.

Check the local host table or the remote name servers (if you are using a name server) for the remote host's entry.

Working with the Job Log and Message Queues

TCP/IP is shipped with several job descriptions.

The job descriptions are stored in the QSYS or QTCP library. They are shipped with a message logging level of 4, a message logging severity of 0, and a message logging text value of *NOLIST. They are shipped with these values to prevent job logs from being created with only job started and job ended messages in them.

If you are having problems with the operation of TCP/IP, one of the first things to do is to change the message logging level on the job description for the application you are having problems with to a message logging text value of *SECLVL. Changing the message logging level generates a job log for that application. You

must end then restart the server for the change to take effect. If you want to change the job immediately, you must use the CHGJOB command to change the message logging level of the active job.

For example, if the problem is with the FTP server, change the QTMFTPS job description by typing the following CL command:

```
CHGJOB JOB(QTCP/QTMFTPS) LOG(4 0 *SECLVL)
```

If the problem is with SMTP, change the QTMSMTPS job description by typing the following CL command:

```
CHGJOB JOB(QTCP/QTMSMTPS) LOG(4 0 *SECLVL)
```

In addition to the QTMSMTPS job description, you might consider changing the logging level of the QSNADS subsystem job description:

```
CHGJOB JOB(QGPL/QSNADS) LOG(4 0 *SECLVL)
```

Determining Problems for SNMP

For information on debugging SNMP, see *Simple Network Management Protocol (SNMP) Support*.

Determining Problems for Serial Line Internet Protocol (SLIP)

To debug SLIP problems, use the following:

- The SLIP session job log
- The SLIP connection script spooled file
- A communications trace of the ASC line.

When a SLIP problem occurs, it is important to determine at what stage of SLIP processing the error occurred. The normal sequence of stages for establishing a SLIP connection are:

1. Point-to-Point configuration profile validation
2. SLIP session job startup
 - a. Validation of ASC line values
 - b. Possible creation of device and controller descriptions
 - c. Varying on of the ASC line, controller and device
3. Modem reset and initialization (unless null modem)
4. Call establishment and modem synchronization
5. Connection script dialog
6. TCP/IP interface and route activation
7. TCP/IP SLIP processing begins.

When a problem occurs, the three sources of debugging information listed previously should indicate at what point the connection failed, and why.

Problem: SLIP Connection Is Failing

If your SLIP connection is failing, there are several places to look for error indications:

1. Look in the job log of the user that issued the STRTCPPTP command for errors. Errors found during point-to-point configuration profile validation will be logged in this job log. Press F1 on the error message issued and follow the recovery section.
2. Look in the SLIP session job log. Use the Work with Point-to-Point TCP/IP (WRKTCPPPTP) command to display a list of profiles. Find the point-to-point profile, and select option 14. When you select option 14, the Work with Job (WRKJOB) display is shown. If the job is still active, you can select option 10 to display the current job log. If the job is no longer active, select option 4 to work with spooled files for this job. Display the QPJOBLOG spooled file and look for any error messages that have occurred. Refer to “Working With Point-to-Point Jobs” on page 136 for more information on working with point-to-point jobs and session job logs.
3. When the STRTCPPTP command is issued, the default for the OUTPUT parameter is *ERROR. When OUTPUT is set to *ERROR, a script dialog is printed if any errors occur during the modem initialization, call establishment, modem synchronization or connection script dialog phases. This spooled file contains information about any errors that have occurred during this process. To see if a spooled file was created, use option 14 of the WRKTCPPPTP command for the point-to-point profile that failed to complete.
 Select option 4 on the Work with Job display to show the Work with Job Spooled Files display. On this display look for the spooled file with the USER DATA field set to STRTCPPTP. Display the spooled file and look for any errors. Examples of typical errors include:
 - Rejected modem commands (examples are shown in Figure 250).
 - Connection rejected by remote system due to incorrect userid or password sent (examples are shown in Figure 251 on page 441).
 - Mismatches in connection scripts with missing data or out of sequence data (shown in Figure 252 on page 441). In Figure 252, the remote system prompted for the password but the connection script on the local system was not set up properly to look for this prompt. Therefore the remote system timed out and closed the connection.

Refer to “Working With Point-to-Point Jobs” on page 136 for more information on working with the connection script dialog spooled file.

```

1 Wed May 1 11:46:30 1996 ==> Attempting modem reset.
2 Wed May 1 11:46:30 1996 ==> AT&FS0=0
3 Wed May 1 11:46:30 1996 ==> Reading modem response.
4 Wed May 1 11:46:33 1996 ==> Reading modem response.
5 Wed May 1 11:46:33 1996 ==> AT&FS0=0 OK
6 Wed May 1 11:46:37 1996 ==> Attempting modem initialization.
7 Wed May 1 11:46:37 1996 ==> AT&D2&C1X4V1Q0S7=70W2\N3&K3&S1
8 Wed May 1 11:46:37 1996 ==> Reading modem response.
9 Wed May 1 11:46:40 1996 ==> Reading modem response.
10 Wed May 1 11:46:40 1996 ==> AT&D2&C1X4V1Q0S7=70W2\N3&K3&S1 ERROR

```

Figure 250. Rejected modem commands — Examples

```

16 Wed May 1 13:37:48 1996 ==> CONNECT 19200 LAPM COMPRESSED
17 Wed May 1 13:37:48 1996 ==>
18 Wed May 1 13:37:51 1996 ==> login:
19 Wed May 1 13:37:51 1996 ==> sliptest8
20 Wed May 1 13:37:56 1996 ==> password:
21 Wed May 1 13:37:56 1996 ==> *****
22 Wed May 1 13:38:00 1996 ==> Access not authorized for user SLIPTEST8
23 Wed May 1 13:38:01 1996 ==> Remote host connection ended.

```

Figure 251. Connection rejection — Examples

```

16 Wed May 1 14:00:22 1996 ==> CONNECT 19200 LAPM COMPRESSED
17 Wed May 1 14:00:22 1996 ==>
18 Wed May 1 14:00:26 1996 ==> login:
19 Wed May 1 14:00:26 1996 ==> sliptest8
20 Wed May 1 14:00:31 1996 ==> password:
21 Wed May 1 14:02:36 1996 ==> Remote host connection ended.

```

Figure 252. Mismatches in Connection Scripts — Examples

4. If both of the following are true:
 - A problem occurs with your communications to the modem, or with communications to or from the remote system (including connection script problems)
 - The spooled file(s) for the point-to-point job does not contain enough information to isolate the error

you may need to run a communications trace on the ASC line while attempting the SLIP connection. When printing the trace data, be sure to specify *CALC for the format of the data you are printing to allow the system to format both EBCDIC and ASCII data properly. The communications trace could contain data to help isolate the error that is occurring.

If you are not familiar with the procedure for collecting a communications trace, refer to “Collecting a Communications Trace” on page 493 and “Formatting and Saving the Communications Trace” on page 499.

5. If you are trying to connect to an Internet Service Provider (ISP) but are unable to complete the connection, verify that no errors occurred up to the point the call is made to the ISP. You can verify this through the Connection Dialog script spooled output. You should see an entry for ‘Attempting modem dial/answer’ followed by an ‘ATDT’ (tone) or ‘ATDP’ (pulse) entry that contains the number you are dialing.

If no errors occurred up to point when you dial your ISP, you may need to contact your ISP to verify the following:

- If you do not receive a ‘CONNECT’ indication after the call is made, verify that the telephone number you are using for the ISP is correct.
- If you need to dial another number to get to an outside line prior to dialing the ISP, make sure that you use enough comma (“,”) special characters to allow enough time for the outside line to become available. For example: ‘9,,,,7771234’ can be defined for the *Remote service phone number* in the ‘Remote system access information’ section of your point-to-point configuration profile, where “9” is the number required to get an outside line. The time delay for each comma is defined by your modem.

- If your user ID or password information is being rejected, verify that the correct user ID and password are configured in your point-to-point configuration profile and that this information is correct for your ISP account.
- Verify that the connection script you are using matches what the ISP is expecting. The connection dialog script spooled file output and a communications trace of the failed connection attempt may help with this verification.

Problem: SLIP Job 'Hung' with STRSSN Status

If you start a point-to-point profile and the status on the WRKTCPPPTP display for the profile appears to be hung at a STRSSN status, this could be an indication that an inquiry message has been sent to the system operator (QSYSOPR message queue) and the job is waiting for a reply. Go to a command line and enter the following command:

```
DSPMSG MSGQ(QSYSOPR)
```

Look for inquiry messages involving the ASC line in use. If you find messages, look for information on the problem that is occurring. Typical problems include:

- The modem is powered off
- The modem cable is not connected properly.

Before continuing, either answer the inquiry message or end the SLIP session by issuing the ENDTCPPTP command. After ending the session, correct the problem and try to start the session again.

Problem: SLIP Connection Complete but Unable to PING

If you **cannot PING REMOTE IP address:**

- Try setting the WAITTIME parameter on the PING command to a value greater than the one second default.

Dial-up connections are much slower than direct connections through LANs and some other types of connections. The default wait time for the PING command is one second, which may be too short a time period for the PING to complete over the dial-up line.

If you **cannot PING LOCAL IP address:**

- If you can PING the remote IP address (showing connectivity to the remote system) but cannot PING the LOCAL IP address, then it is unlikely that there is a problem. It depends on how the remote system is configured. If the remote system is capable of forwarding IP datagrams and has its routing set up correctly to send the IP datagrams back to your local address, the PING should work. But, if the remote system cannot forward IP datagrams or its routing is not set up to send the IP datagrams back to your local address, the PING request will not work. This is because of the way PING works on point-to-point links.
- Refer to "Allow IP Datagram Forwarding" on page 145 for more information on why the remote system must forward IP datagrams to be able to PING your local IP address.

Materials Required for Reporting SLIP Problems

Include the following with any SLIP problem reported to IBM:

1. The type of remote host, operating system, and operating system level.

2. The failing point-to-point session job log.
3. The type and model of the modem used and the modem strings selected.
4. A copy of the options used in the failing point-to-point profile.
5. A copy of the parameter values used in the line description.
6. The connection dialog script being used (if one is being used)
7. The connection dialog script spooled file output (if it exists)
8. A copy of any QSYSOPR messages that may have been logged concerning the SLIP connection
9. A communications trace of the failing SLIP connection (if possible).

Determining Problems with TELNET

If a problem is detected when using the AS/400 TELNET server, use the following flow chart to identify the cause after using the flow chart for general TCP/IP problems. (Figure 247 on page 431). The cause lists that follow identify potential problems.

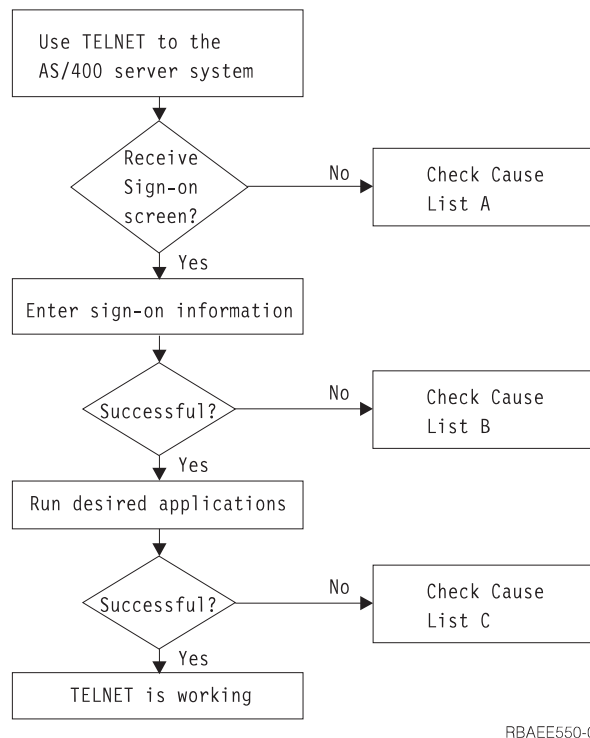


Figure 253. TELNET Server Problem Analysis

Cause List A

1. Verify that the Telnet server jobs are active and that Telnet service is assigned to a valid non-restricted port. Use the WRKACTJOB command to verify that QTVTELNET and QTVDEVICE jobs are active in QSYSWRK subsystem. If these jobs are not active, use the STRTCPSVR *TELNET command to start these jobs. Use NETSTAT *CNN to verify that Telnet service is assigned to a valid port.
2. Verify that the QAUTOVRT system value on the AS/400 server system is properly set to allow the TELNET server to automatically create virtual devices. For example, to allow the creation of 50 virtual devices enter the command:

CHGSYSVAL SYSVAL(QAUTOVRT) VALUE(50)

Note: Since Version 4 Release 2, QAUTOVRT supports numeric values of 0 through 32500, and a special value of *NOMAX.

3. Verify that the network connection between the AS/400 server and the Telnet client is active by using an application such as PING(VFYTCPCNN). If not active, see your network administrator.
4. Verify that the virtual devices on the AS/400 server system that are used by TELNET are defined to a subsystem under which the interactive TELNET jobs should run. Use the Display Subsystem Description (DSPSBSD) command to see which work station entries are defined to a subsystem. Use the Add Work Station Entry (ADDWSE) command to define work stations to a subsystem. For example, you could use the following command to allow all work station types to run under the QINTER subsystem:
ADDWSE SBSDB(QINTER) WRKSTNTYPE(*ALL)
5. Verify that the interactive subsystem (QINTER) is Active. TELNET connections fail if the interactive subsystem is not active. In this situation, the system does not write error messages to the QTVTELNET job log to show you the problem. Use the DSPSBSD SBSDB (subsystem name) command to verify that the subsystem is active. Messages are not written to either QTVTELNET or QTVDEVICE job log.
6. Verify that your local VTxxx client is configured to automatically wrap lines at column 80 if you are operating in VTxxx full-screen mode.
7. Check for a TELNET user exit program registered to exit point QIBM_QTG_DEVINIT format INIT0100, using the WRKREGINF CL command. If there is a registered user exit program, check the TELNET server job log for any errors related to that program. If errors exist, correct the errors in the exit program or remove the exit program with the RMVEXITPGM CL command.
8. Ensure that your client is attempting to use the correct port to connect to Telnet. Use CFGTCP or NETSTAT *CNN to determine what port Telnet service is assigned to.
9. Use the CFGTCP command to verify that the port your client is attempting to connect on is not restricted. Also look in the QTVTELNET job log for messages indicating that the port that you are trying to use is restricted.
10. If you are attempting to connect using SSL Telnet, then in addition to the above items, ensure that the Digital Certificate Manager (DCM), and one of the IBM Cryptographic Provider products, are installed. Also, ensure that a valid, non-expired certificate is assigned to the Telnet server (QIBM_QTV_TELNET_SERVER).

Cause List B

1. Verify your authority to the virtual display device. If you receive message CPF1110 when attempting to sign on the AS/400 system, you are not authorized to the virtual display device. When the AS/400 TELNET server creates virtual devices, the QCRTAUT system value is used to determine the authority granted to user *PUBLIC. This system value should be *CHANGE to allow any user to sign on using TELNET.
2. Verify that the QLMTSECOFR system value is correctly set if you are the security officer or have *SECOFR authority.

Cause List C

1. Verify your word processing choice. If you experience problems when using OfficeVision or the Work with Folders (WRKFLR) command, you may need to

change your configuration so that the Office Adapted Editor is used instead of the Standard Editor. To do this, have your system administrator change your word processing choice in your Environment Information associated with your office user ID.

2. Verify that your local VTxxx client is configured to automatically wrap lines at column 80 if you are operating in VTxxx full-screen mode.
3. If characters are not displayed properly for your VTxxx session, verify that the correct mapping tables are in use for your session.
4. If your VTxxx client beeps every time you press a key, your keyboard may be locked. See “Error Conditions on 5250 Keyboard” on page 207 for more information.
5. Check the QTVTELNET job log and the QTVDEVICE job log for error messages on the AS/400 server system.

Materials Needed when Reporting TELNET Problems

Problems reported to IBM may include one or more of the following as determined by your service representative:

- Telnet Server job logs:
 - QTVTELNET job log(s)
 - QTVDEVICE job log(s)
- Some details on the problem scenario. For example:
 - The type of remote host you were using to Telnet from or to, such as AS/400, S/390, or RS6000. This is particularly useful if you are doing cascaded Telnet scenarios.
 - The type of client attempting to connect to the Telnet server, such as IBM Personal Communications, and AS/400 Client Access.
 - Job log of interactive job running Telnet client (when Telnet client is under investigation).
 - TRCJOB of the failing interactive job (especially important if running Telnet client).

Note: Use TRCJOB *ON to start this trace. The result will be a QPSRVTRC spool file in the interactive job.

- A communications trace of the failure, which contains TCP/IP data only, and is formatted for both ASCII and EBCDIC. If you are not familiar with the procedure for collecting a communications trace, refer to “Collecting a Communications Trace” on page 493 and “Formatting and Saving the Communications Trace” on page 499. You may be directed by your service representative to include broadcast messages in this trace. In addition, you may need to filter this trace on a specific IP address if you have a large amount of traffic on your network, and know the IP address of the failing client.
- Any VLIC logs with major code 0700 and minor code 005x from the time of failure. In addition, there may be some major code 0701 and minor code 005x informational VLIC logs that may be useful but not necessarily critical.
- A Virtual Terminal Manager (VTM) VLIC component trace. This trace can be gathered via the TRCTCPAPP command or via STRSST. For full details on using the TRCTCPAPP command, see the *OS/400 CL Reference* located in the AS/400 Online Library at the following URL address:
<http://www.as400.ibm.com/infocenter>.

It is important to note that running the VTM VLIC trace will have performance impacts. Some examples of using this command are:

- To trace all VTM activity:
TRCTCPAPP APP(*TELNET) SET(*ON)
- To trace the activity on a specific device, when you know the device name:
TRCTCPAPP APP(*TELNET) SET(*ON) DEVD(devicename)
- To trace the activity on a specific device, when you know the IP address of the client:
TRCTCPAPP APP(*TELNET) SET(*ON) RMTNETADR(*INET'www.xxx.yyy.zzz')
- To turn the trace off and spool output:
TRCTCPAPP APP(*TELNET) SET(*OFF)

Note: Specific details of which trace parameters to use for your problem should be received from your service representative prior to running this command. This will ensure that the correct information is gathered for your problem.

TRCTCPAPP Service Program Outputs

For TRCTCPAPP, the listing of the VTM component trace will show up as a spooled file called, VTTRACE, with the user data field set to TELNET. This will be placed into the default output queue of the profile executing the TRCTCPAPP *TELNET *OFF call. At the same time, all server job flight recorders - **rs** - will be dumped to spooled files called QTOCTTRC with user data set to QTVnnnnnn.

Here is an example of what you can expect to see in your interactive job log when you perform a TRCTCPAPP *OFF call:

```
Command Entry                                SYSNAM03
                                           Request level: 1
All previous commands and messages:
> trctcpapp *telnet *off
Spooled printer file 1 opened for output.
Trace data for application TELNET formatted: Spooled VTTRACE user data 'TELNET'.
Trace data for application TELNET formatted: Spooled QTOCTTRC user data 'TV017231'.
Trace data for application TELNET formatted: Spooled QTOCTTRC user data 'TV017230'.
Trace data for application TELNET formatted: Spooled QTOCTTRC user data 'TV017229'.
Trace data for application TELNET formatted: Spooled QTOCTTRC user data 'TV017232'.
Trace data for application TELNET formatted: Spooled QTOCTTRC user data 'TV017233'.
Trace data for application TELNET formatted: Spooled QTOCTTRC user data 'TV017234'.
                                           More...
Type command, press Enter.
====>
F3=Exit   F4=Prompt   F9=Retrieve   F10=Exclude detailed messages
F11=Display full   F12=Cancel   F13=Information Assistant   F24=More keys
```

Here is an example of what you can expect to see in your default output queue:

```

Work with All Spooled Files

Type options, press Enter.
 1=Send  2=Change  3=Hold  4=Delete  5=Display  6=Release  7=Messages
 8=Attributes  9=Work with printing status

Opt  File      User      Device or      User Data      Sts  Total  Cur  Copy
     File      User      Queue          Queue          HLD  Pages Page Copy
VTMTRACE  JEFF      JEFFSOUTQ  TELNET         HLD   46     1     1
QTOCTTRC  JEFF      JEFFSOUTQ  TV017231      HLD    4     1     1
QTOCTTRC  JEFF      JEFFSOUTQ  TV017231      HLD    2     1     1
QTOCTTRC  JEFF      JEFFSOUTQ  TV017231      HLD    2     1     1
QTOCTTRC  JEFF      JEFFSOUTQ  TV017231      HLD    2     1     1
QTOCTTRC  JEFF      JEFFSOUTQ  TV017231      HLD    2     1     1

Parameters for options 1, 2, 3 or command
===>
F3=Exit  F10=View 4  F11=View 2  F12=Cancel  F22=Printers  F24=More keys

Bottom

```

|
|
|
|
|

There is only one file called VTMTRACE that will be created. There can be one or more QTOCTTRC files, if SSL Telnet mode is operational on the server.

Here is an example of one of the QTOCTTRC files. This spooled file is a Telnet server job (QTVTELNET) as opposed to a QTVDEVICE job:

```

                                Display Spooled File
File . . . . . : TV017231                               Page/Line  1/6
Control . . . . .                               Columns  1 - 78
Find . . . . .
*...+...1...+...2...+...3...+...4...+...5...+...6...+...7...+...
5769TC1 V4R4M0 990521 TRCTCPAPP Output SysName Date-12/11/98 Time-14:08:32 Page-
TRCTCPAPP Attributes
  Application.....: Telnet Server
  Buffer size (KB).....: 0
    (Default of 0 means 16MB buffer)
  Trace full action.....: *WRAP
  Job id.....: 017231/QTCP      /QTVTELNET
  Start date/time.....: Fri Dec 11 13:50:33 1998
  End date/time.....: Fri Dec 11 14:08:34 1998
  Trace buffer wrapped.....: No
Telnet Server Attributes
  AutoStart server.....: 'Y'
  Number servers.....: 2
  Session keep alive timeout..: 0
  Default NVT type.....: >*VT100<
  Outgoing EBCDIC/ASCII table.: >*CCSID      <
  Incoming ASCII/EBCDIC table.: >*CCSID      <
  Coded character set id.....: 84542
  Attributes version id.....: >V4R4M0      <
Trace common buffer structure:
  80000000 00000000 161A8753 14001074 | .....g..... | Byte 16
  80000000 00000000 161A8753 14FFFE4   | .....g...U  | Byte 48
  80000000 00000000 161A8753 14005820 | .....g..... | Byte 80
  00FFF000 00000084 F0F1F7F2 F3F1D8E3 | ..0...d017231QT | Byte 112
  C3D74040 40404040 D8E3E5E3 C5D3D5C5 | CP      QTVTELNE | Byte 144
  E340C699 8940C485 8340F1F1 40F1F37A | T Fri Dec 11 13: | Byte 176
  F5F07AF3 F340F1F9 F9F8D8E3 E5F0F1F7 | 50:33 1998QTV017 | Byte 208
  F2F3F140 | 231 | Byte 228
Flight Records:
qtvtnet: Job: QTVTELNET/QTCP/017231
(C) Copyright IBM Corporation, 1999
Licensed Material - Program Property of IBM.
Refer to Copyright Instructions Form No. G120-2083
ProdId: 5769-SS1 Rel: V4R4M0 Vers: V4R4M0 PTR: P3684767
qtvtnet: Program QTVTELNET dated 04 December 1998 running
qtvtnet: Source file: qtvtnet.p1C
qtvtnet: Last modified: Wed Dec 9 11:57:40 1998
qtvtnet: Last compiled at 12:00:10 on Dec 9 1998
qtvtnet: Arguments passed: 1
qtvtnet: Time Started: Fri Dec 11 13:50:34 1998
qtvtnet: sigaction() for SIGUSR1 is EndClientSession()
qtvtnet: Set Telnet Server job identity for OpNav
qtvtnet: Need to setup SSL_Init_Application()
qtvtnet: SSL_Init_Application() successful
qtvtnet: Find Telnet Server control block
qtvtnet: Lock Telnet Server control block
qtvtnet: Open driver to stream
qtvtnet: First Telnet Server Job...

More...

F3=Exit  F12=Cancel  F19=Left  F20=Right  F24=More keys

```

Here is an example of another QTOCTTRC file. This is a device manager spooled file, as opposed to the QTVTELNET server job:

```

Display Spooled File
File . . . . . : TV017230          Page/Line  1/6
Control . . . . .           Columns   1 - 78
Find . . . . .
*...+...1...+...2...+...3...+...4...+...5...+...6...+...7...+...
TRCTCPAPP Attributes
  Application.....: Telnet Server
  Buffer size (KB).....: 0
    (Default of 0 means 16MB buffer)
  Trace full action.....: *WRAP
  Job id.....: 017230/QTCP      /QTVDEVICE
  Start date/time.....: Fri Dec 11 13:50:33 1998
  End date/time.....: Fri Dec 11 14:08:39 1998
  Trace buffer wrapped.....: No
Telnet Server Attributes
  AutoStart server.....: 'Y'
  Number servers.....: 2
  Session keep alive timeout..: 0
  Default NVT type.....: >*VT100<
  Outgoing EBCDIC/ASCII table.: >*CCSID      <
5769TC1 V4R4M0 990521 TRCTCPAPP Output SysName Date-12/11/98 Time-14:08:32 Page-
*...+...1...+...2...+...3...+...4...+...5...+...6...+...7...
  Incoming ASCII/EBCDIC table.: >*CCSID      <
  Coded character set id.....: 84542
  Attributes version id.....: >V4R4M0      <
Trace_common buffer structure:
  80000000 00000000 3DA86C25 5F001074 |.....y...| Byte 16
  80000000 00000000 3DA86C25 5FFFFFFE4 |.....y..U| Byte 48
  80000000 00000000 3DA86C25 5F002F64 |.....y...| Byte 80
  00FF0000 00000084 F0F1F7F2 F3F0D8E3 |..0....d017230QT| Byte 112
  C3D74040 40404040 D8E3E5C4 C5E5C9C3 |CP      QTVDEVIC| Byte 144
  C540C699 8940C485 8340F1F1 40F1F37A |E Fri Dec 11 13: | Byte 176
  F5F07AF3 F340F1F9 F9F8D8E3 E5F0F1F7 |50:33 1998QTV017| Byte 208
  F2F3F040 |230          | Byte 228
Flight Records:
qtvtncsh: >>>> entry
(C) Copyright IBM Corporation, 1999.
Licensed Material - Program Property of IBM.
Refer to Copyright Instructions Form No. G120-2083
ProdId: 5769-SS1 Release: V4R4M0 Version: V4R4M0 PTR: P3684767
qtvtncsh: Program QTVTNCSH dated 04 December 1998 running
qtvtncsh: iActiveLogLevel: 0
qtvtncsh: Source file: qtvtncsh.c
qtvtncsh: Last modified: Wed Dec 9 11:48:33 1998
qtvtncsh: Last compiled at 11:59:42 on Dec 9 1998
qtvtncsh: SignalHandler() registered with signal()
qtvtncsh: Arguments passed: 4
qtvtncsh: argc: 4
qtvtncsh: argv[0]: >QSYS/QTVTNCSH<
qtvtncsh: argv[1]: ><
qtvtncsh: argv[2]: >1p<
qtvtncsh: argv[3]: >s<
SignalHandler: >>>> entry
SignalHandler: Caught signal SIGSEGV
More...
F3=Exit  F12=Cancel  F19=Left  F20=Right  F24=More keys

```

Automatically Generated Diagnostic Information (FFDC Errors)

When certain errors occur within the Telnet server, some automatically generated diagnostic information is also generated. There will be times when your service representative will require this diagnostic information to properly analyze a Telnet server problem.

If any Telnet or device manager job fails with an FFDC error, you will see the spooled files under the WRKSPLF QTCP profile. When a job fails with an FFDC error, each failing job will automatically have two dumps. One is a dump made by calling DSPJOB *PRINT, and the other is made by DSPJOBLOG *PRINT. This way,

you get both the job log and job run attributes dumped and have the user data group output together with a job number identifier. This will let you match up with any VTM Component Trace output.

You will see a total of four spooled files; two for the QTVTELNET job and two for the QTVDEVICE job. These spooled files are automatically generated when an FFDC error is encountered. For an example, see Figure 254:

Work with All Spooled Files

Type options, press Enter.
 1=Send 2=Change 3=Hold 4=Delete 5=Display 6=Release 7=Messages
 8=Attributes 9=Work with printing status

Opt	File	User	Device or Queue	User Data	Sts	Total Pages	Cur Page	Copy
	QPJOBLOG	QTCP	QEZJOBLOG	TV016868	HLD	4		1
	QPDSPJOB	QTCP	QPRINT	TV016868	HLD	7		1
	QPJOBLOG	QTCP	QEZJOBLOG	TV016955	HLD	3		1
	QPDSPJOB	QTCP	QPRINT	TV016955	HLD	7		1
	QPJOBLOG	QTCP	QEZJOBLOG	TV017231	HLD	3		1
	QPJOBLOG	QTCP	QEZJOBLOG	TV017232	HLD	3		1
	QPDSPJOB	QTCP	QPRINT	TV017232	HLD	7		1
	QPDSPJOB	QTCP	QPRINT	TV017231	HLD	7		1

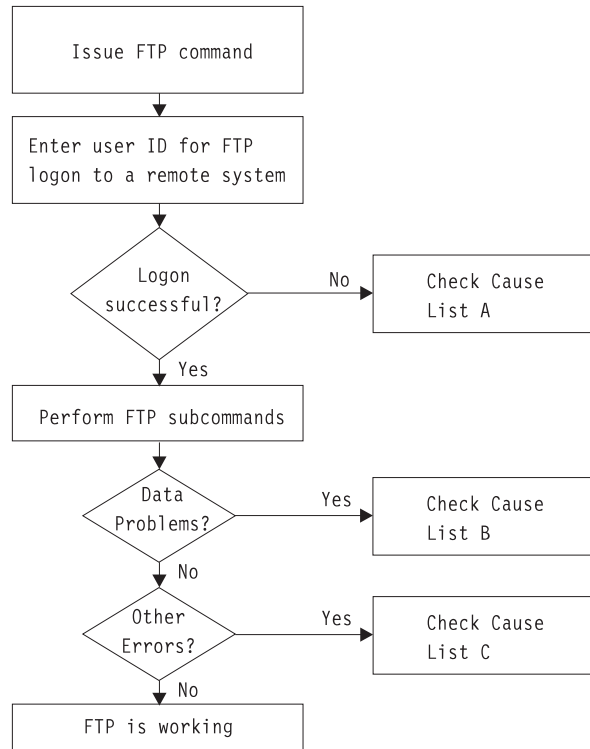
Bottom

Parameters for options 1, 2, 3 or command
 ==>
 F3=Exit F10=View 4 F11=View 2 F12=Cancel F22=Printers F24=More keys

Figure 254. Work with All Spooled Files Display

Determining Problems with FTP

If you detect a problem when using FTP, use the following flow chart to identify the cause after using the flow chart for general TCP/IP problems. (Figure 247 on page 431). The cause lists that follow list steps to help you identify the cause of the problem.



RBAAE551-0

Figure 255. FTP Problem Analysis

Cause List A

1. Is there is a long delay between connecting to the AS/400 FTP server and receiving a prompt for a user id? If so, check the configuration of the domain name server on your AS/400. The FTP server performs a DNS query as soon as a new connection is received. DNS problems may cause the server to hang for several minutes before a response is received.
2. Check to see if an exit program has been added to the FTP Server Logon Exit Point. If yes, then check if the logon that is unsuccessful is allowed by the exit program.
3. Check to see if the remote logon requires a password if a password was requested. Some systems request a password, but the connection can fail because it is not required.
4. Set up a password on the remote system if required. You may have to IPL if you change the security information on the system.
5. Check your user ID and password by attempting to sign on to your remote system. If you are unable to do so, contact the system owner to verify that your user ID and password are correct.

Cause List B

1. Make sure binary mode is in effect if you are transferring binary files.
2. Check to be sure the mapping tables on both the client and server systems are compatible. You need only do this if you are using your own mapping tables.
3. Check to see that the correct CCSID has been specified for the transfer. If not, use the TYPE or LTYPE subcommand to set the correct CCSID value before the transfer is performed.

4. Create a file on the system that you are planning to store data into. Set the proper record length, number of members, and number of increments. Try the data transfer again and verify that it was successful.
5. Make sure that you are authorized to use the file and the file members.
6. Check to see if the transfer file contains packed decimal or zoned decimal data. If yes, then make sure that the transfer is done as described in "Transferring Files that Contain Packed Decimal Data".
7. If you are transferring a Save file, verify that the appropriate method was used as described in "Transferring Save Files".

Cause List C

1. Check file size limits on the remote system.
2. Check to see if the FTP server timer ended. The AS/400 server time-out value can be set using the QUOTE TIME command.
3. Use the NETSTAT command to verify that the *LOOPBACK interface is active. Then re-create the problem doing FTP LOOPBACK (AS/400-to-AS/400 internally).
 - If the problem cannot be recreated, it is probably a remote system problem.
 - If you can re-create the problem, do the following:
 - a. If the problem is an FTP server problem, then start the FTP server trace using the TRCTCPAPP command.
 - b. Create the problem again.
 - c. End the FTP connection.
 - d. End the FTP server trace using the TRCTCPAPP command.
 - e. Find a spooled file with a file name of QTMFFTRC and the user name set to the name of the user issuing the TRCTCPAPP command which stopped and formatted the trace. The trace is a spooled file in the default output queue of the system associated with the FTP server job.
 - f. Send in that spooled file.
 - g. If the problem was on the AS/400 FTP client, a trace can be obtained using the DEBUG 100 client subcommand.
 - h. When running the FTP client interactively, use the F6 (Print) key to create a spool file that contains a history of the FTP client subcommands entered, and the associated FTP server replies. When the FTP client is run in batch unattended mode, then this history of subcommands and server replies is written to the specified OUTPUT file. For more details, see "FTP as Batch Job".

Materials Required for Reporting FTP Problems

Any FTP problem reported to IBM should include the following:

- A communications trace from the time of the failure (Request TCP/IP data only) formatted twice: once for ASCII and once for EBCDIC. If you are not familiar with the procedure for collecting a communications trace, refer to "Collecting a Communications Trace" on page 493 and "Formatting and Saving the Communications Trace" on page 499.
- If the FTP client or server has logged software error data, submit the data.

Note: The system value QSFWERRLOG must be set to *LOG for software error logging to take place. If an error occurs while QSFWERRLOG is set to *NOLOG, change the value to *LOG, try to re-create the error, and submit

the logged software error data. If logged software error data is submitted, there is no need to perform a trace of FTP.

- The QTCPIP and any FTP server or FTP client job logs.
- The FTP client and FTP server debug traces.
- For FTP client problems, a spool file containing the FTP client session (which may be obtained by hitting the print (F6) key in the FTP session).
- If data integrity is the problem, then the file, member, or library causing the problem should be sent in along with a copy of the description of the file, member, or library.

Tracing FTP Server

The FTP server can be traced from any AS/400 or non-AS/400 system that runs TCP/IP. With V4R4, there are now two ways to trace the FTP server. The FTP server DBUG subcommand traces *within* an FTP server session. The new Trace TCP/IP Application (TRCTCPAPP) command allows *system wide* tracing of *all* the FTP servers.

Tracing the FTP server with the DBUG subcommand

The following is an example using the FTP server DBUG subcommand:

```
File Transfer Protocol

Previous FTP subcommands and messages:
Connecting to host name xxxxn timer.xxxxxxxxx.xxx.xxx at address
n.nnn.nn.nnn using port 21.
220-QTCP at xxxxn timer.nnnnnnnn.nnn.nnn.
220 Connection will close if idle more than 5 minutes.
215 OS/400 is the remote operating system. The TCP/IP version is
"V4R4M0".
>
331 Enter password.
230 TEST logged on.
250 Now using naming format "0".
257 "QGPL" is current library.

Enter an FTP subcommand.
====> quote debug

F3=Exit      F6=Print      F9=Retrieve
F17=Top      F18=Bottom   F21=CL command line
```

Figure 256. Using the FTP DBUG Subcommand

To trace the FTP server:

1. Type QUOTE DBUG to start the trace.

```
File Transfer Protocol
```

```

Previous FTP subcommands and messages:
Connecting to host name xxxxxnnn.xxxxxxxx.xxx.xxx at address
n.nnn.nn.nnn using port 21.
220-QTCP at xxxxxnnn.xxxxxxxx.xxx.xxx.
220 Connection will close if idle more than 5 minutes.
215 OS/400 is the remote operating system. The TCP/IP version is
"V4R4M0".
>
331 Enter password.
230 TEST logged on.
250 Now using naming format "0".
257 "QGPL" is current library.
> quote debug
250 Debug mode is now ON.
Enter an FTP subcommand.
==> quote debug

F3=Exit      F6=Print      F9=Retrieve
F17=Top      F18=Bottom   F21=CL command line

```

Figure 257. Using the FTP Subcommand QUOTE DEBUG

2. Perform the FTP operation that you want to trace.
3. Type QUOTE DEBUG again to end the trace. The trace creates a spooled file called QTMFFTRC. The default output queue contains the spooled file. The user is always the name of the user who was logged on to the FTP server when the trace was ended.
4. Type QUIT to end the FTP session.
5. Enter the following command to find the output queue:
DSPSYSVAL QPRTDEV
For example, the following display appears:

```
Display System Value
```

```

System value . . . . . : QPRTDEV
Description . . . . . : Printer device description
Printer device . . . . . : PRT01      Name

```

Figure 258. Display System Value Display

6. Record the name of the printer device. In this example, PRT01 is the printer device.
7. Press F12 (Cancel) to return to the display where you entered the DSPSYSVAL command.
8. Type the following command:
WRKOUTQ OUTQ(printer-device)
Replace printer-device with the printer device recorded in the previous display. PRT01 is the output queue in this example. For example, the following display appears:

```

Work with Output Queue
Queue: PRT01      Library: QGPL      Status: RLS
Type options, press Enter.
  1=Send  2=Change  3=Hold  4=Delete  5=Display  6=Release  7=Messages
  8=Attributes  9=Work with printing status
Opt  File      User      User Data  Sts  Pages  Copies  Form Type  Pty
-   QTCPPRT   QTCP      QTSMTP    HLD   46    1      *STD      5
-   QTMFFTRC  QSECOFR   HLD      44    1      *STD      5

```

Figure 259. Work with Output Queue Display

9. Press F18 (Bottom) to get to the bottom of the spooled file list if More... appears on the display.
10. Find the last file named QTMFFTRC with the same user as the user who was logged on the FTP server when the trace was created.
11. Press F11 (View 2) to view the date and time of the file you want to work with.
12. Verify that you are working with the most recent spooled file, QTMFFTRC.
13. Indicate in the problem report that the trace was tried and it failed. Send whatever trace information there is with the problem report.

Tracing the FTP server with the Trace TCP/IP Application (TRCTCPAPP) command

The Trace TCP/IP Application (TRCTCPAPP) command (new for V4R4) allows *system wide* tracing of *all* the FTP servers.

The TRCTCPAPP command is provided specifically for trained service and development personnel. *SERVICE special authority is required to use this command. Use TRCTCPAPP in situations that require the capturing of trace data for service and development use. This command allows experienced personnel to dynamically start and stop tracing for applications.

With the use of TRCTCPAPP, trace information can be captured for the FTP TCP/IP application:

- Internal trace information can be captured for the AS/400 FTP server. The information that can be captured for the FTP server may be filtered using remote IP address and port or AS/400 user profile. Only one trace can be active at a time on the system.

Here are two examples of the use of the TRCTCPAPP command:

Example 1:

```
TRCTCPAPP APP(*FTP) SET(*ON)
```

This will start tracing for all FTP servers. Tracing for all other TCP applications is not affected.

Example 2:

```
TRCTCPAPP APP(*FTP) SET(*CHK)
```

This command is used to check the status of the tracing for the FTP server job(s). Assume that the last command entered was:

```
TRCTCPAPP APP(*FTP) SET(*ON) USER(JOECOOL)
```

The format of the response to this command would be a set of messages that would look similar to the following:

```
TCP45B7 TRCTCPAPP APP(*FTP) SET(*ON) USER(JOEC00L)
      MAXSTG(*DFT) TRCFULL(*WRAP)
TCP45B1 Tracing active for *FTP.
TCP45B2 Data capture begun for *FTP.
TCP45B3 Data buffer wrapped for *FTP.
```

Tracing FTP Client

To trace the FTP client:

To produce an FTP client trace or display the subcommands sent to the FTP server, use the DEBUG FTP client subcommand. The DEBUG subcommand toggles the debugging mode. If an optional debug-value is specified, it is used to set the debugging level. When debugging is on, each subcommand sent to the server is displayed and preceded by the string '>>>'. The debug-value must be set to 100 to produce an FTP client trace.

DEBug [debug value]

debug value

If the debug-value is 0, debugging is off. If the debug-value is a positive integer, debugging is on.

If no value is specified, the debug-value is toggled from zero to one or from a positive integer to zero.

100 Initiate an FTP client trace. The client continues running the trace until DEBUG is turned off or until the FTP client is ended. (When the trace is ended, there may be a significant delay while the trace data is formatted.)

Note: The FTP client trace should only be used for reporting software problems to IBM. System performance may be adversely affected by this function.

A new capability has been added to the FTP client for debugging for V4R4. This function is similar to the DEBUG 100 described above. When the V4R4 client is started, it first checks for the existence of a dataarea named QTMFTPD100.

You need to create the dataarea QTMFTPD100 in the QTEMP library using this command:

```
CRDTAARA DTAARA(QTEMP/QTMFTPD100) TYPE(*LGL) AUT(*USE)
```

If the QTMFTPD100 dataarea exists, then it will set the debug value to 100 and start an FTP client trace. The purpose of this capability is to enable FTP client debug traces to be done in those situations when an FTP client trace *cannot* be started by issuing the DEBUG 100 subcommand.

Getting a Copy of an FTP Server Job Log

A copy of the FTP server job log may be required to obtain additional information about errors that occur on the FTP server. Issuing the following subcommand from the FTP client causes the server to create a spooled file of its job log:

```
QUOTE RCMD DSPJOBLOG
```

To obtain a copy of error messages written to the server job log, this subcommand must be issued after the error has occurred. The user may then inspect the job log using the WRKSPLF command.

This technique is recommended in those cases where the reply message returned to the client from the server only provides minimal information about an error occurring on the server machine. For example, this method is useful for obtaining details about I/O errors that occur on the server machine.

If the error prevents the FTP server job log from being obtained by the method described here, enter the following command to force a spooled job log to be created for each FTP session:

```
CHGJOB JOB(QTCP/QTMFTPS) LOG(4 00 *SECLVL)
```

Then recreate the scenario which causes the error. To restore the original job log behavior after obtaining the required data, enter the following command:

```
CHGJOB JOB(QTCP/QTMFTPS) LOG(4 00 *NOLOG)
```

Determining Problems for SMTP

SMTP is designed much the same as the other TCP/IP functions and applications. Like TCP/IP or the FTP application, the SMTP jobs run under the QSYSWRK subsystem and produce job logs with information associated with the SMTP jobs. If the SMTP jobs end, the job logs can be used to determine the cause. If the mail is not getting to the desired user, the job logs can contain information that helps with the problem analysis.

The SMTP jobs do not start unless the QSYSWRK subsystem is running. SMTP jobs will start if the QSNADS subsystem is not running. If you do not see distributions on QSMTPQ when trying to send mail, ensure that the QSNADS subsystem is running. Issue STRSBS QSNADS and ensure that the mail server framework job QMSF is running in QSYSWRK by doing a STRMSF.

Notes:

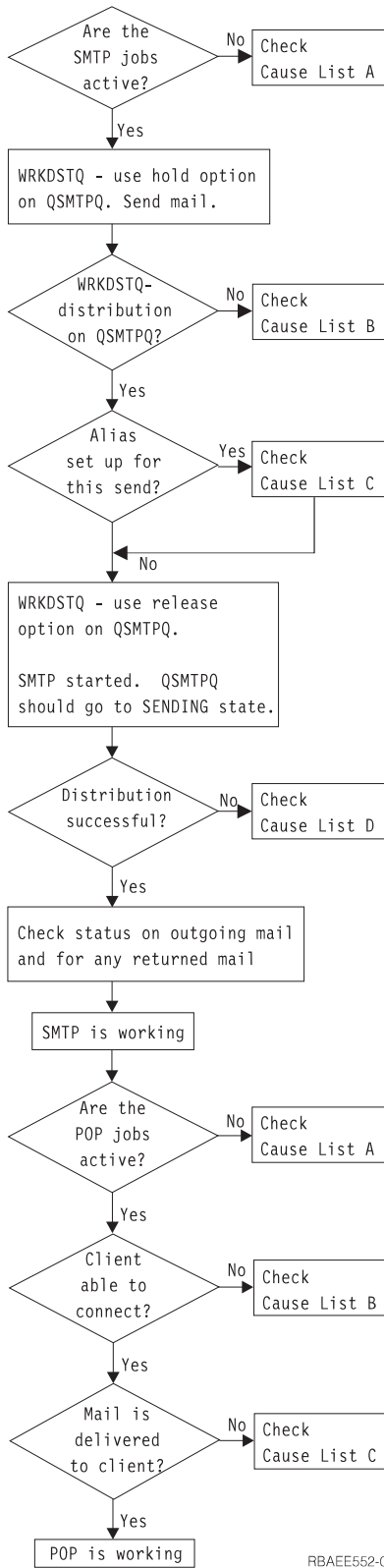
1. You will only see mail queuing up on QSMTPQ, if the recipient's System Distribution Directory (SDD) entry points to the Bridge Client. The SDD entry points to the Bridge Client if you have a Mail Service Level of "User index", a Preferred address of "User ID/Address", and a System name of TCPIP.
2. POP3 Mail, or Domino mail will *never* be seen queuing on QSMTPQ, because POP3 Mail and Domino mail go through the MSF framework. You will only see distributions on QSMTPQ as it applies to mail from OV, or SNDDST.

Configuring an AS/400 system to use SMTP can be an extensive process. The benefits of this extensive configuration are in the usability of the product. After configuration, the use of SMTP is no different than any of the AS/400 mail protocols. The user need not know whether the mail is being delivered by SNADS to another SNA system, or by SMTP. A successful configuration should allow the typical user to use the SMTP protocol without really needing to understand what is going on in the background.

Problems with SMTP are usually associated with mail not being accepted by the remote system or not being sent at all as shown in the following figure.

However, the situation can exist where the SMTP jobs do not fail, but they are not active either. You specified Yes for the *Remote name server retries* field, but the remote name server is not available. Do one of the following:

- Add the local internet address to the host table (use option 10 (Work TCP/IP host table entries) on the Configure TCP/IP menu)
- Establish the connection with the remote name server
- Specify No for the *Remote name server retries* field and start the SMTP jobs again:
STRTCPSVR *SMTP
- Configure the local domain and host names (option 12)



RBAAE552-0

Figure 260. SMTP Problem Analysis

Cause List A

1. Check to see if the SMTP jobs are active in the QSYSWRK subsystem. Use the Work with Active Jobs (WRKACTJOB) command. Type
WRKACTJOB SBS(QSYSWRK) JOB(QT*)

then use the Shift - F2 combination to include the SMTP pre-start jobs in the active jobs display.

The SMTP jobs that run in QSYSWRK are:

- QTSMPCLTD (SMTP client daemon)
- QTSMPSRVD (SMTP server daemon)
- QTSMPBRSR (SMTP bridge server)
- QTSMPBRCL (SMTP bridge client)
- QTSMPCLTP (SMTP client pre-start job(s))
- QTSMPSRVP (SMTP server pre-start job(s))

If these jobs are not running, this indicates that SMTP is not active. If it is known that SMTP had been running, check the job log for the SMTP jobs by using the WRKOUTQ or WRKJOB command. The spooled files can be found in the default queue that is associated with the SMTP jobs. Also check the QSNADS subsystem.

2. Check to see if the QDIA job is running in the QSNADS subsystem. Use the Work with Active Jobs (WRKACTJOB) command.
3. Use option 12 (Change local domain and host names) on the Configure TCP/IP menu to see if the local domain name is configured correctly.
4. Set the default value for acknowledgment to Yes. From the OfficeVision menu select option 4 (Send note). From the Send Note display, press F13 (Change defaults) to set the value to Yes.
5. Set low SMTP retry values so that you are quickly informed of distribution failures. Use option 3 (Change SMTP attributes) on the Configure TCP/IP SMTP menu.

Cause List B

1. Check the user ID and user address you are sending to in the system distribution directory using the Work with Directory Entries (WRKDIRE) command. Also look at the system name for this entry. The system name in the directory entry must match the system name that you entered in the routing table.
All recipients must be listed in the system distribution directory (even if an *ANY qualifier is used). This can be verified using the WRKDIRE command. OfficeVision does not allow the user to send mail to anyone not listed in this directory. When adding an entry to the system distribution directory for SMTP usage, make sure that the system and group names are exactly the same as one of the entries in the SNADS routing table that was configured to use SMTP. The SNADS routing table can be viewed using option 2 (Routing table) on the Configure Distribution Services display.
2. Use option 2 (Routing table) of the Configure Distribution Services (CFGDSTSRV) command to verify that the system name from the previous step is QSMTPQ. If this is not the system name, you are not sending your distributions using SMTP.
3. Verify whether or not the recipient's address has been configured to use SMTP. Use option 2 (Routing table) on the Configure Distribution Services display to look for the system address of the recipient. If the entry does not exist, add it

accordingly. If it does, verify that this address is intended to have its mail sent using the SMTP *RPDS queue called QSMTPQ.

4. Make sure that there is at least one mail server framework (QMSF) job running in subsystem QSYSWRK. If there are no QMSF jobs running, start the mail server framework with the STRMSF command.

Cause List C

1. Verify whether or not an alias entry is necessary to reach the destination.

The SNADS user ID and address are limited to 8 characters each. This means that it is necessary to have an alias entry for all receivers that have user IDs exceeding 8 characters, or that use special characters. The alias entry for each user must be specified using the WRKNAMSMTP command.

Cause List D

1. Verify that the local domain name is correct.

Use option 12 (Change local domain and host names) on the Configure TCP/IP menu to find out the local domain name. Make sure that the host name that was found here exists in the local host table or on the remote name server. To see if this host name is in the host table, use option 10 (Work with TCP/IP host table entries) on the Configure TCP/IP menu. If it is not in the host table or on the remote name server, then add it to the host table by typing option 1 (Add) on the Work with TCP/IP Host Table Entries display.

Each entry in the host table consists of two parts: the internet address and the corresponding host names. The internet address of this local host name about to be added has to be one of this system's local internet addresses defined on the TCP/IP interface that was configured using option 1 (Work with TCP/IP interfaces) on the Configure TCP/IP menu.

If the host name is not in the host table or there is a mismatch between the interface internet addresses and the one in the host table for this host, the local domain name is considered not valid. If the local domain name is not valid, then when the SMTP jobs come up either when the QSNADS subsystem or QSYSWRK subsystem is started, they end. This is because incoming mail would not work, and outgoing mail would have incorrect information in the mail headers.

If the local domain name is incorrectly configured, an error message is put in the job log for the SMTP jobs. The job logs can be found by using the WRKOUTQ or WRKJOB command. The job logs can be found in the default queue that is associated with the SMTP jobs.

2. Verify that the mail router, if specified, is correct. Use Option 3 (Change SMTP attributes) on the Configure TCP/IP SMTP menu to identify the mail router. Correct the mail router entry or delete it, if necessary.
3. Verify that the delimiter for the SMTP user ID, if specified, is correct. Use option 3 (Change SMTP attributes) on the Configure TCP/IP SMTP menu to identify the delimiter character. You should be aware that changing the delimiter value causes the names that use the previous value to not be recognized. You need to change the names in the alias tables to reflect the new delimiter character.

If one of the AS/400 systems in the network uses a delimiter character, it is recommended that all the AS/400 systems in that network use that character as the delimiter.

Note: SMTP is not involved until a distribution goes to the QSMTPQ distribution queue. The problem could be in your SNADS configuration, for example, the directory entries (WRKDIRE command) or the routing and queue tables (CFGDSTSRV command).

Determining Problems for SMTP When Using OfficeVision

When first using SMTP with OfficeVision, it is recommended that you change the defaults to confirm delivery of the note. This enables you to track the progress of the note and provides some indication of where things can go wrong.

To track the progress of your mail, type option 2 for mail from the OfficeVision menu and then press F6 (Outgoing mail status) from the Work with Mail display. The progress of a successful item of mail is shown in Figure 261 and Figure 262.

```

View Outgoing Mail Status Details
Description . . . . . : User1 to User2
Date/Time sent . . . . . : 03/28/90 02:48:38 p.m.
Confirm delivery . . . . . : Y
-----Sent To----- ---Confirmed---
Status      User ID  Address  Description      Date      Time
SENT        USER2   AS400B   Any TCP/IP user on AS400B

```

Figure 261. View Outgoing Mail Status (1 of 2)

```

View Outgoing Mail Status Details
Description . . . . . : User1 to User2
Date/Time sent . . . . . : 03/28/90 02:48:38 p.m.
Confirm delivery . . . . . : Y
-----Sent To----- ---Confirmed---
Status      User ID  Address  Description      Date      Time
DELIVERED   User2   AS400B   Any TCP/IP user on AS400B 03/28/90 14:54

```

Figure 262. View Outgoing Mail Status (2 of 2)

If the mail fails for any reason, the sender receives an outgoing mail status of failed. The returned mail should provide some indication of what went wrong. To view returned mail, enter option 5 (View) next to a mail item from the Work with Mail display. Some of the more typical error messages are shown in the following displays:

```

MAIL P:12                VIEW Instruction        Pg:1      Ln:1
<2.....3.....4.....5.....6v.....7.....8.....9.....
Date: Wed, 28 Mar 90 13:56:51 .
From: QGATE@AS400A.SYSNAM1.COMPANY.COM
To: USER1?AS400A@AS400A.SYSNAM1.COMPANY.COM
Subject: Undeliverable Mail
AS400A.SYSNAM1.COMPANY.COM unable to deliver following mail.
<USER2?AS400B@AS400B.SYSNAM1.COMPANY.COM>
Retries exhausted while attempting to connect to remote host system AS400A.RCH
F3=Exit          F7=Window        F12=Cancel      F16=File remote
F4=Find char     F8=Reset         F13=Edit option F17=Function
F5=Goto          F10=Forward      F14=Delete mail F19=Print
F6=Find          F11=Reply        F15=File local  F21=Nondisplay keys

```

Figure 263. Mail Error Message, Example 1 (1 of 2)

```

MAIL P:12                VIEW                Pg:1      Ln:16
<2.....3.....4.....5.....6v.....7.....8.....9.....
Retries exhausted while attempting to connect to remote host system AS400A.RCH
AS400B
** Text of Mail follows **
Received from AS400A by AS400A.SYSNAM1.COMPANY.COM (SMTP Version 1) Release 3.0
Date: Wed, 28 Mar 90 13:50:47 .
From: USRE1?AS400A%AS400A@AS400A.SYSNAM1.COMPANY.COM
To: USER2?AS400B@AS400B.SYSNAM1.COMPANY.COM
F3=Exit          F7=Window        F12=Cancel      F16=File remote
F4=Find char     F8=Reset         F13=Edit option F17=Function
F5=Goto          F10=Forward      F14=Delete mail F19=Print
F6=Find          F11=Reply        F15=File local  F21=Nondisplay keys

```

Figure 264. Mail Error Message, Example 1 (2 of 2)

Check the following if you do receive an error:

- The SMTP job is active in the QSYSWRK subsystem.
- There is a routing entry in the QSNADS subsystem for SMTP.
- There is an SMTP alias table on the remote host for the recipient.

After receiving the error shown in Figure 265 on page 464 and Figure 266 on page 464, you should verify that there is a directory entry on the remote host for the recipient.

```

MAIL P:12                               VIEW Instruction          Pg:1      Ln:1
<2.....3.....4.....5.....6v.....7.....8.....9.....
Date: Wed, 28 Mar 90 14:20:54 .
From: QGATE@AS400A.SYSNAM1.COMPANY.COM
To: USER1?AS400A@AS400A.SYSNAM1.COMPANY.COM
Subject: Undeliverable Mail
AS400A.SYSNAM1.COMPANY.COM unable to deliver following mail.
<USER2@AS400B>
AS400A.SYSNAM1.COMPANY.COM received negative reply from host:
F3=Exit           F7=Window         F12=Cancel       F16=File remote
F4=Find char      F8=Reset          F13=Edit option  F17=Function
F5=Goto           F10=Forward       F14=Delete mail  F19=Print
F6=Find           F11=Reply         F15=File local   F21=Nondisplay keys

```

Figure 265. Mail Error Message, Example 2 (1 of 2)

```

MAIL P:12                               VIEW                               Pg:1      Ln:16
<2.....3.....4.....5.....6v.....7.....8.....9.....
AS400A.SYSNAM1.COMPANY.COM received negative reply from host:
AS400B
550 User USER2 not recognized.
** Text of Mail follows **
Received from AS400A by AS400A.SYSNAM1.COMPANY.COM (SMTP Version 1) Release 3.0
Date: Wed, 28 Mar 90 14:15:51 .
From: USER1?AS400A%AS400A@AS400A.SYSNAM1.COMPANY.COM
F3=Exit           F7=Window         F12=Cancel       F16=File remote
F4=Find char      F8=Reset          F13=Edit option  F17=Function
F5=Goto           F10=Forward       F14=Delete mail  F19=Print
F6=Find           F11=Reply         F15=File local   F21=Nondisplay keys

```

Figure 266. Mail Error Message, Example 2 (2 of 2)

Determining Problems for SMTP Without Using OfficeVision

You use the SNDDST, QRYDST, and RCV DST commands when not using OfficeVision. If you have successfully sent a distribution using the SNDDST command, you receive this message Send distribution completed successfully. This is the only indication that the distribution has been sent and in no way guarantees the distribution has been received.

It is recommended that you specify Yes for the Confirmation of delivery (CFMDEL) parameter on the SNDDST command. By specifying Yes, a confirmation of delivery notification is sent to the user if the distribution fails, which can help to show where the problem is.

The following examples show the type of information that can be found about messages sent and received.

Figure 267 on page 465 shows how to determine if there are incoming distributions for user QSECOFR on RCHSX383 using the QRYDST command.

```

Query Distributions (QRYDST)

Type choices, press Enter.

Incoming or outgoing . . . . . *IN          *IN, *OUT
User identifier:
  User ID . . . . . >QSECOFR      Character value, *CURRE
  Address . . . . . >RCHSX383    Character value

Additional Parameters

File to receive output . . . . . dstfile   Name, *NONE
Library . . . . . qgpl           Name, *LIBL, *CURLIB
Output member options:
  Member to receive output . . . *FIRST   Name, *FIRST
  Replace or add records . . . . *REPLACE *REPLACE, *ADD
  Status . . . . . *ALL          *ALL, *NEW, *OLD, *OPEN
Command character identifier:
  Graphic character set . . . . *SYSVAL  Number, *SYSVAL, *DEVD
  Code page . . . . .           Number

F3=Exit  F4=Prompt  F5=Refresh  F12=Cancel  F13=How to use this display
F24=More keys

```

Figure 267. Example of Querying for Distributions for QSECOFR

Figure 268 shows two messages that were found as a result of the QRYDST command that may be of interest to the user.

```

Display Physical File Member
File . . . . . : DSTFILE           Library . . . . . : QGPL
Member . . . . . : DSTFILE         Record . . . . . : 1
Control . . . . .           Column . . . . . : 1
Find . . . . .
*...+...1...+...2...+...3...+...4...+...5...+...6...+...7...+...
000004SUN3  TESTER 0027QSECOFR RCHSX383TESTER SUN3 19900403160200000000
000008SYS383 QGATE 0026QSECOFR RCHSX383QGATE SYS383 19900403154800000000
***** END OF DATA *****

```

Figure 268. Example of Distributions Received by QSECOFR

These two distributions are received using the RCVDST command as shown in Figure 269 on page 466 and Figure 270 on page 466.

```

                                Receive Distribution (RCVDST)
Type choices, press Enter.
Distribution identifier . . . . sun3  tester  0027
User identifier:
  User ID . . . . . *CURRENT      Character value, *CURRENT
  Address . . . . .                Character value
  Document . . . . . *NONE        Name, *NONE
  In folder . . . . . *NONE
  File to receive output . . . . msg      Name, *NONE
  Library . . . . . tcpliba       Name, *LIBL, *CURLIB
Output member options:
  Member to receive output . . . *FIRST   Name, *FIRST
  Replace or add records . . . . *REPLACE *REPLACE, *ADD
  Type of data for output . . . . *DFT     *DFT, *ALL, *DSTINFO, *MSG...
      + for more values
  Acknowledge receipt . . . . . *YES      *YES, *NO
  Distribution ID extension . . . *NONE    1-99, *NONE
                                                    More...
F3=Exit  F4=Prompt  F5=Refresh  F10=Additional parameters  F12=Cancel
F13=How to use this display  F24=More keys

```

Figure 269. Receiving the First Distribution into a File

```

                                Receive Distribution (RCVDST)
Type choices, press Enter.
Distribution identifier . . . . > 'sys383 qqgate  0026'
User identifier:
  User ID . . . . . *CURRENT      Character value, *CURRENT
  Address . . . . .                Character value
  Document . . . . . *NONE        Name, *NONE
  In folder . . . . . *NONE
  File to receive output . . . . > MSG      Name, *NONE
  Library . . . . . > TCPLIBA     Name, *LIBL, *CURLIB
Output member options:
  Member to receive output . . . *FIRST   Name, *FIRST
  Replace or add records . . . . *add     *REPLACE, *ADD
  Type of data for output . . . . *DFT     *DFT, *ALL, *DSTINFO, *MSG...
      + for more values
  Acknowledge receipt . . . . . *YES      *YES, *NO
  Distribution ID extension . . . *NONE    1-99, *NONE
                                                    More...
F3=Exit  F4=Prompt  F5=Refresh  F10=Additional parameters  F12=Cancel
F13=How to use this display  F24=More keys

```

Figure 270. Receiving the Second Distribution into a File

The second message is added in the file so that both can be viewed at the same time.

The following eight displays show the two messages received by this user. These displays show the information contained in the database member using F20 (Right) to view the information. The lines of information do not wrap to the next line but instead continue on through the series of displays.

The first message is a successful reply from User Tester at SUN3. The second is an error message indicating that the distribution failed. The reasons for failure are mainly the TCP/IP configuration errors, and in this case the fact that the SNADS job was not active in the QSYSWRK subsystem.

```

Display Physical File Member
File . . . . . : MSG          Library . . . . . : TCPLIBA
Member . . . . . : MSG          Record . . . . . : 1
Control . . . . .           Column . . . . . : 1
Find . . . . .
*...+...1...+...2...+...3...+...4...+...5...+...6...+...7...+...
000001SUN3  TESTER 0027QSECOFR RCHSX383TESTER SUN3 19900403160200000000
000002SUN3  TESTER 0027QSECOFR RCHSX383TESTER SUN3 19900403160200000000
000003SUN3  TESTER 0027QSECOFR RCHSX383TESTER SUN3 19900403160200000000
000001SYS383 QGATE 0026QSECOFR RCHSX383QGATE SYS383 19900403154800000000
000002SYS383 QGATE 0026QSECOFR RCHSX383QGATE SYS383 19900403154800000000
000003SYS383 QGATE 0026QSECOFR RCHSX383QGATE SYS383 19900403154800000000
000004SYS383 QGATE 0026QSECOFR RCHSX383QGATE SYS383 19900403154800000000
***** END OF DATA *****

```

Figure 271. Received Messages (1 of 8)

```

Display Physical File Member
File . . . . . : MSG          Library . . . . . : TCPLIBA
Member . . . . . : MSG          Record . . . . . : 1
Control . . . . .           Column . . . . . : 79
Find . . . . .
.8...+...9...+...0...+...1...+...2...+...3...+...4...+...5...+...
0045400002IBM AS/400 337256004010****
0045400002IBM AS/400 337256008105SMTP0001
0045400002IBM AS/400 000000454800Received from sun3.sysnam123.company.com by
0066200002IBM AS/400 337256004010****
0066200002IBM AS/400 337256018105Undeliverable Mail
0066200002IBM AS/400 000000500800Date: Tue, 03 Apr 90 15:48:40 . F
0066200002IBM AS/400 00000016280015:43:22 . From: QSECOFR?RCHSX383
***** END OF DATA *****

```

Figure 272. Received Messages (2 of 8)

```

Display Physical File Member
File . . . . . : MSG          Library . . . . . : TCPLIBA
Member . . . . . : MSG          Record . . . . . : 1
Control . . . . .           Column . . . . . : 157
Find . . . . .
...6...+...7...+...8...+...9...+...0...+...1...+...2...+...3....
3.SYSNAM1.COMPANY.COM (SMTP Version 1) Release 3.0 with TCP. Received: by sun3
rom: QGATE@SYS383.SYSNAM1.COMPANY.COM To: QSECOFR?RCHSX383@SYS383.SYSNAM1.COM
%SYS383@SYS383.SYSNAM1.COMPANY.COM To: QSECOFR@SYS378C Subject: NO SUBJECT
***** END OF DATA *****

```

Figure 273. Received Messages (3 of 8)

```

Display Physical File Member
File . . . . . : MSG          Library . . . . . : TCPLIBA
Member . . . . . : MSG          Record . . . . . : 1
Control . . . . .           Column . . . . . : 235
Find . . . . .
+....4....+....5....+....6....+....7....+....8....+....9....+....0....+....1..
hland.company.com (4.0/SMI-4.0) id AA14974; Tue, 3 Apr 90 15:57:48 CDT Date:
Subject: Undeliverable Mail      SYS383.SYSNAM1.COMPANY.COM unable to deliver
Hello there can you answer this ??
***** END OF DATA *****

```

Figure 274. Received Messages (4 of 8)

```

Display Physical File Member
File . . . . . : MSG          Library . . . . . : TCPLIBA
Member . . . . . : MSG          Record . . . . . : 1
Control . . . . .           Column . . . . . : 313
Find . . . . .
..+....2....+....3....+....4....+....5....+....6....+....7....+....8....+....9
e, 3 Apr 90 15:57:48 CDT From: tester@sun3 Message-Id: <9004032057.AA14974@s
lowing mail. <QSECOFR@SYS378C> Retries exhausted while attempting to co
***** END OF DATA *****

```

Figure 275. Received Messages (5 of 8)

```

Display Physical File Member
File . . . . . : MSG          Library . . . . . : TCPLIBA
Member . . . . . : MSG          Record . . . . . : 1
Control . . . . .           Column . . . . . : 391
Find . . . . .
.....+....0....+....1....+....2....+....3....+....4....+....5....+....6....+...
un3.sysnam123.company.com> To: QSECOFR?RCHSX383@SYS383.sysnam123.company.com
nnect to remote host system SYS383.SYSNAM1.COMPANY.COM.      SYS378C
***** END OF DATA *****

```

Figure 276. Received Messages (6 of 8)

```

Display Physical File Member
File . . . . . : MSG          Library . . . . . : TCPLIBA
Member . . . . . : MSG          Record . . . . . : 1
Control . . . . .           Column . . . . . : 469
Find . . . . .
.7....+....8....+....9....+....0....+....1....+....2....+....3....+....4....+
re SECOFR this is tester on SUN 3 replying to your distribution.  Bye for
** Text of Mail follows ** Received from SYS383 by SYS383.SYSNAM1.COMPANY.COM
***** END OF DATA *****

```

Figure 277. Received Messages (7 of 8)


```

Display Physical File Member
File . . . . . : MSG          Library . . . . . : TCPLIBA
Member . . . . . : MSG          Record . . . . . : 1
Control . . . . .           Column . . . . . : 547
Find . . . . .
...5....+...6....+...7....+...8....+...9....+...0....+...1....+...2....
                                01RCHSX3830
                                01RCHSX3830
now, Tester.....                01RCHSX3830
                                01RCHSX3830
                                01RCHSX3830
                                01RCHSX3830
(SMTP Version 1) Release 3.0 with BSMTMP id. Date: Tue, 03 Apr 90 01RCHSX3830
                                01RCHSX3830
                                ***** END OF DATA *****

```

Figure 278. Received Messages (8 of 8)

Tracing SMTP Distributions

Each of the four SMTP jobs uses a flight recorder to collect trace information as distributions are processed by SMTP. The flight recorders are created automatically as rolling buffers in user space objects that can store approximately 2000 records of trace information. Once the flight recorder user spaces become full, the oldest trace records are overwritten when new trace records are generated. The five flight recorder user spaces are found in the QTEMP library for each job as *USRSPC object types with object names QTMSFLRCS1 through QTMSFLRCS5. To view the contents of an SMTP flight recorder:

1. Type the following command to check that SMTP jobs are running:
WRKACTJOB JOB(QTSM*)
2. If no active jobs are displayed, type the following command to start the SMTP jobs:
STRTCPSVR SERVER(*SMTP)
3. Enter option 6 (Release) next to the job for the flight recorder that you wish to view. A message will appear at the bottom of the screen with the job number/user/job name. Note this information and enter STRSRVJOB (Start Service Job) with the information as parameters on the command.
4. Type the following command to dump the flight recorder for the SMTP job that is being serviced:
DMPYSOBJ OBJ(QTMSFLRCS1) CONTEXT(QUSRSYS)

The message Dump output directed to spooled file appears.

The object name in the example (QTMSFLRCS1) will dump the flight recorder user space for job 1 (QTSMTPCLTP, an SMTP client pre-start job) only. To dump the flight recorder spaces for the other jobs, repeat the command substituting the numbers '2', '3', '4', or '5' in place of the number '1'.

5. Type the following command to end job servicing:
ENDSRVJOB

The job names corresponding to these job numbers are QTMSFLRCS2 for the QTSMTPSRVP server pre-start jobs, QTMSFLRCS3 for the QTSMTPBRCCL bridge client job, QTMSFLRCS4 for the QTSMTPBRSR bridge server job, and QTMSFLRCS5 for the QMSF message forwarding jobs.

6. Type the following command to view the spooled file(s) created:
WRKSPLF
7. Press F18 (Bottom) to get to the bottom of the spooled file list if More... appears on the display.
8. Press F11 (View 2) to view the date and time of the file you want to work with.
9. Verify that you are working with the most recent spooled files.
10. Print all of the spooled files and job logs associated with the SMTP jobs.

You can gather information to analyze potential problems in the Mail Server Framework with this procedure:

1. Type this command to register the dump exit program:

```
WRKREGINF
```

All MSF exit point names begin with QIBM_QZMFMSF_ so you can run this command:

```
WRKREGINF EXITPNT(QIBM_QZMFMSF_*)
```

2. Pick an exit point in which to register. QIBM_QZMFMSF_ACT is usually a good choice if no MSF jobs are ending. If some MSF jobs are ending, pick the same exit point which is causing the MSF job to end. Type option 8 to select the exit point.
3. If one of the exit programs listed is causing an MSF job to end, remember its Exit Program Number.
4. Type option 1 to add an exit point. Do *not* type a program name. Press Enter.
5. Enter an Exit Program Number. This is a sequence number. The MSF runs exit programs in order from the lowest to the highest sequence number (if the programs are registered for the appropriate types). If an MSF job is ending, enter a smaller number than the exit program which is causing the MSF job to end.
6. Type the program name QZMFDUMP, library QSYS.
7. Press F10 for additional parameters, then scroll downward.
8. Type Length of data 12. The program data is SPCL01009999 (the 9999 indicates this: register for all types).
9. Restart MSF (enter ENDMSF, then STRMSF).
10. Output an QMSF spool file, WRKSPLF QMSF.
11. Once you get the dump(s) you want, unregister QZMFDUMP and end and restart MSF.

Journaling for SMTP is a tool included in the V4R4 release that allows for tracking the flow of mail through the major mail components. This tool could help you to explain the reasons why mail may not be delivered. The tool provides a trail for the delivery of mail at periodic intervals. It is enabled by setting an SMTP configuration parameter that is new for V4R4.

Type the following command to enable journaling for SMTP:

```
CHGSMTPA JOURNAL(*YES)
```

Type the following command to disable journaling, so that journal entries are no longer generated:

```
CHGSMTPA JOURNAL(*NO)
```

The name of the journal used for SMTP journaling is QUSRSYS/QZMF.

The journaling subtypes and codes that are specific to SMTP and related mail functions are noted in the following tables. They can be identified in the SMTP journal (QUSRSYS/QZMF) by appending a code from the Journal Entry Codes table to a function identifier from the Journal Entry Types table. All other information provided with the subtypes and codes for the specific functions is noted in the tables and examples that follow .

Table 38. Journal Entry Types for LG (Extensions)

Function Identifier	Description
6	Bridge client entry
7	Bridge server entry
8	SMTP client
9	SMTP server
A	MSF non delivery
B	MSF local delivery
C	MSF message forwarding
D	POP create message
E	Send mail API
F	Domino MTA
G	Tunneling snap-in
H	SNADS (switcher)
I	MIME parser (a local delivery snap-in)
L	FAX (local delivery)
M	SNADS

Journal Entry Codes:

```

JRN_ORG = '1' /* STC O Originator name follows */
, JRN_RCP = '2' /* STC R Recipient name */
, JRN_UDL = '3' /* STC U Undeliverable recipient */
, JRN_LIN = '4' /* STC LIN TO SRVR IPADDR (from host) */
, JRN_RIN = '5' /* STC RIN TO SRVR IPADDR */
, JRN_TOQ = '6' /* STC PGMNAME TO QTMSOUTQ */
, JRN_TIQ = '7' /* STC PGMNAME TO QTMSINQ */
, JRN_DLV = '8' /* STC DLVED IPADDRESS (of host) */
, JRN_MID = '9' /* STC MSGID: <interal 822 Msg Id> */
, JRN_FIQ = 'A' /* STC QTMSINQ TO PGMNAME */
, JRN_FOQ = 'B' /* STC QTMSOUTQ TO PGMNAME */
, JRN_TRQ1 = 'C' /* STC PGMNAME TO QTMSRTQ1 */
, JRN_TRQ2 = 'D' /* STC PGMNAME TO QTMSRTQ2 */
, JRN_FRQ1 = 'E' /* STC QTMSRTQ1 TO QTMSOUTQ */
, JRN_FRQ2 = 'F' /* STC QTMSRTQ2 TO QTMSOUTQ */
, JRN_TID = 'G' /* STC MSGID MAP TO <MSF ID > */
, JRN_TMSF = 'H' /* STC PGMNAME to MSF */
, JRN_RERR = 'K' /* STC RESERR errno domain */
, JRN_TSSQ = 'L' /* STC PGMNAME TO QTMSBSSQ */
, JRN_FSSQ = 'M' /* STC QTMMBSSQ TO PGMNAME */
, JRN_TMFQ = 'N' /* STC PGMNAME TO QMSFQUEUE */
, JRN_TSDS = 'O' /* STC PGMNAME to SNADS */
, JRN_UNDL = 'P' /* STC UNDELIVERABLE NOTICE */
, JRN_TUNIN= 'Q' /* STC LOCAL DEL BY TUNNELING */
, JRN_COD = 'R' /* STC CRT COD MSG */
, JRN_SPAM = 'S' /* STC CONNECT REFUSED IPADDR */
, JRN_SIZE = 'T' /* STC MSG SIZE */
, JRN_UDEL = 'U' /* STC DELETING UNDEL MSG */
, JRN_RELAY = 'V' /* STC RELAY REFUSED IPADDR */

```

Table 39. Bridge Client

Type	Action	SubType/Code(s)	Comments
LG	Sending mail	61 62	Just prior to switch, log SNADS originator and recipients

Table 39. Bridge Client (continued)

Type	Action	SubType/Code(s)	Comments
LG	Sending remote mail	66	When enqueueing mail on output queue
LG	Sending bad mail back to SNADS	67	This log is only made if a problem was found with the mail that SNADS sent to the Bridge client

Table 40. Bridge Server

Type	Action	SubType/Code(s)	Comments
LG	Getting mail off of the "IN" queue	7A	Log mail being dequeued from QTMSINQ
LG	Passing off mail to SNADS	7O	Record successful transfer to QSNADS
LG	Putting container on the "BUSY" queue because of space usage	7L	Record when mail is enqueued on QTMSBSSQ because of threshold overflow
LG	Getting mail off of "BUSY" queue	7M	Record dequeuing mail from QTMSBSSQ, space was reclaimed, and the mail can now be processed
LG	Passing message to MSF	7H 71 72	Record when message gets inserted into the framework
LG	Creating COD message	7R 7G	Record when COD message gets inserted into the framework; log MSF MSGID since the new COD message is being created
LG	Cannot deliver this piece of mail to a recipient	7P	Log the fact that an undeliverable notice is being created
LG	Cannot deliver an undeliverable notice because of a bad return address	7U	Log the fact that the note is being deleted

Table 41. SMTP Client

Type	Action	SubType/Code(s)	Comments
LG	Dequeuing of container for processing	8B	Just after floater tag is set log dequeue of Mail
LG	Successful mail delivery	88 82	Log each successful send to recipient
LG	Undeliverable mail	83	Log undelivered mail
LG	1rst level time-out	8C	Log when adding to 1st level retry queue
LG	2nd level time-out	8D	Log when adding to 2nd level retry queue
LG	Mail is ready to be retried	8E 8F	Log when retried mail is put back on QTMSOUTQ
LG	COD is being sent back to originator	87	Log when COD is enqueued on BRSR queue

Table 41. SMTP Client (continued)

Type	Action	SubType/Code(s)	Comments
LG	Cannot process, resource is busy	86	Log when mail gets put back on QTMSOUTQ because connection matrix is full
LG	Examine recipient records	86	Log when mail gets put back on QTMSOUTQ because the recipient status changed
LG	Undeliverable	87	Log transfer of mail to QTMSINQ for undelivery notice
LG	MX query	8K	Log a resend failure

Table 42. SMTP Server

Type	Action	SubType/Code(s)	Comments
LG	Receiving mail	94 91 92 9T 99	Log reception of mail just after receiving ending sequence CRLF <.>CRLF (local). Log both originator and recipient. Message size nnnnn where nnnnn is the number of bytes. Log RFC822 MsgID.
LG	Receiving relayed mail	95 91 92	Log MCB just after receiving ending sequence CRLF <.>CRLF(relayed). Log both originator and recipient.
LG	Passing off mail to Bridge client	97	Log entry of MCB into QTMSINQ (incoming mail)
LG	Passing off mail to client for remote delivery	96	Log entry of MCB into QTMSOUTQ (relayed mail)
LG	Rejecting connect	9S	Mail message was rejected because the source IP address was in the reject connection list
LG	Rejecting relayed mail	9V	Mail message was rejected because the source IP address was not in the accept relay list

Table 43. MSF Message Forwarding Snap-In (Exit Program)

Type	Action	SubType/Code(s)	Comments
LG	Checking availability	CN	Record MsgId that was put back on QMSF queue due to SMTP not being started
LG	Enqueuing the mail	C6 C1 C2	Log mail being put onto QTMSOUTQ

Table 44. Additional MSF and POP Journal Points

Type	Action	SubType/Code(s)	Comments
LG	Reinsertion of parsed MIME note into framework	IH I1 I2 IG	Log when the parsed MIME message is reinserted into the MSF. Message size nnnnn where nnnnn is the size of message (all attachments)

Table 44. Additional MSF and POP Journal Points (continued)

Type	Action	SubType/Code(s)	Comments
LG	Sending COD message into MSF	BR B1 B2	Record insertion of COD message into the MSF
LG	Creation of nondelivery message	AP A1 A2	Record non delivery message being inserted into MSF
LG	POP receives message from MAPI client and sends it into MSF	D1 D2 DT D9 DH	Record POP inserting message into the MSF. (This is a use of the XTND XDESCRIPT).
LG	Use of the SendMail API	EH E1 E2 ET	Record creation of message by SendMail API. Message size nnnnn where nnnnn is the size of message (all attachments)
LG	Mail is targeted to a SNADS bridged remote system	G8 G2	Record when message is tunneled. Include system message is sent to.
LG	Mail tunneled through a SNADS bridge is received	GQ G2	Record receiving tunneled message for local delivery
LG	Mail is delivered into a POP mail box	B8 B2	Record delivery of message to local POP mail box. IP address will be the POP mailbox directory. Recipient will also be listed.

Table 45. Journaling Points Not in SMTP or the Framework

Type	Action	SubType/Code(s)	Comments
LG	Address resolution SNADS switches either from/to	H1	SNADS switched a message into the MSF Originator (H1) and Recipients (H2) are listed. Message size is nnnnn where nnnn is the number of bytes.

Note: For more information about setting up journals and viewing the entries, see *Backup and Recovery*, SC41-5304-03. For more information on the QZMF journal, see *AnyMail/400 Mail Server Framework Support*, SC41-5411-00 .

You can enable a serviceability tool for gathering trace data associated with the SMTP server (inbound connections) and client (outbound connections) paths. You can enable this tool through the CL command TRCTCPAPP.

Trace data for inbound connections can be filtered by remote IP address and port, or by mail addressed to a particular recipient. Trace data for outbound connections can be filtered by the recipient's host name, the address of a particular recipient, or to include only mail exchanger information.

To start collecting trace data for the SMTP client for mail sent to a particular recipient's host, type the following:

```
TCPTCPAPP APP(*SMTPCLT) SET(*ON) HOST('abc.def.com')
```

Note: For more information regarding this command and the various parameters and filters, see *System API Reference: Program and CL Command APIs*, SC41-5870-03.

Materials Required for Reporting SMTP Problems

Any SMTP problem reported to IBM should include the following:

- QTCPIP and the SMTP job logs for user QTCP (QTSMPCLTD, QTSMP SRVD, QTSMPBRSR, QTSMPBRCL, QTSMPCLTP, QTSMP SRVP).
- If the trace information from any of the flight recorder user spaces (QTMSFLRCS1 through QTMSFLRCS5 from each job's QTEMP library) was dumped, the spooled file containing the contents of each flight recorder user space dumped is also required.
- Any status distributions.
- A communications trace from the time of the failure (Format TCP/IP data only formatted for ASCII. If you are not familiar with the procedure for collecting a communications trace, refer to "Collecting a Communications Trace" on page 493.
- File QATMSMTP containing the SMTP configuration information.
- The code release and the latest PTF for SMTP that is loaded and applied.

If any of the SMTP jobs logged software error data (FFDC), submit that information.

Note: The system value QSFWERRLOG must be set to *LOG for software error logging to occur. If an error occurs while QSFWERRLOG is set to *NOLOG, change the value to *LOG. Then try to re-create the error, and submit the logged software error data.

Cleaning Up Unprocessed SMTP Distributions

You can create a single-character data area that causes SMTP to destroy as it is starting distributions that it cannot process. Call the data area QTMSCLEAN and put it in the QUSRSYS library. Use the character in this data area to indicate the extent of mail control block clean-up:

- If you want a "cold" start to free all mail control blocks, use an upper- or lower-case "c" in this data area.
- Any other character causes a "warm" start to free only the first "floater" mail control block found. A **floater** is an MCB that was dequeued but was not fully processed, or failed processing when SMTP ended (for whatever reason).

When SMTP starts, a recovery function looks in the QUSRSYS library for this data area and proceeds accordingly.

Since a warm start frees only a single MCB, it may be necessary to stop SMTP, create the data area, and restart SMTP more than once to get to the MCB that is causing the problem. If there are a large number of MCBs, a cold start may be the fastest way to correct the problem.

To create the QTMSCLEAN data area, use the Create Data Area (CRTDTAARA) command.

Cold Start

The following command causes a cold start to occur and frees all the mail control blocks found the next time SMTP starts up:

```
CRTDTAARA DTAARA(QUSRSYS/QTMSCLEAN) TYPE(*CHAR)
LEN(1) VALUE('c') AUT(*ALL)
```

Important!: When you do a cold start, all unprocessed SMTP distributions found in the system are destroyed. Mail will be lost.

Warm Start

To cause a warm start to occur and free only the first floater mail control block found the next time SMTP starts up, use this command:

```
CRTDTAARA DTAARA(QUSRSYS/QTMSCLEAN) TYPE(*CHAR)
LEN(1) VALUE('W') AUT(*ALL)
```

Once the clean-up is complete, code within SMTP deletes the QTMSCLEAN data area from library QUSRSYS (provided the data area was created using the *ALL authority).

Determining Problems with the POP Server

The Post Office Protocol (POP) has multiple personalities. The standard POP server is similar to other TCP/IP functions and applications. The Client Access connections to the POP server come in IP, IPX and SNA versions. All of these versions run in subsystem QSYSWRK and produce job logs. If the POP server jobs end or if mail is not getting to the destination these job logs can be used to determine the cause.

Problems with Mail Delivery

The POP server requires the SMTP and QMSF products to be running in subsystem QSYSWRK for mail to be delivered. If these products are not running, the client can make a connection to the POP server, but no mail can be received by the client. OfficeVision/400 and the QSNADS subsystem must also be operational to allow the exchange of mail between POP clients and OfficeVision clients or to allow the connection of SNA clients to the POP server. For Client Access connections, the "Host Servers" must also have been started using the Start Host Server (STRHOSTSVR) command. IPX must also be running for the POP IPX server to function.

Note: If IPX is not running, the POP IPX server will fail on startup and post a job log stating that IPX is not running.

Entries in the System Distribution Directory (SDD) are required for mail to be delivered to a POP client. Each attached client must have:

- A user profile and valid password on the AS/400
- An entry in the SDD for that user with a valid SMTP address entry. (See "Adding POP Mail Users to the System Distribution Directory" on page 289 for details.)

Problems with the POP server are generally associated with improperly set up entries in the SDD or improperly set up clients. Ensure that all of your SDD entries are correct and follow the instructions for setup that come with your POP client.

Note: Remember, only clients that support a POP connection can be connected to the POP server.

Other things to check:

1. Make sure that the POP server jobs are running in subsystem QSYSWRK. For the standard IP, Client Access IP and IPX jobs you can determine this by using the Work with Active Jobs (WRKACTJOB) command, specifying the QSYSWRK subsystem on the AS/400 system:

```
WRKACTJOB SBS(QSYSWRK)
```

The POP server jobs start with the prefix "QTPO", and are:

- QTPOP — Standard POP IP server connection
- QTPOC — Client Access POP IP server connection
- QTPOI — Client Access POP IPX/SPX server connection
- QTPOABCH — Client Access address book connection

POP server SNA jobs can be found by using the WRKJOB JOB(QTPOSNA) command. Look for jobs listed as job name QTPOSNA with a status of ACTIVE to verify that the POP server jobs are available for a SNA connection.

2. Make sure the QSYSWRK subsystem is up and running.
3. If the POP server is not running, you can start it using the Start TCP/IP Servers command (STRTCPSVR *POP).
4. Verify that the POP server is listening for a client connection.
 - a. If you are using an IP or Client Access IP connection, use the Network Status command (NETSTAT *CNN). Look for an entry for either port 110 or pop3 if you are using a standard IP connection or as-pop3 if you have a Client Access server connection.
 - b. If you are using an IPX connection, use the WRKIPXSTS OPTION(*SELECT) command with option 4 to display IPX connection status. You should see a LISTEN state connection. If you display details you will see an associated user profile of QTCP. This indicates that the POP server is listening on the IPX port.
 - c. Make sure the STMP and QMSF jobs are running in subsystem QSYSWRK. If not, you can use the STRTCPSVR (STRTCPSVR *SMTP) and Start Mail Server Framework (STRMSF) commands to start them.

Problem Determination Flows

Use the following flow chart to isolate the problem if other efforts have been unsuccessful. The cause lists that follow identify potential problems.

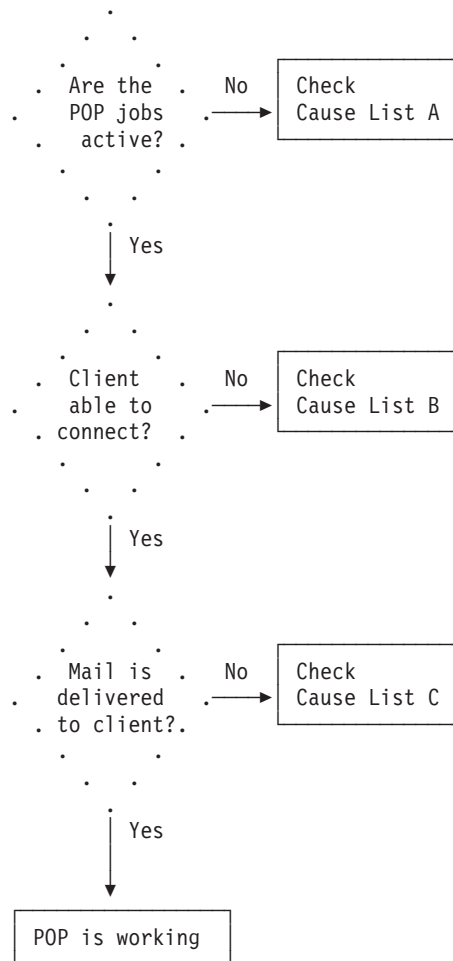


Figure 279. POP Problem Analysis

Cause List A

1. Check to see if subsystem QSYSWRK is running.
 - a. If you are using IP, Client Access IP or IPX connections, use the Work with Active Jobs (WRKACTJOB) command. Type WRKACTJOB SBS(QSYSWRK) to see the display. POP server job names start with the prefix "QTPO":
 - QTPOP — Standard POP IP server connection
 - QTPOC — Client Access POP IP server connection
 - QTPOI — Client Access POP IPX/SPX server connection
 - QTPOABCH — Client Access address book connection
 - b. If you are running Client Access SNA servers, use the WRKJOB JOB(QTPOSNA) command to verify that there are QTPOSNA jobs in the ACTIVE state.
2. Try to start the POP server jobs by using the STRTCPSVR *POP command.

3. Verify that the TCP/IP Connectivity Utilities for AS/400 product has been properly installed on the system.
4. Look at job logs by typing “WRKSPLF QTCP” and look for any job logs with a job name having the prefix “QTPO.” Correct any problems indicated in these job logs.

One potential problem is that the system storage threshold has been exceeded. Verify that this has not occurred.

Cause List B

1. If using a Client Access connection, verify the following have been properly installed on the system:
 - SS1 product, Host Servers
 - XA1 product, Client Access Family - Base
 - XD1 product, Client Access for Windows 95/NT
2. Verify that TCP/IP is up and running.
3. Use the NETSTAT *CNN command to verify that there is an entry in the local port list for port 110 or “pop3.”
 - a. If using a Client Access IP connection, use the NETSTAT *CNN command to verify that the as-pop3 entry exists in the local port list.
 - b. If using a Client Access connection, use the NETSTAT *CNN command to verify that the following host server connections exist in the local port list:
 - as-signon
 - as-cent
 - as-svrmap
 - c. Start the Host Server using the STRHOSTSVR *ALL command. Look for jobs that start with “QZ” in subsystem QSYSWRK to verify that host server jobs are active.
4. If using a Client Access IPX/SPX connection, verify that an IPX job exists in the QSYSWRK subsystem. Use the STRIPX command to start the IPX job.
 - Verify that the client can “talk” to the server. Use the PING command to verify the path from the client to the server, and the server to the client.
5. Restart the POP server using the ENDTCPSVR *POP and STRTCPSVR *POP commands.
6. Verify that the client setup is correct. The user ID and password that the client uses for the connection must be the same as the user profile and password on the AS/400. Also make sure your client supports the POP protocol.
7. Verify that the user profile is enabled and that the password has not expired.
8. Ensure that the entry for the client in the system distribution directory is correct. Each POP user needs to have a Mail service level of *System message store* and a Preferred address of *SMTP name*. Each POP user must also have an SMTP address defined for them.

Cause List C

1. Verify that SMTP and QMSF are up and running. If not, use the STRTCPSVR *SMTP and STRMSF commands to start them.
2. Make sure that the client entries in the system distribution directory are correct.
3. Look for any mail server framework (MSF) job logs using the WRKSPLF QMSF command. Verify the timestamps of the logs and correct any problems described.

- If the mail is not being delivered to OfficeVision users, ensure that the SNADS subsystem is running. Use WRKACTJOB SBS(QSNADS) to look for jobs running in this subsystem. To start the SNADS subsystem use the STRSBS SBS(QSNADS) command.

Determining Problems with the Workstation Gateway Server

You can use the Work with Active Jobs (WRKACTJOB) command to check on the status of server jobs, as follows:

```
WRKACTJOB SBS(QSYSWRK)
```

When the server is not active, the Work with Active Jobs display might be similar to Figure 280:

```

                                Work with Active Jobs                                RS002
                                                                                   04/29/96 15:06:43
CPU %:      .0    Elapsed time:  00:00:00    Active jobs:  157

Type options, press Enter.
  2=Change  3=Hold  4=End  5=Work with  6=Release  7=Display message
  8=Work with spooled files  13=Disconnect ...

Opt  Subsystem/Job  User      Type  CPU %  Function      Status
   QTWSG83808     QTMTWSG   BCH     .0     SELW
   QTWSG83853     QTMTWSG   BCH     .0     TIMW

                                                                                   Bottom

Parameters or command
====>

F3=Exit      F5=Refresh  F10=Restart statistics  F11=Display elapsed data
F12=Cancel   F23=More options  F24=More keys

```

Figure 280. WRKACTJOB SBS(QSYSWRK) — Display 2

When the workstation gateway server is active, this display will show that there are at least two workstation gateway server (WSG) jobs running: one in status TIMW and one in SELW. There will be only one job in the TIMW status. There may be several jobs in the SELW status depending on the number of clients per server that you have configured for WSG.

To find out if server jobs have ended abnormally, check for spooled files owned by the QTMTWSG user profile.

First Failure Data Capture (FFDC)

Irrecoverable errors attempt to log FFDC information to simplify customer intervention in collecting debug data. Two types of FFDC symptom records can be created: RCxxxxyyy and MSGxxxxxxx. The RC-type string occurs at known locations in the source code, and usually indicates a problem with the installation

settings or required system objects. The MSG-type strings occur on unexpected errors. Since these are not anticipated, they usually occur because of unanticipated circumstances or customer environments not seen during development and test. The exact location in the code for MSG-type errors must be determined from other information.

As part of the goal to provide quick turnaround for error isolation, the server takes the additional step of dumping the job log (DMPJOBLOG) and job information (DMPJOB) to spooled files on all FFDC errors. Therefore, two spooled files will be generated for these conditions. These dumps are done at the time of error, and are done by the signal handler function. The primary reason for doing this is to get relevant information immediately following the error, before the job is ended.

The actual spooled file dump commands are:

```
QSYS/DSPJOBLOG JOB(*) OUTPUT(*PRINT)
QSYS/DSPJOB OUTPUT(*PRINT)
```

These spooled files are owned by the QTMTWSG user profile.

If a problem is detected when using the AS/400 WSG server, use the following flow chart to identify the cause after using the flow chart for general TCP/IP problems (Figure 247 on page 431). The cause lists that follow identify potential problems.

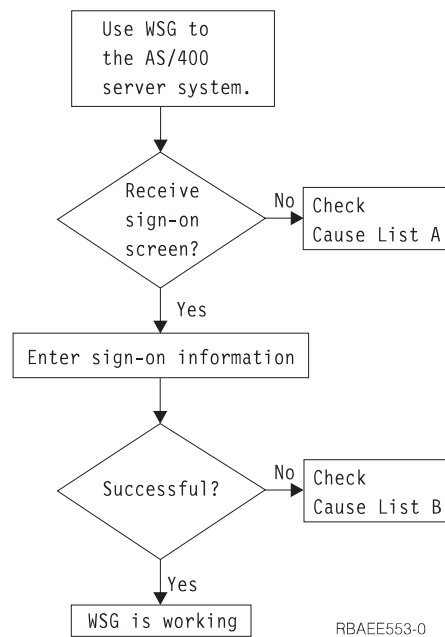


Figure 281. WSG Server Problem Analysis

Cause List A

1. Verify that the QAUTOVRT system value on the AS/400 server system is properly set to allow automatic creation of virtual devices. For example, to allow the creation of 50 virtual devices enter the command:
CHGSYSVAL SYSVAL(QAUTOVRT) VALUE(50)
2. Verify that the virtual devices on the AS/400 server system that are used by WSG are defined to a subsystem under which the interactive jobs should run. Use the Display Subsystem Description (DSPSBSD) command to see which

workstation entries are defined to a subsystem. Use the Add Work Station Entry (ADDWSE) command to define work stations to a subsystem. For example, you could use the following command to allow all workstation types to run under the QINTER subsystem:

```
ADDWSE SBS(DQINTER) WRKSTNTYPE(*ALL)
```

3. Check if the QIBM_QTMT_WSG exit point format QAPP0100 is registered. If this exit point is registered, the client may be getting rejected by the exit program.
4. If exit point QIBM_QTMT_WSG format QAPP0100 is not registered, check to see if the CHGWSGA configuration parameter DSPSGN is set to *YES.
5. If you still cannot determine the problem, set the ACCLOG parameter of CHGWSGA to *YES. This enables logging of server messages related to sign-on requests and may help identify the problem. The log file is QATMTLOG in library QUSRSYS.

Cause List B

1. Verify your authority to the virtual display device. If you receive message CPF1110 when attempting to sign on the AS/400 system, you are not authorized to the virtual display device. When the AS/400 server creates virtual devices, the QCRTAUT system value is used to determine the authority granted to user *PUBLIC. This system value should be *CHANGE to allow any user to sign on.
2. Verify that the QLMTSECOFR system value is correctly set if you are the security officer or have *SECOFR authority.

Determining Problems for DNS Server

DNS operates much the same as other TCP/IP functions and applications. Like SMTP or FTP applications, DNS jobs run under the QSYSWRK subsystem and produce job logs under the user profile QTCP with information associated with the DNS job. If a DNS job ends, you can use the job logs to determine the cause. If the DNS server is not returning the expected responses, the job logs may contain information that can help with problem analysis.

The configuration of a DNS server is not an easy task. The DNS configuration consists of several files with several different types of records in each file. The objective of this section is to help in the finding and correcting of problems of a DNS server. The section assumes that TCP/IP and Client Access Operations Navigator are already installed and functioning correctly.

Problems with the DNS server are generally the result of incorrect entries in the DNS configuration files. When a problem occurs, verify that the DNS configuration files contain the entries you expect.

Problem Determination Tools

The following features can help you troubleshoot problems with your DNS server:

- Name server lookup (nslookup)
- View DNS server active database
- Log each query the DNS server receives
- Log DNS server debug information

For details on these features, go to Managing host names (DNS) in the TCP/IP topic in the AS/400 Information Center. You can access the Information Center from the AS/400e Information Center CD-ROM, or from the following Web site: <http://www.as400.ibm.com/infocenter>

Also see Operations Navigator on-line Help for more information on the features listed above.

Problem Determination Flows

If you detect a problem when using DNS, use the following flow chart to identify the cause after using the flow chart for general TCP/IP problems. (Figure 247 on page 431). The cause lists that follow identify potential problems.

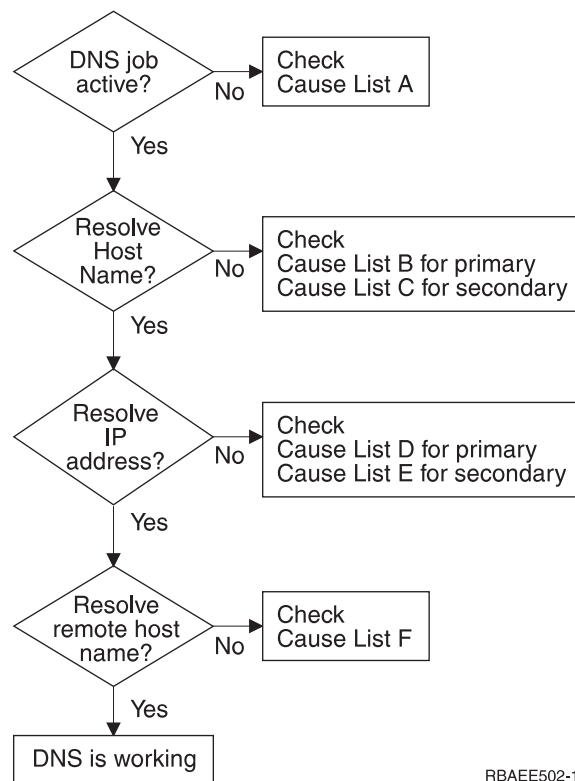


Figure 282. DNS Server Problem Analysis

Cause List A

1. Verify that you have the correct client/server software installed and that you have all necessary command and integrated file system directory and file authorities.
2. Verify the QTOBDNS job is active in the QSYSWRK subsystem. Use the command `WRKACTJOB SBS(QSYSWRK) JOB(QTOB*)` to display the active DNS jobs in the QSYSWRK subsystem. If the QTOBDNS job is not active, use Operations Navigator to start DNS.
 - a. Follow the path `\Network\Servers\OS400`
 - b. Right click on DNS Server
 - c. Select start

3. Look for DNS job logs using the command WRKSPLF QTCP. Look for job logs that have QTOBDNS listed under user data. Examine the timestamps of the logs and correct the problems described.
4. Use the command NETSTAT *CNN to verify that there are entries in the local port list for port 53 or domain.

Remote Opt	Remote Address	Remote Port	Local Port	Idle Time	State
*		*	domain	019:44:14	Listen
*		*	domain	019:40:41	*UDP

5. Use the command WRKSVRTBLE to verify there are TCP and UDP domain service entries listed for port 53.

Opt	Service	Port	Protocol
	domain	53	tcp
	domain	53	udp

6. Verify the AS/400 Host Domain Information is correct. Verify the Host name and Domain name information are correct. Verify the IP address of your primary DNS is correct and at the top of the domain name server list. Use Operations Navigator to display Host Domain Information.
 - a. Follow the path \Network\Protocols\TCP/IP
 - b. Select Host Domain Information

Cause List B

1. Verify the specified primary domain exists and that you have enabled it under Primary Domains on the DNS configuration window.
2. Verify the specified host name exists and is correct in the specified primary domain.
3. Verify the DNS server knows about the specified host name. If the specified host name has been recently created be sure to use the Operations Navigator Update Server function to make changes made to your DNS configuration known to the DNS server.
4. Look for DNS job logs using the command WRKSPLF QTCP. Look for job logs that have QTOBDNS listed under user data. Examine the timestamps of the logs and correct the problems described.
5. Look at the DNS server active job log. Examine the job log entries and correct any problems.
6. View the Active Server Database using Operations Navigator. Verify the specified host name appears in the list of information displayed.
7. Verify that your DNS server is actually receiving the given query. See the General tab listed under DNS Server Properties for information on how to collect this information.
8. Use the DNS server debug feature to log DNS debug information for use by service and product development.

Cause List C

1. Verify the specified secondary domain exists under Secondary Domains on the DNS configuration window.
2. Look for DNS job logs using the command WRKSPLF QTCP. Look for job logs that have QTOBDNS and QTOBXFER listed under user data. The QTOBXFER job logs contain information about the transfer of zone information from the primary server to the secondary server. Examine the timestamps of the logs and correct the problems described.

If an AS/400 is a primary server to your DNS secondary server look for job logs on the primary AS/400 that have QTOBXMIT listed under user data. The QTOBXMIT job logs contain information about the transfer of zone information from the primary server to the secondary server. Examine the timestamps of the logs and correct the problems described.

3. Look at the DNS server active job log. Examine the job log entries and correct any problems.
4. Verify that the primary server for this secondary server is up and running.
5. Verify that you have enabled the Save copies of master server data check box for the secondary domain. The advantage of enabling this check box is that the secondary server can function even if the primary server is down.
6. Verify the specified host name exists and is correct in the specified primary domain.
7. Check the secondary refresh interval to see how often your secondary server checks with the primary server to make sure its data is current. If you have created the specified host name recently, verify that the system has updated the secondary server with current information. Use the NSLOOKUP (STRDNSQRY-Start DNS Query) to display the secondary refresh interval defined by the primary server. Type the command:

```
NSLOOKUP
> set type=soa
> 'enter the fully qualified name of the primary domain
your DNS server uses to load its data'.
```
8. View the Active Server Database using Operations Navigator. Verify the specified host name appears in the list of the information displayed.
9. Verify that your DNS server is actually receiving the given query. Refer to the General tab listed under DNS Server Properties for information on how to collect this information.
10. Use the DNS server debug feature to log DNS debug information for use by service and product development.

Cause List D The IP to host name mapping information is found in the reverse mapping primary domains. Use Cause List B and replace forward mapping primary domain with reverse mapping primary domain and replace host name with IP address in to help determine why a specified IP address can not be resolved to a host name.

1. Verify that you have enabled the Create reverse mapping records for this host check box for the specified host. You can find the check box on the Reverse Mapping page of Host Properties for the specified host name. When this check box is enabled the an entry will be created in the reverse mapping primary domain.
2. Verify that you have enabled the Create and delete reverse mappings check box for the forward mapping primary domain. You can find the check box on the General page of Primary Domain Properties for the specified domain. When you enable this check box, the system creates an entry in the reverse mapping primary domain whenever you add a new host name to a forward mapping primary domain.

Cause List E The IP to host name mapping information is found in the reverse mapping primary domains. Use Cause List C and replace forward mapping primary domain with reverse mapping primary domain and replace host name with IP address to help determine why a specified IP address can not be resolved.

Cause List F

1. Verify that the forwarder(s) are configured **OR** that root servers are configured. If forwarders are used, make sure that 'Contact only forwarders for off-site queries' is checked.
2. Verify that the forwarder is answering queries using NSLOOKUP.
3. Verify that the root server addresses are listed and correct. To display the root server addresses, use the root servers tab for DNS Server Properties
4. Whether you are using root servers or not, you must **NEVER** list the 'local host' or the DNS's own address as a root server.
5. Look at the DNS server active job log. Examine the job log entries and correct any problems.

Determining Problems for LPR

This section discusses how the AS/400 system handles the various parameters passed to the LPR command and how they may need to be formatted to work correctly. Examples are given for some of the most common error conditions.

LPR Command Considerations

When the system passes alphabetic characters to a command processing program from an AS/400 command, it converts all of the characters to uppercase. Enclosing the parameter in apostrophes preserves the case of the characters. This is required if you use filters for the DESTOPT parameter. It may be required for the PRTQ parameter of the LPR command, depending on the case sensitivity of the destination system.

Common Error Messages

If a problem is detected when using the AS/400 LPR client, most problem analysis can be done by examining the job logs of the user and the message help text for the issued error message.

Send request failed for spooled file XYZ is the error message for both TCP3701 and TCP3719. The message help text is different, however, and should be examined. If the message is TCP3719, the error may be that the printer queue name is incorrectly spelled or the case of the printer queue name does not match that of the printer queue on the destination system.

If the message is TCP3701, look in the job log for the previous messages to determine the failure. Common failures are caused by the following:

- Destination system name is misspelled.

Note: The case of the destination system name is not important.

- Destination system name is not defined in the TCP/IP host table
- LPD server is not started on the destination system
- TCP/IP is not started on the destination system

The message help text for any previous messages should be examined to determine error recovery procedures.

Materials Required for Reporting LPR Problems

Any LPR problem reported to IBM should include the following:

- The QTCPIP and LPR client job logs.
- If file or data integrity is compromised, then any files that were being sent.
- If the file being sent is being transformed, a copy of the workstation customizing object being used.
- All options taken on the LPR command when trying to send a file.
- The type of remote host, operating system, and operating system version to which the LPR command was attempted, for example, PS/2 to OS/2, PS/2 to DOS, or RS/6000 to AIX.
- Special authorities in the user profile of the sender; the owner of the file; and the output queue parameters AUTCHK, OPRCTL, and the DSPOBJAUT to get all authorities of the user to the output queue.
- A communications trace from the time of the failure (*Format TCP/IP data only* field), formatted for ASCII. If you are not familiar with the procedure for collecting a communications trace, refer to “Collecting a Communications Trace” on page 493 and “Formatting and Saving the Communications Trace” on page 499.

Determining Problems for LPD

The most common problems, causes, and solutions for LPD are shown in Table 46.

Table 46. LPD Problem Analysis

Problem	Cause	Solution
All files on the receiving AS/400 system are of the same printer device type but not *USERASCII	The QPTMPLPD printer file is set to a type other than *USERASCII	Change the printer file (CHGPRTF command) to printer device type *USERASCII ¹

Table 46. LPD Problem Analysis (continued)

Problem	Cause	Solution
Spooled file is not on the print queue that you requested	<ul style="list-style-type: none"> • Destination print queue name is misspelled. • User did not have authority to the requested print queue. • The requested print queue is not in the user ID library list path. 	<ul style="list-style-type: none"> • Display the messages sent by LPD to check where the spooled file was placed (you must be logged on the system where the LPD server is running to see the messages). • If you are logged on with the same user ID as was used to issue the LPR command, use the Display the Message (DSPMSG) command. If you are logged on to a different user ID than was used to send the LPR command, you must use the DSPMSG userid command if the user ID exists. If the user ID does not exist, use DSPMSG QTCP/QTMLPD. QTMLPD is the default user profile used if there are any problems with the client profile. • If the messages are not able to help locate the file or the problem, then check the print queue QPRINT in library QGPL. This is the default destination where all files are spooled if there are any problems with the requested print queue.
LPD receives only a portion of the spooled file sent	The LPR job failed before sending all the data. LPD assumes that when the connection is closed, all the data has been sent.	Send the spooled file again.
Spooled files do not show up on the receiving system from a non-AS/400, even though LPR indicates it was successfully sent.	If the sending user ID does not match a user ID on the receiving system, LPD uses QTMLPD user profile to spool the file. However, if the QTMLPD user profile does not exist, or the PUBLIC authority is set to *EXCLUDE, the file is not spooled. ²	Change the authority for the QTMLPD profile to *OBJOPR, or find a system with a matching user ID to accept the spooled files.
LPD does not run on an AS/400 system with a QSECURITY system value of 50. A pointer error is received when attempting to access a user domain object in library QUSRSYS.	The QALWUSRDMN system value (Allow user domain objects in libraries) does not have QUSRSYS library specified. This value is required for the LPD servers to access the user spaces that hold print data. The LPR clients also need to access user spaces in the QTEMP library.	Change the QALWUSRDMN system value to specify the QTCP library.

Table 46. LPD Problem Analysis (continued)

Problem	Cause	Solution
<p>Notes:</p> <ol style="list-style-type: none"> 1. The QPTMPLPD printer file should be set to *USERASCII because LPD expects to receive ASCII data from non-AS/400 systems. The *USERASCII device type on the printer file does not mean that the data stream of the spooled file is ASCII. If, for example, AFPDS data is sent from a non-AS/400 system, the data is sent as *USERASCII and will not print correctly. 2. The sending system does not know that the file was not spooled because LPD does not search for a place to store the file until after it has received all the data from the sending system. 		

Materials Required for Reporting LPD Problems

Any LPD problem reported to IBM should include the following:

- Any problems that cause LPD to fail unexpectedly will generate spooled files as part of the LPD error handling. Three spooled files are created inside the failing job from these commands:

```
QSYS/DSPJOBLOG JOB(*) OUTPUT(*PRINT)
QSYS/DSPJOB OUTPUT(*PRINT)
QSYS/DMPOBJ OBJ(QUSRSYS/QTMPLPD8MM)
OBJTYPE(*USRQ)
```

These 3 files will be owned by either the sending user profile, the default user profile QTMPLPD or the QTCP user profile.

- The QTCPIP and LPD client job logs.
- If file or data integrity is compromised, then any files that are being sent.
- If the file being sent is being transformed, a copy of the workstation customizing object being used.
- If the file being received is an AS/400 spooled file sent with TRANSFORM=*YES, or is coming from a non-AS/400 client, include the description of the QUSRSYS/QPTMPLPD printer file.
- The type of remote host, operating system, and operating system version from which the LPR command was attempted, for example, PS/2 to OS/2, PS/2 to DOS, or RS/6000 to AIX.
- A communications trace from the time of the failure (format TCP/IP data only), formatted for both ASCII and EBCDIC. If you are not familiar with the procedure for collecting a communications trace, refer to “Collecting a Communications Trace” on page 493 and “Formatting and Saving the Communications Trace” on page 499.

Determining Problems with REXEC

If a problem is detected when using the REXEC server, use the following flow chart to identify the cause after using the flow chart for general TCP/IP problems (Figure 247 on page 431). The cause lists that follow identify potential problems.

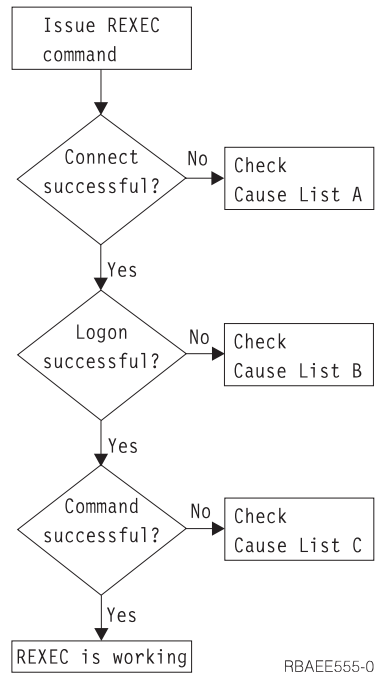


Figure 283. REXEC Server Problem Analysis

Cause List A

1. Check to see that the REXEC server is running. If not, start it with the `STRTCPSVR SERVER(*REXEC)` command.
2. If the message "Connection refused" is returned to the REXEC client, check the exit program associated with the exit point `QIBM_QTMX_SERVER_REQ`. This exit program has either specified that the connection should be rejected, returned a value that is not correct for the Allow Operation parameter, or ended abnormally. Examine the REXEC server job log for messages. Resolve any problems with the exit program and install the corrected version.

Cause List B

1. Check your user ID and password by logging on to the system. If you are unable to do so, contact the system administrator to verify that your user ID and password are correct.
2. Check the exit program associated with exit point `QIBM_QTMX_SERVER_LOGON` (if any). This exit program has either specified that the connection should be rejected, returned a value this is not correct for the Allow Operation parameter, or ended abnormally. Examine the REXEC server job log for messages.

Cause List C

1. Check for any job log messages returned to the REXEC client. Resolve any indicated problems and try the command again.
2. If the message "Command Rejected" is returned to the REXEC client, check the exit program associated with the exit point `QIBM_QTMX_SERVER_REQ`. This exit program may be specifying that the command should be rejected, returning a value this is not correct for the Allow Operation parameter, or ending abnormally. Examine the REXEC server job log for messages. Resolve any problems with the exit program and install the corrected version.

3. Verify that the correct ASCII CCSID is configured for the REXEC server. If not, set the correct CCSID with the CHGRXCA command.

Materials Required for Reporting REXEC Problems

Any REXEC problem reported to IBM should include the following:

- A communications trace from the time of the failure (Request TCP/IP data only) formatted for ASCII. If you are not familiar with the procedure for collecting a communications trace, refer to “Collecting a Communications Trace” on page 493 and “Formatting and Saving the Communications Trace” on page 499.
- If the REXEC server has logged software error data, submit this information.

Note: The system value QSFWERRLOG must be set to *LOG for software error logging to take place. If an error occurs while QSFWERRLOG is set to *NOLOG, change the value to *LOG, try to recreate the error, and submit the logged software error data. If logged software error data is submitted, there is no need to perform a trace of the REXEC server.

- The QTCPIP and any REXEC server job logs.

Getting a Copy of an REXEC Server Job Log

To have the REXEC server save job logs, see “Creating REXEC Server Spooled Job Logs” on page 403.

Tracing the REXEC Server

The REXEC server can be traced by creating a data area. Note that running the REXEC server with trace running may cause a significant performance impact.

To trace the REXEC server:

1. Create the data area using the following command: CRTDTAARA
DTAARA(QUSRSYS/QTMRXCDBG) TYPE(*LGL) LEN(1)
2. Perform the REXEC operation that you want to trace.
3. Delete the data area using the following command: DLTDTAARA
DTAARA(QUSRSYS/QTMRXCDBG)
4. Enter the following command to find the output queue:
DSPSYSVAL QPRTDEV

For example, the following display appears:

```
Display System Value
System value . . . . . : QPRTDEV
Description . . . . . : Printer device description
Printer device . . . . . : PRT01      Name
```

Figure 284. Display System Value Display

The printer device is also the name of the default system output queue.

5. Record the name of the printer device. In this example, PRT01 is the printer device.

6. Press F12 (Cancel) to return to the display where you entered the DSPSYSVAL command.
7. Type the following command:
 WRKOUTQ OUTQ(printer-device)
 Replace printer-device with the printer device recorded in the previous display. PRT01 is the output queue in this example. For example, the following display appears:

```

Work with Output Queue
Queue: PRT01      Library: QGPL      Status: RLS
Type options, press Enter.
  1=Send  2=Change  3=Hold  4=Delete  5=Display  6=Release  7=Messages
  8=Attributes  9=Work with printing status
Opt  File      User      User Data  Sts  Pages  Copies  Form Type  Pty
-   QTCPPRT    QTCP     QTMSMTP   HLD   46    1      *STD      5
-   QPSRVTRC   QSECOFR   HLD     44    1      *STD      5

```

Figure 285. Work with Output Queue Display

8. Press F18 (Bottom) to get to the bottom of the spooled file list if More... appears on the display.
9. Find the last file named QPSRVTRC with the same user as the user who was logged on the REXEC server when the trace was created.
10. Press F11 (View 2) to view the date and time of the file you want to work with.
11. Verify that you are working with the most recent spooled file, QPSRVTRC.
12. Indicate in the problem report that the trace was tried and it failed. Send whatever trace information there is with the problem report.

Tracing TCP/IP Protocol Layer Problems

If a TCP/IP protocol layer problem occurs, you can perform a Licensed Internal Code (LIC) trace of the Transmission Control Protocol (TCP/IP) component to capture the problem.

This LIC component can be traced using the system service tools (SST). You can access this service function by using the Start System Service Tools (STRSST) command. This function allows you to generate a LIC trace of all TCP/IP protocol layer traffic.

APPC Over TCP/IP Debugging Capabilities

Debugging APPC over TCP/IP problems is often a matter of doing standard debug of an APPC application. However, there are some additional problems that can occur when first attempting to run the application over a TCP/IP network.

- When an APPC application on the AS/400 attempts to open to a remote location across a TCP/IP network, there are two configuration steps that are often overlooked:
 - If the APPN remote configuration list does not have an entry for the remote location the APPC program is attempting to open to, the AS/400 attempts to find an APPN route to the remote location. This will fail with a CPF8933 message (or similar failure), stating that a route to the specified location was not found.

You need to tie this location to APPC over TCP/IP by adding the remote location to the remote configuration list. See the *Communications Configuration* book for details.

- If the host table or remote name server do not contain the location name to IP address mapping, the APPC over TCP/IP job is not able to resolve the TCP/IP routing to the remote location. A TCP4F05 message is logged as a QSYSOPR message. Additionally, the QAPPCTCP job in the QSYSWRK subsystem logs both a TCP4F04 and TCP4F05 message.

You need to tie this location to a particular IP address by adding the remote location to IP address mapping in the host table or remote name server.

- When APPC over TCP/IP is running, the QAPPCTCP job is started in the QSYSWRK subsystem, and for the most part will be in TIMW status. Also you can use the Work With Connection Status option of NETSTAT to show the APPC over TCP/IP connections. Before there are any active APPC over TCP/IP connections, you will find an APPCoverTCPIP entry for both a *UDP and Listen State connection on local port 397. These are both waiting for incoming APPC over TCP/IP traffic.

If both of these situations are not true, make sure TCP/IP is started and that the ALWANYNET (Allow AnyNet) network attribute is set to *YES.

Tracing APPC over TCP/IP Problems

When a valid APPC over TCP/IP problem occurs, you may need to perform a SLIC trace to capture the problem.

Multiprotocol Transport Network (MPTN) is an option of the Component Trace option of SLIC Trace (just as TCP/IP is). Choosing MPTN and the option beneath it called APPC over TCP/IP will take a SLIC trace of APPC over TCP/IP traffic.

Collecting a Communications Trace

A communications trace can be used to isolate errors. Communications traces can be run from SST or using the following CL commands:

- Check Communications Trace (CHKCMNTRC)
- Delete Communications Trace (DLTCMNTRC)
- End Communications Trace (ENDCMNTRC)
- Print Communications Trace (PRTCMNTRC)
- Start Communications Trace (STRCMNTRC)

You should be able to use the communications line while the communications trace is running. You should know the name of the line before starting the procedure. In order to determine the line that you need to trace for TCP/IP problems, look at the interface you defined and what line description that interface uses.

Note: Contact the security officer or system administrator to get service authority to use SST. This authority is necessary to use SST.

Planning to Set up a Trace

Before starting to work with a communications trace:

1. If you have not created the library IBMLIB or output queue IBMOUTQ, enter the following commands:

```
CRTLIB LIB(IBMLIB)
CRTOUTQ OUTQ(IBMLIB/IBMOUTQ)
```

2. Enter the following commands to add the library IBMLIB to your library list and to change the output queue for your job to output queue IBMOUTQ:

```
ADDLIB LIB IBMLIB
CHGJOB * OUTQ(IBMLIB/IBMOUTQ)
```

3. If the QTCPRT printer file does not exist on your system, then enter the following commands to create it:

```
CRTPRTF FILE(QTCP/QTCPRT) DEV(*JOB)
        RPLUNPT(*YES) SCHEDULE(*FILEEND)
        FILESEP(0) LVLCHK(*NO)
        TEXT('TCP/IP printer file')
CHGOBJOWN OBJ(QTCP/QTCPRT) OBJTYPE(*FILE)
        NEWOWN(QSYS)
```

4. Enter the following commands to send the spooled file QTCPRT containing the communications trace to the output queue IBMOUTQ in library IBMLIB:

```
OVRPRTF FILE(QTCPRT) OUTQ(IBMLIB/IBMOUTQ)
OVRPRTF FILE(QPCMPRT) TOFILE(QTCP/QTCPRT)
```

The printer file overrides are not in effect after your job ends.

5. Obtain the name of the line description associated with the TCP/IP interface with which you are having the problem or which is used by the application or network with which you are having a problem. Use NETSTAT *IFC to determine the name of the line description associated with the interface.
6. Ensure that the line is varied on and that the TCP/IP interface associated with the line has been started so that TCP/IP data can be sent and received over the interface and the line. Use NETSTAT *IFC to verify that the interface is active.

To access SST:

1. Enter STRSST on a command line. The following display appears:

System Service Tools (SST)

Select one of the following:

1. Start a service tool
2. Work with active service tools
3. Work with disk units
4. Work with diskette data recovery

Selection

F3=Exit F10=Command entry F12=Cancel

2. Select option 1 (Start a service tool). The following display is shown:

```

Start a Service Tool

Warning: Incorrect use of this service tool can cause damage
to data in this system. Contact your service representative
for assistance.

Select one of the following:

    1. Product activity log
    2. Trace Licensed Internal Code
    3. Work with communications trace
    4. Display/Alter/Dump
    5. Licensed Internal Code log
    6. Main storage dump manager
    7. Hardware service manager

Selection

F3=Exit      F12=Cancel      F16=SST menu

```

3. Select option 3 (Work with communications trace).

Starting a Communications Trace

To start a communications trace:

1. If you are instructed by the service provider to vary off the line, enter the following command; otherwise, continue with the next step.

```
VRYCFG CFGOBJ(line-name) CFGTYPE(*LIN)
STATUS(*OFF)
```

2. After selecting option 3, the following display is shown:

```

Work with Communications Traces

Type options, press Enter.
  2=Stop trace      4=Delete trace    6=Format and print trace
  7=Display message 8=Restart trace

Configuration
Opt  Object          Type  Trace Description      Protocol  Trace Status

(No active traces)

F3=Exit  F5=Refresh  F6=Start trace  F10=Change size
F11=Display buffer size  F12=Cancel

```

3. Press F6 to start a trace. The following display appears:

```

                                Start Trace

Type choices, press Enter.

Configuration object . . . . . _____
Type . . . . . 1      1=Line, 2=Network interface
                        3=Network server

Trace description . . . . . _____

Buffer size (in kilobytes) . . . . 1      1=128K, 2=256K, 3=2M, 4=4M
                                           5=6M, 6=8M, 7=16M, 8=32M
                                           9=64M

Stop on buffer full . . . . . N      Y=Yes, N=No

Data direction . . . . . 3      1=Sent, 2=Received, 3=Both

Number of bytes to trace:
Beginning bytes . . . . . *CALC  Value, *CALC, *MAX
Ending bytes . . . . . *CALC  Value, *CALC

F3=Exit  F5=Refresh  F12=Cancel

```

4. Type in the name of the line associated with the TCP/IP interface over which the problem occurs. These examples use TRNLINE as the line name.
5. Type a 6 in the *Buffer size* prompt. The display should look like this:

```

                                Start Trace

Type choices, press Enter.

Configuration object . . . . . TRNLINE
Type . . . . . 1      1=Line, 2=Network interface
                        3=Network server

Trace description . . . . . TCP/IP trace

Buffer size (in kilobytes) . . . . 6      1=128K, 2=256K, 3=2M, 4=4M
                                           5=6M, 6=8M, 7=16M, 8=32M
                                           9=64M

Stop on buffer full . . . . . N      Y=Yes, N=No

Data direction . . . . . 3      1=Sent, 2=Received, 3=Both

Number of bytes to trace:
Beginning bytes . . . . . *CALC  Value, *CALC, *MAX
Ending bytes . . . . . *CALC  Value, *CALC

F3=Exit  F5=Refresh  F12=Cancel

```

6. Press the Enter key. The following display appears:

```

Select Trace Options

Configuration object . . . . . TRNLINE
Type . . . . . LINE

Select one of the following:

1. All data (no filtering)
2. Remote controller data
3. Remote MAC address data
4. Remote SAP data
5. Local SAP data
6. IP protocol number
7. IP address data

Selection

1

F3=Exit      F5=Refresh      F12=Cancel

```

7. Press the Enter key. The following display appears:

```

Work with Communications Traces

Type options, press Enter.
2=Stop trace      4=Delete trace   6=Format and print trace
7=Display message 8=Restart trace

Configuration
Opt  Object      Type  Trace Description      Protocol  Trace Status
    TRNLINE    LINE  TCP/IP TRACE           TRN       ACTIVE

F3=Exit  F5=Refresh  F6=Start trace  F10=Change size
F11=Display buffer size  F12=Cancel

```

8. Do one of the following:

Table 47. Trace Status

If the Trace Status Is	Go to Step
Waiting	9 on page 498
Error	1 on page 499 (Formatting and Saving the Communications Trace)
Other than waiting or error	12 on page 498

- Press F3 (Exit) until you obtain the display requesting that you press Enter to continue ending SST.

Note: The trace does not stop if you exit SST while the trace is running.

- Press the Enter key to exit SST.
- Enter the following command to vary on the line:

```

VRYCFG CFGOBJ(line-name) CFGTYPE(*LIN)
        STATUS(*ON)

```

Ensure that the interface associated with the line description is started. You can use the STRTCPIFC command to start an interface.

- Enter the commands and programs that caused the problem. Trace the data on the line for a period of time specified by your service provider.
- Go to “Stopping a Communications Trace” after the error occurs.

Stopping a Communications Trace

To stop the trace:

- If you are at the Work with Communications Traces display, type a 2 in the Opt column next to the name of the line you want to stop tracing.

Note: If the status is waiting, press F5 (Refresh) to refresh the status on the Work with Communications Traces display.

- Press the Enter key. In this example, the trace status changed to stopping. The following display appears:

```

Work with Communications Traces

Type options, press Enter.
 2=Stop trace      4=Delete trace   6=Format and print trace
 7=Display message 8=Restart trace

Configuration
Opt  Object      Type  Trace Description  Protocol  Trace Status
   TRNLINE     LINE  TCP/IP TRACE      TRN       STOPPING

F3=Exit   F5=Refresh   F6=Start trace  F10=Change size
F11=Display buffer size  F12=Cancel

```

- Press F5 (Refresh) to refresh the display. In this example, the trace status changed to stopped. The following display appears:

```

Work with Communications Traces

Type options, press Enter.
 2=Stop trace      4=Delete trace   6=Format and print trace
 7=Display message 8=Restart trace

Opt  Configuration
     Object      Type  Trace Description  Protocol  Trace Status
     TRNLINE    LINE  TCP/IP TRACE      TRN       STOPPED

F3=Exit   F5=Refresh   F6=Start trace  F10=Change size
F11=Display buffer size  F12=Cancel

```

4. Continue with “Formatting and Saving the Communications Trace” to format and save the trace.

Formatting and Saving the Communications Trace

1. Type a 6 in the Option column next to the name of the stopped line or line with a trace status of error that you want to format and print.
2. Press the Enter key. For example, the following two displays appear for a token-ring protocol:

```

Format Trace Data

Configuration object . . . . . : TRNLINE
Type . . . . . : LINE

Type choices, press Enter.

Controller . . . . . *ALL      *ALL, name

Data representation . . . . . 3      1=ASCII, 2=EBCDIC, 3=*CALC

Format RR, RNR commands . . . . . N      Y=Yes, N=No
Format Broadcast data . . . . . Y      Y=Yes, N=No
Format MAC or SMT data only . . . . . N      Y=Yes, N=No
Format UI data only . . . . . N      Y=Yes, N=No
Format SNA data only . . . . . N      Y=Yes, N=No
Format TCP/IP data only . . . . . N      Y=Yes, N=No
Format IPX data only . . . . . N      Y=Yes, N=No

F3=Exit   F5=Refresh   F12=Cancel

```

Note: The format choices shown in this display are not available for every protocol.

Note: To format TCP/IP traces, you may want to consider setting the "Format Broadcast data" field to N (N=No) and the "Format TCP/IP data only" field to Y (Y=Yes).

```

                                Select IP addresses to format

Configuration object . . . . . : TRNLINE
Type . . . . . : LINE

Type choices, press Enter.

IP address . . . . . *ALL          *ALL, address
IP address . . . . . *ALL          *ALL, address
Port . . . . . *ALL          *ALL, 1-65535

F3=Exit      F5=Refresh      F12=Cancel

```

Note: The format choices shown in this display are not available for every protocol.

- For FTP, LPR, LPD, and TELNET, you should format two traces: one with a data representation for ASCII (1=ASCII) and a second one with a data representation for EBCDIC (2=EBCDIC). For SMTP and SNMP, you need only format the trace with a data representation for ASCII. For other applications you should format two traces: one with a data representation for ASCII (1=ASCII) and a second one with a data representation for EBCDIC (2=EBCDIC). The *Format TCP/IP data* field should be Y. All other fields should be N.

On the second display you may choose to format the trace showing all IP addresses or a specific IP address, including the IP address associated with a TCP/IP interface in this AS/400. An IP address must be entered in decimal dot notation, xxx.xxx.xxx.xxx, where xxx is a decimal number in the range 0-255 inclusive. If *ALL is specified for both the IP address fields, all TCP/IP frames for all IP addresses will be formatted. If specific IP addresses are specified for both the IP address fields, only frames that have source and destination, or destination and source IP addresses matching those specified will be formatted. This is useful for formatting all the frames passing between two systems. If a specific IP address and *ALL are specified in the IP address fields, only frames that have a source or destination IP address matching the IP address specified will be formatted.

Press the Enter key to format the trace data for the format choices shown. Wait for the following message to appear:

```
Format of trace data complete
```

Spooled file QTCPPRT containing the trace is created.

- Press F3 (Exit) until you exit the communications trace and the system service tools (SST).
- Press the Enter key to exit SST.

- Go to “Verifying the Contents of the Communications Trace” to display the trace.

Verifying the Contents of the Communications Trace

To verify the contents of the communications trace:

- Enter the following command.

```
WRKOUTQ OUTQ(IBMLIB/IBMOUTQ)
```

For example, the following display appears:

```

Work with Output Queue

Queue:  IBMOUTQ      Library:  IBMLIB      Status:  RLS

Type options, press Enter.
 1=Send  2=Change  3=Hold  4=Delete  5=Display  6=Release  7=Messages
 8=Attributes  9=Work with printing status

Opt  File      User      User Data  Sts  Pages  Copies  Form Type  Pty
   QTCPPRT  JSMITH
                                     RDY   34      1   *STD     5

                                                                 Bottom

Parameters for options 1, 2, 3 or command
===>
F3=Exit  F11=View 2  F12=Cancel  F20=Writers  F22=Printers
F24=More keys

```

- Press F11 (View 2) to view the date and time of the spooled file or files you want to work with. For example, the following display appears:

```

Work with Output Queue

Queue:  IBMOUTQ      Library:  IBMLIB      Status:  RLS

Type options, press Enter.
 1=Send  2=Change  3=Hold  4=Delete  5=Display  6=Release  7=Messages
 8=Attributes  9=Work with printing status

Opt  File      File Nbr  Job      User      Number  Date      Time
   QTCPPRT      1  QPADEV0009  JSMITH    006728  05/15/94  13:34:19

                                                                 Bottom

Parameters for options 1, 2, 3 or command
===>
F3=Exit  F11=View 1  F12=Cancel  F20=Writers  F22=Printers
F24=More keys

```

3. If 'More...' appears on the display and you need to continue searching for the spooled file, page forward or backward through the list of files; otherwise, continue with the next step.
4. Type a 5 in the Opt column next to the spooled file QTCPPRT that you want to display. QTCPPRT contains the communications trace.

Note: The last files contain the communications traces if you just ran the trace.

5. Press the Enter key. For example, the following display appears:

```

Display Spooled File
File . . . . . : QTCPPRT                      Page/Line  1/1
Control . . . . . :                          Columns   1 - 78
Find . . . . . :
*...+...1...+...2...+...3...+...4...+...5...+...6...+...7...+...
COMMUNICATIONS TRACE      Title: TCP/IP TRACE      04/08/97

Trace Description . . . . . : TCP/IP TRACE
Configuration object . . . . . : TRNLINE
Type . . . . . : 1                      1=Line, 2=Network Interface
                                           3=Network server

Object protocol . . . . . : TRN
Start date/Time . . . . . : 04/08/97 10:10:35.699
End date/Time . . . . . : 04/08/97 10:11:38.900
Bytes collected . . . . . : 41425
Buffer size . . . . . : 1                1=128K, 2=256K, 3=2M, 4=4M
                                           5=6M, 6=8M, 7=16M, 8=32M
                                           9=64M

Data direction . . . . . : 3                1=Sent, 2=Received, 3=Both

Stop on buffer full . . . . . : N          Y=Yes, N=No
More...

F3=Exit  F12=Cancel  F19=Left  F20=Right  F24=More keys

```

6. Verify that this is a communications trace for the line traced and that the time the trace started and ended are correct.

The trace that is collected by the service tool is automatically deleted if you perform an IPL; however, the spooled file is still saved.

If you page down to the start of the actual TCP/IP data in the communications trace you will see a display similar to the following:

```

Display Spooled File
File . . . . . : QTCPPRT          Page/Line  3/1
Control . . . . .           Columns   1 - 78
Find . . . . .
*...+...1...+...2...+...3...+...4...+...5...+...6...+...7...+...
COMMUNICATIONS TRACE      Title: TCP/IP TRACE      04/08/97

Record      Data      Record      Controller  Destination  Source
Number  S/R  Length  Timer           Name          MAC Address  MAC Address
-----  - - - - -  - - - - -  - - - - -  - - - - -  - - - - -
      7   R      537  10:10:35.79229           FFFFFFFF     C0001F02A

      Routing Info . : C63099019610
                   Frame Type : IP          TOS: NORMAL      Lentg
                   Src Addr:10.7.9.1      Dest Addr:
                   SNAP Header: 0000000800
                   IP Header  : 450002142A0A000001116B46090508010905087F

More...

F3=Exit  F12=Cancel  F19=Left  F20=Right  F24=More keys

```

Much of the IP, TCP and UDP header information is parsed and displayed behind appropriate titles. This includes the source and destination IP addresses, length of the complete IP datagram, the type of service (TOS), source and destination ports, and acknowledgment (ACK) numbers. This information should help you debug the problem that you are having with TCP/IP on this AS/400 or in the network which this AS/400 is part of.

Additional Information on TCP/IP Communications Trace

The following documentation might help you to further debug a problem.

- The V2R3 TCP/IP Redbook, *IBM AS/400 TCP/IP Configuration and Operation*, GG24-3442, contains an appendix that breaks down traces for TN5250, VT100 server, VT220 client TN3270 server, SMTP server, LPR client, and FTP server. The formatting of the TCP/IP communication traces may have changed, but for the most part the protocol that flows between the AS/400 and the remote host should be the same.
- Another book from the ITSO that would be helpful to those wanting to analyze TCP/IP communications traces would be the *TCP/IP Tutorial and Technical Overview*, GG24-3376.

Using the Product Activity Log for TCP/IP Problem Analysis

The TCP/IP LIC code creates an entry in the Product Activity Log whenever a TCP/IP datagram is discarded because of a protocol error.

For outbound TCP/IP datagrams, an example of such a protocol error is a failure to establish an X.25 connection over which the datagram was to be sent. In this case, an error is reported to the user and the outbound datagram is discarded.

Inbound datagrams cause an entry in the Product Activity Log to be created when both of these conditions are met:

- The Log Protocol Errors TCP/IP Attribute is set to *YES

- The datagram has failed one of the TCP/IP protocol validity tests specified in RFC 1122, causing the system to discard it. (**Silently discarded** means the following: Discard the received datagram without reporting an error to the originating host device.) Examples of such datagrams are those with checksums or destination addresses that are not valid.

When a datagram is discarded as described above, the IP and TCP/UDP datagrams headers are logged in the detailed data of the Product Activity Log entry. The Reference Code for these Product Activity Log entries is 7004.

Appendix A. Configuring a Physical Line for TCP/IP Communication

On the AS/400 system, communication occurs through objects called lines, controllers, and devices. The communications objects for AS/400 TCP/IP are the line descriptions, the network controller descriptions, and the network device descriptions.

TCP/IP communicates over a variety of physical line types and network interfaces (NWI). The command that defines the characteristics of the physical line connection or network interface depends on the type of communications adapter, as shown in Table 48.

Table 48. Line Types and Network Interfaces Supported by TCP/IP

Line type	Configuration command
Asynchronous	Create Line Description (Async) (CRTLINASC) See "Asynchronous Line Description Parameters" on page 152.
DDI	Create Line Description (DDI Network) (CRTLINDDI)
Ethernet	Create Line Description (Ethernet) (CRTLINETH)
Frame relay	Create Line Description (Frame Relay Network) (CRTLINFR)
Frame relay NWI using a frame relay, token ring, Ethernet, or DDI line description	The frame relay NWI is created using the Create Network Interface Frame Relay Network (CRTNWIFR) command. The line description is created using the appropriate Create Line Description command and attached to the frame relay NWI by specifying the NWI and NWIDLCL parameters.
ISDN NWI using an X.25 line description	The ISDN NWI is created using the Create Network Interface ISDN (CRTNWIISDN) command. The X.25 line is created using the Create Line X.25 (CRTLINX25) command and attached to the ISDN NWI by specifying the NWI, NWICHLTYPE, NWICHLNBR, and SWTNWILST parameters.
Point-to-Point	Create Line Description (PPP) (CRTLINPPP) See "Configuring Point-to-Point Network Connections" on page 96.
Token-ring	Create Line Description (Token-Ring Network) (CRTLINTRN)
Twinax	Create Line Description (TDLC) (CRLINTDLC)
Wireless	Create Line Description (Wireless Network) (CRTLINWLS)
X.25	Create Line Description (X.25) (CRTLINX25)

You can describe the characteristics of the communications controllers by using the *Create Controller Description (Network)* (CRTCTLNET) command or by letting the system create the controller automatically when you activate TCP/IP. You only need one network controller to describe all the systems with which you communicate over any given line description.

You can describe the characteristics of the communications devices using the by *Create Device Description (Network)* (CRTDEVNET) command or by letting the system create the device automatically when you activate TCP/IP.

If you want to change controller or device descriptions, use the *Change Controller Description (Network)* (CHGCTLNET) and *Change Device Description (Network)*

(CHGDEVNET) commands. For more information on changing controller or device descriptions, see *CL Reference (Abridged)*.

Configuration Steps

To connect any of the communications adapters listed in Table 48 on page 505 to the network, perform the following steps:

1. Create a line description (see “Creating the Line Description”).
2. Set the line description maximum frame size or SSAP maximum frame size. You must consider this value when setting the maximum transmission unit (MTU) of the TCP/IP interface (see “Setting the Maximum Transmission Unit” on page 507). This is not a required step because there are default MTU values for all line types.

Creating the Line Description

If you have already configured a physical line, this existing line can be shared between TCP/IP data and data from other protocols like SNA or OSI at the same time. There is no need for a separate physical line to support TCP/IP. If a line description does not exist for a physical IOP, you must create a new one. Use one of the commands mentioned in Table 48 on page 505 to create a line description or network interface appropriate for your communications adapter. For more information on creating line descriptions, see *LAN, Frame-Relay and ATM Support, X.25 Network Support, and Communications Configuration*. Pay particular attention to the following items when creating or changing a line description for TCP/IP communications:

- Line description name.
- Source Service Access Point (SSAP).

Line Description Name

You need the name of the line description when you configure TCP/IP on your system (see “Step 1—Configuring a Line Description” on page 30). Remember the name you choose when you create the line description, or use the *Work with Configuration Status (WRKCFGSTS)* command to find the name of an existing line.

Source Service Access Point

If the line type supports source service access points (SSAP), you must specify X'AA' as entries in the SSAP list. SSAP examples include Token-ring, Ethernet IEEE802.3, DDI, and wireless. This occurs by default when you create a new line description and leave the SSAP parameter at its default value of *SYSGEN. If you have an existing line description, use the appropriate change line description command and add X'AA' to the SSAP list.

If the Ethernet standard prompt value is *ALL or IEEE8023, then you must specify X'AA' as entries in the SSAP list. This occurs by default when you create a new line description and leave the SSAP parameter at its default value of *SYSGEN.

If the Ethernet standard prompt is *ETHV2, the system sends and receives all TCP/IP data in Ethernet Version 2 frames. You do not need to configure any additional SSAPs for TCP/IP.

Setting the Maximum Transmission Unit

The maximum transmission unit (MTU) parameter that you can enter on the *Add TCP/IP Interface* (ADDTCPIFC) command, *Add TCP/IP Route* (ADDTCPRTE) command, *Change TCP/IP Interface* (CHGTCPIFC) command, or *Change TCP/IP Route* (CHGTCPRTE) command depends on the type of line that you use. The following is a list of the maximum MTU values that you can specify, based on the line type:

Asynchronous (SLIP)	1006
DDI	4352
Ethernet 802.3	1492
Ethernet Version 2	1500
Frame relay	8177
Point-to-Point (PPP)	4096
Token ring (4 meg)	4060
Token ring (16 meg)	16388
Wireless 802.3	1492
Wireless Version 2	1500
X.25	4096

Notes:

1. TCP/IP processing uses a small part of each datagram. Therefore, the whole datagram size is unavailable for user data.
2. The value of the maximum transmission unit used by TCP/IP processing depends on the value that you specify for the route on the MTU parameter of the route or interface commands mentioned previously. It also depends on the type of physical line that you use, the maximum frame size of the network line, and the SSAP maximum frame size.

Determining the Maximum Size of Datagrams

For a communications line, specify the maximum frame size on the appropriate *Create Line Description* command. The maximum frame size is compared to the MTU value of the route or interface. TCP/IP uses the lesser of these two values to determine the maximum size of datagrams that it sends by over this line.

For example, if you specify 1024 for the MTU parameter for a route attached to a communications line and the line description contained a value of 512 for a maximum frame size, the maximum datagram size value for the route that TCP/IP uses is 512. If the line is varied *off* and you change the maximum frame size on the Token-ring line description to 1994, and then the line is varied *on*, the maximum transmission unit used for the route is reset to 1024 when the next TCP/IP operation occurs that causes a datagram to be sent.

Appendix B. TCP/IP Security

This section discusses security as it relates to the TCP/IP commands, automatic configuration, network devices, and programs. For additional information on securing TCP/IP communications, see *Tips and Tools for Securing Your AS/400*.

TCP/IP Command Security

Table 49. OS/400 TCP/IP CL Commands

Command Name
Add TCP/IP Host Table Entry (ADDTCPHTE) ¹
Add TCP/IP Interface (ADDTCPIFC) ¹
Add TCP/IP Port Entry (ADDTCPPORT) ¹
Add TCP/IP Remote System Information (ADDTCPRSI) ¹
Add TCP/IP Route (ADDTCPRTE) ¹
Change BOOTP Attributes (CHGBPA) ¹
Change TCP/IP Attributes (CHGTCPA) ¹
Change TCP/IP Domain Information (CHGTCPDMN) ¹
Change TCP/IP Host Table Entry (CHGTCPHTE) ¹
Change TCP/IP Interface (CHGTCPIFC) ¹
Change TCP/IP Route (CHGTCPRTE) ¹
Change TFTP Attributes (CHGTFTPA) ¹
Change REXEC Attributes (CHGRXCA) ¹
Change RouteD Attributes (CHGRTDA) ¹
Configure TCP/IP (CFGTCP) ^{2, 4}
Configure TCP/IP Applications (CFGTCPAPP) ^{2, 4}
Configure TCP/IP BOOTP (CFGTCPBP) ^{2, 4}
Configure TCP/IP Point-to-Point (CFGTCPPTP) ^{2, 4}
Configure TCP/IP REXEC (CFGTCPRXC) ^{2, 4}
Configure TCP/IP RouteD (CFGTCPRTD) ^{2, 4}
Convert TCP/IP Control Language (CVTTCPCL) ³
End TCP/IP (ENDTCP) ³
End TCP/IP Connection (ENDTGPCNN) ³
End TCP/IP Interface (ENDTCPIFC) ³
End TCP/IP Point-to-Point (ENDTCPPTP) ³
End TCP/IP Server (ENDTCPSVR) ³
Merge TCP/IP Host Table (MRGTCPHT) ¹
NETSTAT ^{2, 4}
PING ²
Remove TCP/IP Host Table Entry (RMVTCPHTE) ¹
Remove TCP/IP Interface (RMVTCPIFC) ¹
Remove TCP/IP Port Entry (RMVTCPPORT) ¹
Remove TCP/IP Remote System Information (RMVTCPRSI) ¹
Remove TCP/IP Route (RMVTCPRTE) ¹
Rename TCP/IP Host Table Entry (RNMTCPHTE) ¹
Start TCP/IP (STRTCP) ³
Start TCP/IP Interface (STRTCPIFC) ³
Start TCP/IP Point-to-Point (STRTCPPTP) ³
Start TCP/IP Server (STRTCPSVR) ³
Verify TCP/IP Connection (VFYTCPCNN) ²
Work with BOOTP Table (WRKBPTBL) ^{2, 4}
Work with RouteD Configuration (WRKRTDCFG) ^{2, 4}
Work with TCP/IP Network Status (WRKTCPSTS) ^{2, 4}
Work with TCP/IP Point-to-Point (WRKTCPPTP) ^{2, 4}

Table 49. OS/400 TCP/IP CL Commands (continued)

Command Name
<p>Notes:</p> <ol style="list-style-type: none"> 1. *IOSYSCFG is the special authority required by this TCP/IP CL command to use or change any object. This command is shipped with the following command object authorities: <ul style="list-style-type: none"> QSYS *ALL *PUBLIC *USE 2. This TCP/IP CL command does not require any special authority. This command is shipped with the following command object authorities: <ul style="list-style-type: none"> QSYS *ALL *PUBLIC *USE 3. This TCP/IP CL command does not require any special authority. This command is shipped with the following command object authorities: <ul style="list-style-type: none"> QSYS *ALL QPGMR *USE QSYSOPR *USE QSRVBAS *USE QSRV *USE *PUBLIC *EXCLUDE 4. This command displays a menu or a list. If a user lacks the authority required to perform the function of a menu or list option, that option is not displayed.

Table 50. TELNET CL Commands

CL Command
Change TELNET Attributes (CHGTELNA) ¹ Configure TCP/IP TELNET (CFGTCPTELN) ² Change VT Map (CHGVTMAP) ² Display VT Map (DSPVTMAP) ² Set VT Map (SETVTMAP) ² Set VT Mapping Tables (SETVTTBL) ² Start TCP/IP TELNET (STRTCPTELN) ² TELNET ²
<p>Notes:</p> <ol style="list-style-type: none"> 1. *IOSYSCFG is the special authority required by this TCP/IP CL command to use or change any object. This command is shipped with the following command object authorities: <ul style="list-style-type: none"> QSYS *ALL *PUBLIC *USE 2. This TCP/IP CL command does not require any special authority. This command is shipped with the following command object authorities: <ul style="list-style-type: none"> QSYS *ALL *PUBLIC *USE

Table 51. File Transfer Protocol CL Commands

CL Command
Change FTP Attributes (CHGFTP) ^{1,2} Configure TCP/IP FTP (CFGTCPFTP) ^{1,2} FTP ² Start TCP/IP FTP (STRTCPFTP) ²

Table 51. File Transfer Protocol CL Commands (continued)

CL Command
<p>Notes:</p> <ol style="list-style-type: none"> *IOSYSCFG is the level of authority required by this TCP/IP CL command to use or change any object. This command is shipped with the following command object authorities: <ul style="list-style-type: none"> QSYS *ALL *PUBLIC *USE

Table 52. Simple Mail Transfer Protocol CL Commands

CL Command
<p>Change SMTP Attributes (CHGSMTPA)^{1,2} Configure TCP/IP SMTP (CFGTCPSMTP)² Convert Name SMTP (CVTNAMSMTP)⁴ Work with Names for SMTP (WRKNAMSMTP)²</p> <p>Notes:</p> <ol style="list-style-type: none"> *IOSYSCFG is the level of authority required by this TCP/IP CL command to use or change any object. This command is shipped with the following command object authorities: <ul style="list-style-type: none"> QSYS *ALL *PUBLIC *USE *SECADM special authority is needed to access options 10, 11, and 12. Options 1 and 2 are displayed for all users. *SECADM special authority is required to use this command.

Table 53. POP Mail Server Commands

CL Command
<p>Change POP Mail Server Attributes (CHGPOPA)¹</p> <p>Notes:</p> <ol style="list-style-type: none"> *IOSYSCFG is the level of authority required by this TCP/IP CL command to use or change any object.

Table 54. Line Printer Daemon CL Commands

CL Command
<p>Change LPD Attributes (CHGLPDA)¹ Configure TCP/IP LPD (CFGTCPLPD)¹ Send TCP/IP Spooled File (SNDTCPSPLF)</p> <p>Notes:</p> <ol style="list-style-type: none"> *IOSYSCFG is the level of authority required by this TCP/IP CL command to use or change any object.

Table 55. Bootstrap Protocol CL Commands

CL Command
<p>Change BOOTP Attributes (CHGBPA)¹ Configure TCP/IP BOOTP (CFGTCPBP)¹ Work with BOOTP Table (WRKBPTBL)</p> <p>Notes:</p> <ol style="list-style-type: none"> *IOSYSCFG is the level of authority required by this TCP/IP CL command to use or change any object.

Table 56. Trivial File Transfer Protocol CL Commands

CL Command Change TFTP Attributes (CHGTFTP) ¹
Notes: 1. *IOSYSCFG is the level of authority required by this TCP/IP CL command to use or change any object.

Table 57. Route Daemon CL Commands

CL Command Change RouteD Attributes (CHGRTDA) ¹ Configure TCP/IP RouteD (CFGTCPRD) ¹ Work with RouteD Configuration (WRKRTDCFG)
Notes: 1. *IOSYSCFG is the level of authority required by this TCP/IP CL command to use or change any object.

Table 58. Remote Execution CL Commands

CL Command Change REXEC Attributes (CHGRXCA) ¹ Configure TCP/IP REXEC (CFGTCPRXC) ¹
Notes: 1. *IOSYSCFG is the level of authority required by this TCP/IP CL command to use or change any object.

Table 59. HTTP Server Commands

CL Command Change HTTP Attributes (CHGHTTPA) ¹ Configure TCP/IP HTTP (CFGTCPHTTP) Work with HTTP Configuration (WRKHTTPCFG)
Notes: 1. *IOSYSCFG is the level of authority required by this TCP/IP CL command to use or change any object.

Table 60. Workstation Gateway Server commands

CL Command Change Workstation Gateway Server Attributes (CHGWSGA)
Notes: 1. *IOSYSCFG is the level of authority required by this TCP/IP CL command to use or change any object.

Authority can be changed using the RVKOBJAUT command and the GRTOBJAUT command. To remove the authority of the QSYSOPR user profile to the ENDTCPNN command, the following example command may be used:

```
RVKOBJAUT OBJ(ENDTCPNN)  
  OBJTYPE(*CMD) USER(QSYSOPR)  
  AUT(*USE)
```

To give another user authority to the ENDTCPNN command, the GRTOBJAUT command can be used as follows:

```
GRTOBJAUT OBJ(ENDTCPNN)  
  OBJTYPE(*CMD) USER(RUNEB)  
  AUT(*USE)
```

These commands can be used in the same way to revoke or grant user authority to almost any object on AS/400. To be able to view which users have authority to an object, use the EDTOBJAUT command. This command can be used by a user with all rights to the object, to revoke and grant user authority to the object from a single display.

Object Security for Network Configuration

TCP/IP uses the information in the line description associated with a TCP/IP interface to determine which communications line to use. The line description can also be used to identify the attached network controller and the network device objects that the TCP/IP protocol uses when the interface is started. To view or change the line description associated with a TCP/IP interface, use option 1 (Work with TCP/IP interfaces) from the Configure TCP/IP (CFGTCP) command.

If the network controller and network device have not been created previously, TCP/IP uses automatic configuration support to create them when it starts an interface that uses the line. If the network controller or device are created by automatic configuration, *PUBLIC will be granted *CHANGE authority to these objects. The objects are not secure.

Note: TCP/IP is an OS/400 system service and will function correctly no matter what authority you specify for the network configuration objects it uses. Use the GRTOBJAUT command to change the authority to access a communications line configuration object.

The QTCPIP job that runs in the QSYSWRK subsystem will lock the network device when any interface that uses that network device is started. This is done to prevent the line from being varied off while a TCP/IP interface is using it. To remove the lock held by the QTCPIP job, do one of the following:

- Use the End TCP/IP Interface (ENDTCPIFC) command to end all interfaces that are using the line.
- Use the End TCP/IP (ENDTCP) command to end all TCP/IP processing. This ends the QTCPIP job, which will end all TCP/IP interfaces.

IBM-Written Programs Security

The IBM applications shipped with TCP/IP carry out the following security features.

File Transfer Protocol (FTP)

Requires the user to provide a user ID and password (if a secure system). FTP also verifies that a user profile has authority to any file that is to be transferred. You access this function through the AS/400 Start TCP/IP FTP (STRTCPFTP) command or by connecting to the AS/400 FTP server using another system's FTP client.

The FTP protocol definition provides no way to encrypt password information.

Note: There are three exit points provided with AS/400 FTP that allow you to set up security and validation controls. They are described in "Appendix E. TCP/IP Application Exit Points and Programs" on page 535.

Line Printer Requester (LPR)

Verifies that the user has authority to the spooled file to be sent. This function is accessed through the AS/400 Send TCP/IP Spooled File (SNDTCPSPLF) command.

Remote Execution (REXEC) Server

On secured systems REXEC requires the user to provide a user ID and password. The REXEC protocol definition does not provide a way to encrypt password information.

Note: There are two exit points provided with the AS/400 REXEC server that allow you to set up security and validation controls. They are described in “Appendix E. TCP/IP Application Exit Points and Programs” on page 535.

Packet Internet Groper (PING)

PING is an internet control message protocol (ICMP) function that allows you to verify the connection to another system without accessing the data on that system. You access this function through the AS/400 Verify TCP/IP Connection (VFYTCPCNN) command.

Simple Mail Transfer Protocol (SMTP)

Is not accessible by user-written programs. SMTP is only accessible through SNADS, thus mail is always in a known format.

TELNET

Requires the user to provide a user ID and password (if a secure system).

The AS/400 TELNET server application includes two exit points that allow you to hook into TELNET's sign-on and termination logic. You can use the AS/400 WRKREGINF (Work with Registration Information) or ADDEXITPGM (Add Exit Program) commands to associate your custom exit program to an exit point. The exit points are:

- QIBM_QTG_DEVINIT
- QIBM_QTG_DEVTERM

Customer-Written Programs Security

AS/400 TCP/IP provides a program interface to the TCP and UDP layers. Use the Revoke Object Authority (RVKOBJAUT) command to control who can access these layers. QTMTTCINT is the name of the program object in library QTCP that is shipped with the TCP/IP licensed program. You can revoke authority to this object to secure the user interface to TCP/IP. Then, you can use the Grant Object Authority (GRTOBJAUT) command to give specific user profiles access to the user interface to TCP/IP.

User-Supplied Mapping Tables

User-defined mapping tables specified on the attributes command for a particular application should be created with public authority of *USE. The following list identifies the attributes commands:

- Change TELNET Attributes (CHGTELNA)
- Change FTP Attributes (CHGFTP)
- Change SMTP Attributes (CHGSMTP)
- Change POP Mail Server Attributes (CHGPOPA)
- Change LPD Attributes (CHGLPDA)
- Change HTTP Attributes (CHGHTTTPA)
- Change Workstation Gateway Server Attributes (CHGWWSGA)

Appendix C. Mapping Tables Associated with TCP/IP Function

A **mapping table** is an object that contains a set of hexadecimal characters used to map data from one character set and code page to another. For example, unprintable characters can be mapped to blanks, and lowercase alphabetic characters can be mapped to uppercase characters. The system-recognized identifier for the object type is *TBL.

This appendix:

- Identifies which command you use to specify your mapping table
- Discusses creating, reading, and changing mapping tables
 - ASCII and EBCDIC when using TELNET and FTP
 - ASCII line drawing character set when using TELNET
 - 3270 when using TELNET
- Shows the ASCII and EBCDIC mapping tables used by TELNET and FTP previous to Version 2 Release 1 Modification 1

The default character mapping between ASCII and EBCDIC and between EBCDIC and ASCII for TCP/IP was changed at Version 2 Release 1 Modification 1 to make the mapping for some special characters consistent across the TCP/IP applications on systems using the English language character set/code page of 101/37.

TCP/IP uses ASCII-to-EBCDIC and EBCDIC-to-ASCII character mapping tables during TELNET, FTP and SMTP processing. Previous to Version 2 Release 1 Modification 1, the character mapping tables used for FTP and TELNET on English language systems were QTCPASC and QTCPEBC, which were found in the QUSRSYS library. The QTCPASC mapping table was used for the EBCDIC-to-ASCII character mapping. The QTCPEBC mapping table was used for the ASCII-to-EBCDIC character mapping. Beginning in Version 2 Release 1.1, the default character mapping for FTP and TELNET on English language systems was changed to be consistent with the SMTP application. The default character mapping is changed on English language systems only; the default character mapping remains unchanged on other systems.

Note: Message CPX8416 is used to determine the EBCDIC-to-ASCII character mapping and the ASCII-to-EBCDIC character mapping on the AS/400 system.

National Language Support Mapping

The subject of national language support (NLS) mapping with coded character set identifiers (CCSIDs) is discussed in the *International Application Development*. Many TCP/IP functions now allow the use of CCSIDs.

Using CCSID support is preferable to using mapping tables. IBM recommends that you use CCSID support when it is available.

Summary of Mapping Tables

Table 61 shows the mapping tables associated with the AS/400 TCP/IP function and identifies which command you use to specify these mapping tables.

Table 61. Summary Mapping Table Specification

AS/400 TCP/IP Function	Mapping Tables	Specified in
FTP client	ASCII/EBCDIC	FTP command
FTP server	ASCII/EBCDIC	CHGFTP command
SMTP client	ASCII/EBCDIC	CHGSMTP command
SMTP server	ASCII/EBCDIC	CHGSMTP command
TELNET client 5250 full-screen mode	none	
TELNET server 5250 full-screen mode	none	
TELNET client 3270 full-screen mode	3270 table	TELNET command
TELNET server 3270 full-screen mode	none	
TELNET server line mode	ASCII/EBCDIC can be used	Handled in user program
TELNET client VTxxx full-screen mode	ASCII/EBCDIC	TELNET command
TELNET server VTxxx full-screen mode	ASCII/EBCDIC	CHGTELNA command or the SETVTTLBL command

Creating ASCII and EBCDIC Mapping Tables

Two tables are required for data processing in TELNET and FTP:

- One to map incoming data (user data and protocol information) *from* the remote system character set and code page (ASCII-to-EBCDIC mapping)
- One to map outgoing data (user data and protocol information) *to* the remote system character set and code page (EBCDIC-to-ASCII mapping).

You can use the mapping tables supplied with the system (see “EBCDIC and ASCII Character Sets” on page 524) or create your own tables.

Note: For FTP, only the contents of the file are mapped; the file name is not mapped.

To create a user-defined mapping table, you must do the following:

1. Use the Start Source Entry Utility (STRSEU) command to create a source member containing the hexadecimal values you want included in the mapping table. Each source member must contain 512 hexadecimal characters because each mapping table contains 256 bytes of data.
2. Use the Create Table (CRTTBL) command to create the mapping table from the source member.
3. Use the Grant Object Authority (GRTOBJAUT) command to grant user QTCP *USE authority to the mapping table.

Note: You must grant this authority.

See the *CL Programming* for additional information about creating and editing tables.

Creating a Source Member for Incoming Data

The following example shows the syntax for creating the source member for an incoming mapping table. It will map data from ASCII to EBCDIC. You specify the incoming mapping table by entering the TBLASCIN value on the SRCMBR parameter:

```
STRSEU SRCFILE(QGPL/QTBLSRC)
      SRCMBR(TBLASCIN) TYPE(TXT)
```

The following format shows an example of source member data for an incoming mapping table.

```
*****BEGINNING OF TBLASCIN DATA*****
00010203372D2E2F1605250B0C0D0E0F101112133C3D322618193F27221D351F
405A7F7B5B6C507D4D5D5C4E6B604B61F0F1F2F3F4F5F6F7F8F97A5E4C7E6E6F
7CC1C2C3C4C5C6C7C8C9D1D2D3D4D5D6D7D8D9E2E3E4E5E6E7E8E9ADE0BD5F6D
79818283848586878889919293949596979899A2A3A4A5A6A7A8A9C04FD0A107
00010203372D2E2F1605250B0C0D0E0F101112133C3D322618193F27221D351F
405A7F7B5B6C507D4D5D5C4E6B604B61F0F1F2F3F4F5F6F7F8F97A5E4C7E6E6F
7CC1C2C3C4C5C6C7C8C9D1D2D3D4D5D6D7D8D9E2E3E4E5E6E7E8E9ADE0BD5F6D
79818283848586878889919293949596979899A2A3A4A5A6A7A8A9C04FD0A107
*****END OF DATA*****
```

Figure 286. Example of Incoming Data (TBLASCIN) to the Client System

Creating a Source Member for Outgoing Data

The following example shows the syntax for creating the source member for an outgoing mapping table. It will map data from EBCDIC to ASCII. You specify the outgoing mapping table by entering the TBLASCOU value on the SRCMBR parameter:

```
STRSEU SRCFILE(QGPL/QTBLSRC)
      SRCMBR(TBLASCOU) TYPE(TXT)
```

The following format shows an example of source member data for an outgoing mapping table.

```
*****BEGINNING OF TBLASCOU DATA*****
000102031A091A7F1A1A1A0B0C0D0E0F101112131A1A081A18191A1A1C1D1E1F
1A1A1C1A1A0A171B1A1A1A1A1A0506071A1A161A1A1E1A041A1A1A1A14151A1A
20A6E180EB909FE2AB8B9B2E3C282B7C26A9AA9CDBA599E3A89E21242A293B5E
2D2FDFDC9ADDDE989DACBA2C255F3E3FD78894B0B1B2FCD6FB603A2340273D22
F861626364656667686996A4F3AFAEC58C6A6B6C6D6E6F7071729787CE93F1FE
C87E73747576777879AEFC0DA5BF2F9B5B6FDB7B8B9E6BBBCBD8DD9BF5DD8C4
7B414243444546474849CBCABEE8ECED7D4A4B4C4D4E4F505152A1ADF5F4A38F
5CE7535455565758595AA0858EE9E4D130313233343536373839B3F7F0FAA7FF
*****END OF DATA*****
```

Figure 287. Example of Outgoing Data (TBLASCOU) from the Client System

Creating a Mapping Table

After you have created the source members, you can create the mapping tables from the source by using the CRTTBL command. The following example shows how to create the incoming mapping table from the source member in the QGPL/QTBLSRC file:

```
CRTTBL TBL(TBLASCIN) SRCFILE(QGPL/QTBLSRC)
GRTOBJAUT OBJ(QGPL/TBLASCIN) OBJTYPE(*TBL)
USER(QTCP) AUT(*USE)
```

Once the table has been created, you receive this message indicating the operation was successful: Table TBLASCIN in library QGPL created.

To create an outgoing mapping table you specify the TBLASCOUT keyword for the TBL parameter:

```
CRTTBL TBL(TBLASCOUT) SRCFILE(QGPL/QTBLSRC)
GRTOBJAUT OBJ(QGPL/TBLASCOUT) OBJTYPE(*TBL)
USER(QTCP) AUT(*USE)
```

Once the table has been created, you receive this message indicating the operation was successful: Table TBLASCOUT in library QGPL created.

Specifying User-Defined ASCII and EBCDIC Mapping Tables

You can use user-defined mapping tables for TELNET, FTP or SMTP by specifying the mapping tables in the appropriate parameter values.

- For the TELNET client, the parameter values of TBLASCIN and TBLASCOUT must be specified when you start TELNET:

```
STRTCPTELN RMTSYS(remote system name)
TBLVTIN(TBLASCIN)
TBLVTOUT(TBLASCOUT)
```

- For the VT100 TELNET server, the parameters TBLVTIN and TBLVTOUT must be specified when you use the CHGTELNA command:

```
CHGTELNA TBLVTIN(TBLASCIN)
TBLVTOUT(TBLASCOUT)
```

- For the FTP client, the parameter values of TBLFTPIN and TBLFTPOUT must be specified when you start FTP:

```
STRTCPFTP RMTSYS(remote system name)
TBLFTPIN(TBLASCIN)
TBLFTPOUT(TBLASCOUT)
```

- For the FTP server, you can specify parameter values on the CHGFTP command for the FTP server mapping tables as shown below:

```
CHGFTP TBLFTPIN(TBLASCIN)
TBLFTPOUT(TBLASCOUT)
```

- For SMTP, beware that OfficeVision expects data to be in code page 500. SMTP maps from ASCII to EBCDIC and EBCDIC to ASCII in code page 500.

OfficeVision maps data from code page 500 to the code page of the AS/400 system.

- For SMTP, the parameter values of TBLSMTPIN, and TBLSMTPOUT must be specified when you use the CHGSMTP command:

```
CHGSMTP TBLSMTPIN(TBLASCIN)
TBLSMTPOUT(TBLASCOUT)
```

Notes:

1. Mapping tables specified using the CHGSMTPA command are not in effect until the next time the QSYSWRK subsystem is started.
2. If you are using user-defined mapping tables, unpredictable results can occur because the FTP, SMTP, and TELNET protocol commands are ASCII characters and are mapped using the user-defined mapping tables.

Creating 3270 Mapping Tables

Two tables can be used for TCP/IP TELNET's 3270 full-screen mode: when the server system language is different from the local AS/400 language.

- One to map the incoming data from the server system character set and code page to the character set and code page of the local AS/400 5250 terminal.
- One to map the outgoing data from the local AS/400 5250 terminal character set and code page to the server system character set and code page.

The incoming 3270 mapping table specifies a table that maps data from the character set and code page of the server system to the character set and code page of the 5250 terminal from which the user is using TELNET.

The outgoing 3270 mapping table specifies a table that maps data from the character set and code page of the 5250 terminal to the character set and code page of the server system.

To create a user-defined mapping table, you must do the following:

- Use the Start Source Entry Utility (STRSEU) command to create a source member containing the hexadecimal values you want included in the mapping table. Each source member must contain 512 hexadecimal characters because each mapping table contains 256 bytes of data.
- Use the Create Table (CRTTBL) command to create the mapping table from the source member.

See the *CL Programming* for additional information about creating and editing tables.

Notes:

1. Not all programming considerations or techniques are illustrated in the examples shown in this section. You should review the example before you begin application design and coding.
2. The examples of incoming and outgoing data are based on the U.S. basic keyboard with the KBDTYPE(USB) parameter.

Creating a Source Member for Incoming Data

The following example shows the syntax for creating the source member for an incoming mapping table. This table will map data from 3270 data streams to 5250 data streams. You specify the incoming data by entering the TBLIN value on the SRCMBR parameter:

```
STRSEU SRCFILE(QGPL/QTBLSRC)
      SRCMBR(TBLIN) TYPE(TXT)
```

The following format shows an example of source member data for an incoming mapping table.

```
****BEGINNING OF TBLIN DATA****
00606060600560606060600C0D60606011121360156060601960601C1D1E60
6060606060606060606060606060606060606060606060606060606060606060603C60603F
404142434445464748494A4B4C4D4E4F505152535455565758595A5B5C5D5E5F
606162636465666768696A6B6C6D6E6F707172737475767778797A7B7C7D7E7F
808182838485868788898A8B8C8D8E8F909192939495969798999A9B9C9D9E9F
A0A1A2A3A4A5A6A7A8A9AAABACADAEAFB0B1B2B3B4B5B6B7B8B9BABBBCBDBEBF
C0C1C2C3C4C5C6C7C8C9CACBCCCDCECFD0D1D2D3D4D5D6D7D8D9DADBDCDDDEDF
E0E1E2E3E4E5E6E7E8E9EAEBECEDEEEFF0F1F2F3F4F5F6F7F8F9FAFBFCFDFEFF
*****END OF DATA*****
```

Figure 288. Example of Incoming Data (TBLIN) to the Host System

Creating a Source Member for Outgoing Data

The following example shows the syntax for creating the source member for an outgoing mapping table. This table will map data from 5250 data streams to 3270 data streams. You specify the outgoing data by entering the TBLOUT value on the SRCMBR parameter:

```
STRSEU SRCFILE(QGPL/QTBSRC)
      SRCMBR(TBLOUT) TYPE(TXT)
```

The following format shows an example of source member data for an outgoing mapping table.

```
****BEGINNING OF TBLOUT DATA****
000102030405060708090A0B0C0D0E0F101112131415161718191A1B1C1D1E1F
202122232425262728292A2B2C2D2E2F303132333435363738393A3B3C3D3E3F
404142434445464748494A4B4C4D4E4F505152535455565758595A5B5C5D5E5F
606162636465666768696A6B6C6D6E6F707172737475767778797A7B7C7D7E7F
808182838485868788898A8B8C8D8E8F909192939495969798999A9B9C9D9E9F
A0A1A2A3A4A5A6A7A8A9AAABACADAEAFB0B1B2B3B4B5B6B7B8B9BABBBCBDBEBF
C0C1C2C3C4C5C6C7C8C9CACBCCCDCECFD0D1D2D3D4D5D6D7D8D9DADBDCDDDEDF
E0E1E2E3E4E5E6E7E8E9EAEBECEDEEEFF0F1F2F3F4F5F6F7F8F9FAFBFCFDFEFF
*****END OF DATA*****
```

Figure 289. Example of Outgoing Data (TBLOUT) from the Host System

Creating a Mapping Table

After you have created the source members, you can create the mapping tables from the source by using the CRTTBL command. The following example shows how to create the incoming mapping table from the source member in the QGPL/QTBSRC file:

```
CRTTBL TBL(TBLIN) SRCFILE(QGPL/QTBSRC)
      GRTOBJAUT OBJ(QGPL/TBLIN) OBJTYPE(*TBL)
      USER(QTCP) AUT(*ALL)
```

Once the table has been created, you receive this message indicating the operation was successful: Table TBLIN in library QGPL created.

To create an outgoing mapping table, you specify the TBLOUT keyword for the TBL parameter:

```
CRTTBL TBL(TBLOUT) SRCFILE(QGPL/QTBLSRC)
GRTOBJAUT OBJ(QGPL/TBLOUT) OBJTYPE(*TBL)
USER(QTCP) AUT(*ALL)
```

Once the table has been created, you receive this message indicating the operation was successful: Table TBLOUT in library QGPL created.

Using Mapping Tables for 3270 Full-Screen Mode

You can use mapping tables for TCP/IP TELNET by specifying the following values for the KBDTYPE, TBL3270IN, and TBL3270OUT parameters when you start TELNET:

```
STRTCPTELN RMTSYS(remote system name)
KBDTYPE(*TRNTBL) TBL3270IN(TBLIN)
TBL3270OUT(TBLOUT)
```

Reading a Mapping Table

The hexadecimal values in each mapping table represent EBCDIC or ASCII values to which data in the data stream are converted. To find a mapping value in the table, count to the position corresponding to the hex value of the character to be converted. Several examples follow.

In Figure 286 on page 519, the mapping value of characters with a value of X'04' is X'37', as shown in the highlighted position of X'04' (row 1, columns 9 and 10). Alternatively in some cases, characters are not converted; they remain their original values.

In Figure 287 on page 519, the mapping value for characters with a value of X'0B' remains X'0B', as shown in the highlighted position of X'0B' (row 1, column 23 and 24).

In Figure 288 on page 522, the mapping value of characters with a value of X'01' is X'60', as shown in the highlighted position of X'01' (row 1, columns 3 and 4).

Alternatively, in Figure 289 on page 522, characters are not converted; they remain their original values. For example, the mapping value for characters with a value of X'00' remains X'00', as shown in the highlighted position of X'00' (row 1, column 1).

Changing a Mapping Table

To change a mapping table, you must do the following:

- Change the source member for the incoming or outgoing data by using the STRSEU command.
- Delete the old mapping table by using the Delete Table (DLTTBL) command.
- Create the new mapping table from the source member by using the CRTTBL command.

Sample Mappings

This section contains several mapping examples.

This example maps an EBCDIC capital 'E' (X'C5') being sent to the remote system to an ASCII capital 'E' (X'45') as shown by the highlighted X'C5' in row 7, columns 11 and 12. (Remember, the hexadecimal value of a character represents the position in which to find the mapping value.)

```

****BEGINNING OF TBLASCOUT DATA****
000102031A091A7F1A1A1A0B0C0D0E0F101112131A1A081A18191A1A1C1D1E1F
1A1A1C1A1A0A171B1A1A1A1A1A0506071A1A161A1A1E1A041A1A1A1A14151A1A
20A6E180EB909FE2AB8B9B2E3C282B7C26A9AA9CDBA599E3A89E21242A293B5E
2D2FDFDC9ADDDE989DACBA2C255F3E3FD78894B0B1B2FCD6FB603A2340273D22
F861626364656667686996A4F3AFAEC58C6A6B6C6D6E6F7071729787CE93F1FE
C87E737475767778797AEFC0DA5BF2F9B5B6FDB7B8B9E6BBBCBD8DD9BF5DD8C4
7B414243444546474849CBCABEE8ECED7D4A4B4C4D4E4F505152A1ADF5F4A38F
5CE7535455565758595AA0858EE9E4D130313233343536373839B3F7F0FAA7FF
*****END OF DATA*****

```

Figure 290. Sample Mapping of an Outgoing EBCDIC 'E' to ASCII 'E'

This example maps an ASCII '7' (X'37') coming from the remote system to an EBCDIC '7' (X'F7') as shown by the highlighted character in position 37 (row 2, columns 47 and 48).

```

****BEGINNING OF TBLASCIN DATA****
00010203372D2E2F1605250B0C0D0E0F101112133C3D322618193F27221D351F
405A7F7B5B6C507D4D5D5C4E6B604B61F0F1F2F3F4F5F6F7F8F97A5E4C7E6E6F
7CC1C2C3C4C5C6C7C8C9D1D2D3D4D5D6D7D8D9E2E3E4E5E6E7E8E9ADE0BD5F6D
79818283848586878889919293949596979899A2A3A4A5A6A7A8A9C04FD0A107
00010203372D2E2F1605250B0C0D0E0F101112133C3D322618193F27221D351F
405A7F7B5B6C507D4D5D5C4E6B604B61F0F1F2F3F4F5F6F7F8F97A5E4C7E6E6F
7CC1C2C3C4C5C6C7C8C9D1D2D3D4D5D6D7D8D9E2E3E4E5E6E7E8E9ADE0BD5F6D
79818283848586878889919293949596979899A2A3A4A5A6A7A8A9C04FD0A107
*****END OF DATA*****

```

Figure 291. Sample Mapping of an Incoming ASCII '7' to EBCDIC '7'

This example maps the character { from the French EBCDIC code page to the US EBCDIC code page. { is X'51' in the French EBCDIC code page and X'C0' in the US EBCDIC code page. The example maps X'51' to X'C0' as shown by the highlighted character in position X'51'.

```

000102030405060708090A0B0C0D0E0F101112131415161718191A1B1C1D1E1F
202122232425262728292A2B2C2D2E2F303132333435363738393A3B3C3D3E3F
404142437C454647E049904B4C4D4E4F50C05253D05556575859B55B5C5D5E5F
60616263646566676869DD6B6C6D6E6F707172737475767778A07AB1447D7E7F
808182838485868788898A8B8C8D8E8F4A9192939495969798999A9B9C9D9E9F
79BDA2A3A4A5A6A7A8A9AAABACADAEAFB07BB2B3B45AB6B7B8B9BABBBCA1BEFB
51C1C2C3C4C5C6C7C8C9CACBCCDCECF54D1D2D3D4D5D6D7D8D9DADBDC6ADEDF
48E1E2E3E4E5E6E7E8E9EAEBECEDEEEFF0F1F2F3F4F5F6F7F8F9FAFBFCDFEFF
*****END OF DATA*****

```

Figure 292. Sample Mapping of the French EBCDIC { to the US EBCDIC {

EBCDIC and ASCII Character Sets

The default character mapping between ASCII and EBCDIC and between EBCDIC and ASCII for TCP/IP was changed at Version 2 Release 1 Modification 1 to make the mapping for some special characters consistent across the TCP/IP applications on systems using the English language character set/code page of 101/37.

The old default character mapping tables, QTCPEBC (see Table 66 on page 528) and QTCPASC (Table 67 on page 528), remain on the AS/400 system. These mapping tables can be used by specifying the following:

- For TELNET ASCII line mode, specify QUSRSYS/QTCPASC for the TBLVTOUT parameter and QUSRSYS/QTCPEBC for the TBLVTIN parameter.
- For FTP client, specify QUSRSYS/QTCPASC for the TBLFTPOUT parameter and QUSRSYS/QTCPEBC for the TBLFTPIN parameter.
- For FTP server, specify QUSRSYS/QTCPASC for the TBLFTPOUT parameter and QUSRSYS/QTCPEBC for the TBLFTPIN parameter. The FTP server must be ended and started for this change to take effect.

TCP/IP supports both ASCII and several variations of EBCDIC character sets. The tables in this section describe the character sets and the mapping between them.

Table 62 and Table 63 describe the EBCDIC character set and code page that are shipped with the system.

Note: Table 62 and Table 63 are used only if ASCII-to-EBCDIC or EBCDIC-to-ASCII mapping tables cannot be found or generated based on the current national language. Refer to the *International Application Development* for the 037 (EBCDIC) code page.

Table 62. EBCDIC Character Set

Main Storage Bit Positions 4,5,6,7		Main Storage Bit Positions 0,1,2,3							
		0000	0001	0010	0011	0100	0101	0110	0111
	Hex	0	1	2	3	4	5	6	7
0000	0	NUL	DLE			SP	&	-	
0001	1	SOH	DC1					/	
0010	2	STX	DC2		SYN				
0011	3	ETX	DC3						
0100	4								
0101	5	HT	NL	LF					
0110	6		BS	ETB					
0111	7	DEL		ESC	EOT				
1000	8		CAN						
1001	9		EM						,
1010	A						!		:
1011	B	VT				.	\$,	#
1100	C	FF	FS		DC4	<	*	%	@
1101	D	CR	GS	ENQ	NAK	()	_	'
1110	E	SO	RS	ACK		+	;	>	=
1111	F	SI	US	BEL	SUB		^	?	"

Table 63. EBCDIC Character Set

Main Storage Bit Positions 4,5,6,7		Main Storage Bit Positions 0,1,2,3							
		1000	1001	1010	1011	1100	1101	1110	1111
	Hex	8	9	A	B	C	D	E	F
0000	0					{	}	\	0

Table 63. EBCDIC Character Set (continued)

Main Storage Bit Positions 4,5,6,7		Main Storage Bit Positions 0,1,2,3							
		1000	1001	1010	1011	1100	1101	1110	1111
	Hex	8	9	A	B	C	D	E	F
0001	1	a	j	~		A	J		1
0010	2	b	k	s		B	K	S	2
0011	3	c	l	t		C	L	T	3
0100	4	d	m	u		D	M	U	4
0101	5	e	n	v		E	N	V	5
0110	6	f	o	w		F	O	W	6
0111	7	g	p	x		G	P	X	7
1000	8	h	q	y		H	Q	Y	8
1001	9	i	r	z		I	R	Z	9
1010	A								
1011	B								
1100	C								
1101	D			[]				
1110	E								
1111	F								

USA Standard 7-Bit ASCII Character Set

Table 64 and Table 65 on page 527 show the characters defined by the USA Standard 7-bit ASCII character set. Table 65 on page 527 does not contain any characters.

Table 64. USA Standard Alphabet ASCII Character Set

Main Storage Bit Positions 4,5,6,7		Main Storage Bit Positions 0,1,2,3							
		0000	0001	0010	0011	0100	0101	0110	0111
	Hex	0	1	2	3	4	5	6	7
0000	0	NUL	DLE	SP	0	@	P	`	p
0001	1	SOH	DC1	!	1	A	Q	a	q
0010	2	STX	DC2	"	2	B	R	b	r
0011	3	ETX	DC3	#	3	C	S	c	s
0100	4	EOT	DC4	\$	4	D	T	d	t
0101	5	ENQ	NAK	%	5	E	U	e	u
0110	6	ACK	SYN	&	6	F	V	f	v
0111	7	BEL	ETB	'	7	G	W	g	w
1000	8	BS	CAN	(8	H	X	h	x
1001	9	HT	EM)	9	I	Y	i	y
1010	A	LF	SUB	*	:	J	Z	j	z
1011	B	VT	ESC	+	;	K	[k	{
1100	C	FF	FS	,	<	L	\	l	
1101	D	CR	GS	-	=	M]	m	}

Table 64. USA Standard Alphabet ASCII Character Set (continued)

Main Storage Bit Positions 4,5,6,7		Main Storage Bit Positions 0,1,2,3							
		0000	0001	0010	0011	0100	0101	0110	0111
	Hex	0	1	2	3	4	5	6	7
1110	E	SO	RS	.	>	N	^	n	~
1111	F	SI	US	/	?	O	_	o	DEL

Table 65. USA Standard Alphabet ASCII Character Set

Main Storage Bit Positions 4,5,6,7		Main Storage Bit Positions 0,1,2,3							
		1000	1001	1010	1011	1100	1101	1110	1111
	Hex	8	9	A	B	C	D	E	F
0000	0								
0001	1								
0010	2								
0011	3								
0100	4								
0101	5								
0110	6								
0111	7								
1000	8								
1001	9								
1010	A								
1011	B								
1100	C								
1101	D								
1110	E								
1111	F								

EBCDIC-to-ASCII Mapping Table

Table 66 on page 528 and Table 67 on page 528 show the hexadecimal values that may be used when mapping characters from EBCDIC to ASCII.

For example, EBCDIC uses X'82' to represent the letter b; the ASCII equivalent for the letter b, as shown in Table 66 on page 528, is X'62'.

System-supplied table object QTCPPASC in library QUSRSYS is the old default character mapping table. This table was used prior to Version 2 Release 1 Modification 1 to generate mapping tables used for TELNET and FTP. The mapping tables that are now used are built dynamically and are based on your national language.

Table 66. EBCDIC-to-ASCII Mapping

Main Storage Bit Positions 4,5,6,7		Main Storage Bit Positions 0,1,2,3							
		0000	0001	0010	0011	0100	0101	0110	0111
	Hex	0	1	2	3	4	5	6	7
0000	0	00	10	1A	1A	20	26	2D	D7
0001	1	01	11	1A	1A	A6	A9	2F	88
0010	2	02	12	1C	16	E1	AA	DF	94
0011	3	03	13	1A	1A	80	9C	DC	BO
0100	4	1A	1A	1A	1A	EB	DB	9A	B1
0101	5	09	1A	0A	1E	90	A5	DD	B2
0110	6	1A	08	17	1A	9F	99	DE	FC
0111	7	7F	1A	1B	04	E2	E3	98	D6
1000	8	1A	18	1A	1A	AB	A8	9D	FB
1001	9	1A	19	1A	1A	8B	9E	AC	60
1010	A	1A	1A	1A	1A	9B	21	BA	3A
1011	B	0B	1A	1A	1A	2E	24	2C	23
1100	C	0C	1C	1A	14	3C	2A	25	40
1101	D	0D	1D	05	15	28	29	5F	27
1110	E	0E	1E	06	1A	2B	3B	3E	3D
1111	F	0F	1F	07	1A	7C	5E	3F	22

Table 67. EBCDIC-to-ASCII Mapping

Main Storage Bit Positions 4,5,6,7		Main Storage Bit Positions 0,1,2,3							
		1000	1001	1010	1011	1100	1101	1110	1111
	Hex	8	9	A	B	C	D	E	F
0000	0	F8	8C	C8	B5	7B	7D	5C	30
0001	1	61	6A	7E	B6	41	4A	E7	31
0010	2	62	6B	73	FD	42	4B	53	32
0011	3	63	6C	74	B7	43	4C	54	33
0100	4	64	6D	75	B8	44	4D	55	34
0101	5	65	6E	76	B9	45	4E	56	35
0110	6	66	6F	77	E6	46	4F	57	36
0111	7	67	70	78	BB	47	50	58	37
1000	8	68	71	79	BC	48	51	59	38
1001	9	69	72	7A	BD	49	52	5A	39
1010	A	96	97	EF	8D	CB	A1	A0	B3
1011	B	A4	87	C0	D9	CA	AD	85	F7
1100	C	F3	CE	DA	BF	BE	F5	8E	F0
1101	D	AF	93	5B	5D	E8	F4	E9	FA
1110	E	AE	F1	F2	D8	EC	A3	E4	A7
1111	F	C5	FE	F9	C4	ED	8F	D1	FF

ASCII-to-EBCDIC Mapping Table

Table 68 and Table 69 show the hexadecimal values that may be used when mapping characters from ASCII to EBCDIC.

For example, ASCII uses X'4E' to represent the letter N; the EBCDIC equivalent for the letter N, as shown in Table 68, is X'D5'.

System-supplied table object QTCPEBC in library QUSRSYS is the old default character mapping table. This table was used prior to Version 2 Release 1 Modification 1 to generate mapping tables used for TELNET and FTP. The mapping tables that are now used are built dynamically and are based on your national language.

Table 68. ASCII-to-EBCDIC Mapping

Main Storage Bit Positions 4,5,6,7		Main Storage Bit Positions 0,1,2,3							
		0000	0001	0010	0011	0100	0101	0110	0111
	Hex	0	1	2	3	4	5	6	7
0000	0	00	10	40	FO	7C	D7	79	97
0001	1	01	11	5A	F1	C1	D8	81	98
0010	2	02	12	7F	F2	C2	D9	82	99
0011	3	03	13	7B	F3	C3	E2	83	A2
0100	4	37	3C	5B	F4	C4	E3	84	A3
0101	5	2D	3D	6C	F5	C5	E4	85	A4
0110	6	2E	32	50	F6	C6	E5	86	A5
0111	7	2F	26	7D	F7	C7	E6	87	A6
1000	8	16	18	4D	F8	C8	E7	88	A7
1001	9	05	19	5D	F9	C9	E8	89	A8
1010	A	25	3F	5C	7A	D1	E9	91	A9
1011	B	0B	27	4E	5E	D2	AD	92	C0
1100	C	0C	22	6B	4C	D3	E0	93	4F
1101	D	0D	1D	60	7E	D4	BD	94	D0
1110	E	0E	35	4B	6E	D5	5F	95	A1
1111	F	0F	1F	61	6F	D6	6D	96	07

Table 69. ASCII-to-EBCDIC Mapping

Main Storage Bit Positions 4,5,6,7		Main Storage Bit Positions 0,1,2,3							
		1000	1001	1010	1011	1100	1101	1110	1111
	Hex	8	9	A	B	C	D	E	F
0000	0	00	10	40	FO	7C	D7	79	97
0001	1	01	11	5A	F1	C1	D8	81	98
0010	2	02	12	7F	F2	C2	D9	82	99
0011	3	03	13	7B	F3	C3	E2	83	A2
0100	4	37	3C	5B	F4	C4	E3	84	A3
0101	5	2D	3D	6C	F5	C5	E4	85	A4
0110	6	2E	32	50	F6	C6	E5	86	A5

Table 69. ASCII-to-EBCDIC Mapping (continued)

Main Storage Bit Positions 4,5,6,7		Main Storage Bit Positions 0,1,2,3							
		1000	1001	1010	1011	1100	1101	1110	1111
	Hex	8	9	A	B	C	D	E	F
0111	7	2F	26	7D	F7	C7	E6	87	A6
1000	8	16	18	4D	F8	C8	E7	88	A7
1001	9	05	19	5D	F9	C9	E8	89	A8
1010	A	25	3F	5C	7A	D1	E9	91	A9
1011	B	0B	27	4E	5E	D2	AD	92	C0
1100	C	0C	22	6B	4C	D3	E0	93	4F
1101	D	0D	1D	60	7E	D4	BD	94	D0
1110	E	0E	35	4B	6E	D5	5F	95	A1
1111	F	0F	1F	61	6F	D6	6D	96	07

ASCII Line Drawing Character Set

The mapping tables that are used for the ASCII line drawing character set depend on what national language you are using. These tables are built dynamically and are based on your national language.

You can identify the table object (*TBL) using the Work with Objects command: `WRKOBJ OBJ(QUSRSYS/Q*) OBJTYPE(*TBL)`. All of the system table objects are in the QUSRSYS library. If the character set is EBCDIC to ASCII, the table object is named QxxxA0MA5K where xxx is the EBCDIC code page that you are using. If the character set is ASCII to EBCDIC, the table object is named QA5K697xxx where xxx is the EBCDIC code page that you are using. For example, on US systems, the code page value is usually 037. On non-US systems, the code page value would be a different number.

You can view this object using the Work with Tables (WRKTBL) command.

Appendix D. TELNET 3270 Keyboard Mappings

You can use the TELNET 3270 (TN3270) program to access the AS/400 system in full-screen, block mode if you are on a host that supports the TN3270 client. This includes DEC/VAX, RT-PC, RS/6000, HP/9000, Sun Work station, and a personal computer or a PS/2 system running DOS or OS/2.

When accessing the AS/400 system from a TN3270 client, you may want to customize the keyboard mapping to make the applications that you run easier to use. To accomplish this, create a simple CL program that contains the Change Keyboard Mapping (CHGKBDMAP) command with your preferred mapping specified. This CL program can be specified as the initial program (INLPGM parameter) in your user profile. The CHGKBDMAP command does not affect sessions in which you are not using 3270 TELNET.

AS/400 CL Programs for the CHGKBDMAP Command

The following OS/400 CL program can be used when using a TN3270 client to connect to an AS/400 system from a PS/2 system running DOS, OS/2, or AIX, and from an RS/6000 running AIX. This CL program provides the mappings described in Table 70 on page 532, Table 71 on page 532, and Table 72 on page 533.

```
PGM
MONMSG MSGID(CPF8701 CPF0000)
EXEC(GOTO CMDLBL(CALLCMD))
CHGKBDMAP PF1(*HELP) PF2(*HLP3270) PF3(*F3)
PF4(*F4) PF5(*F5) PF6(*F6) PF7(*DOWN)
PF8(*UP) PF9(*F9) PF10(*F10) PF11(*F11)
PF12(*F12) PF13(*F13) PF14(*F14)
PF15(*F15) PF16(*F16) PF17(*F17)
PF18(*F18) PF19(*F19) PF20(*F20)
PF21(*F21) PF22(*F22) PF23(*F23)
PF24(*F24) PA1PF1(*SYSREQ) PA1PF2(*ATTN)
PA1PF3(*CLEAR) PA1PF4(*PRINT) PA1PF5(*RESET)
CALLCMD: CALL PGM(QCMD)
ENDPGM
```

The following OS/400 CL program can be used when connecting to an AS/400 system from VTxxx terminals using the TN3270 program in WIN/TCP for VMS**. There is one item on the VAX system that needs to be customized. To do this, edit the WIN/TCP "MAP3270.;" file in the directory TWG\$TCP: [NETDIST.ETC]. In the VT100 section, change the program attention key definitions to read:

```
pa1 = "-a"; (for example, Shift-6 followed by a)
pa2 = "-p"; (for example, Shift-6 followed by p)
```

This customization needs to be done only once by the WIN/TCP administrator. The following CL program provides the mapping described in Table 73 on page 533.

```
PGM
MONMSG MSGID(CPF8701 CPF0000)
EXEC(GOTO CMDLBL(CALLCMD))
CHGKBDMAP PF1(*HELP) PF2(*HLP3270) PF3(*F3)
PF4(*F4) PF5(*F5) PF6(*F6) PF7(*DOWN)
PF8(*UP) PF9(*F9) PF10(*F10)
PF11(*F11) PF12(*F12) PF13(*F13)
PF14(*F14) PF15(*F15) PF16(*F16)
PF17(*F17) PF18(*F18) PF19(*F19) +
PF20(*SAME) PF21(*SAME) PF22(*SAME)
PF23(*SAME) PF24(*SAME) PA1PF1(*SYSREQ)
PA1PF2(*ATTN) PA1PF3(*CLEAR) PA1PF4(*PRINT)
```

```

PA1PF5(*RESET) PA1PF6(*SAME) PA1PF7(*SAME)
PA2PF2(*F22) PA2PF3(*F23) PA2PF4(*F24)
PA2PF10(*F20)
CALLCMD: CALL PGM(QCMD)
ENDPGM

```

Table 70. 5250 Keyboard Mapping from DOS TN3270 (PC/TCP)

5250 Function	
Keys	Keystrokes for DOS on the PS/2 System
F1	F1
F2	F2
F3	F3
F4	F4
F5	F5
F6	F6
F7	F7 (This is the Roll Down or Page Up key)
F8	F8 (This is the Roll Up or Page Down key)
F9	F9
F10	F10
F11	F11
F12	F12
F13	Shift_F3 (Hold down the Shift key and press F3)
F14	Shift_F4 (Hold down the Shift key and press F4)
F15	Shift_F5 (Hold down the Shift key and press F5)
F16	Shift_F6 (Hold down the Shift key and press F6)
F17	Shift_F7 (Hold down the Shift key and press F7)
F18	Shift_F8 (Hold down the Shift key and press F8)
F19	Shift_F9 (Hold down the Shift key and press F9)
F20	Shift_F10 (Hold down the Shift key and press F10)
F21	Ctrl_F1 (Hold down the Ctrl key and press F1)
F22	Ctrl_F2 (Hold down the Ctrl key and press F2)
F23	Ctrl_F3 (Hold down the Ctrl key and press F3)
F24	Ctrl_F4 (Hold down the Ctrl key and press F4)
SysReq	Alt_F1+F1 (Hold down the Alt key and press F1, release both, then press F1)
Clear	Alt_F1+F3 (Hold down the Alt key and press F1, release both, then press F3)
Print Screen	Alt_F1+F4 (Hold down the Alt key and press F1, release both, then press F4)
Error Reset	Alt_F1+F5 (Hold down the Alt key and press F1, release both, then press F4)
Field Exit	KP_+Tab (Press the Keypad "-" release, then press Tab)

Table 71. 5250 Keyboard Mapping from OS/2 TN3270 (PMANT)

5250 Function	
Keys	Keystrokes for OS/2 on the PS/2 System
F1	F1
F2	F2
F3	F3
F4	F4
F5	F5
F6	F6
F7	F7 (This is the Roll Down or Page Up key)
F8	F8 (This is the Roll Up or Page Down key)
F9	F9
F10	F10
F11	F11
F12	F12
F13	Shift_F3 (Hold down the Shift key and press F3)
F14	Shift_F4 (Hold down the Shift key and press F4)

Table 71. 5250 Keyboard Mapping from OS/2 TN3270 (PMANT) (continued)

5250 Function Keys	Keystrokes for OS/2 on the PS/2 System
F15	Shift_F5 (Hold down the Shift key and press F5)
F16	Shift_F6 (Hold down the Shift key and press F6)
F17	Shift_F7 (Hold down the Shift key and press F7)
F18	Shift_F8 (Hold down the Shift key and press F8)
F19	Shift_F9 (Hold down the Shift key and press F9)
F20	Shift_F10 (Hold down the Shift key and press F10)
F21	Ctrl_F1 (Hold down the Ctrl key and press F1)
F22	Ctrl_F2 (Hold down the Ctrl key and press F2)
F23	Ctrl_F3 (Hold down the Ctrl key and press F3)
F24	Ctrl_F4 (Hold down the Ctrl key and press F4)
SysReq	Pause+F1 (Press the Pause key, release, then press F1)
Clear	Pause+F3 (Press the Pause key, release, then press F3)
Print Screen	Pause+F4 (Press the Pause key, release, then press F4)
Error Reset	Pause+F5 (Press the Pause key, release, then press F5)
Field Exit	Ctrl_End+Tab (Hold down the Ctrl key and press the End key, release both, then press Tab)

Table 72. 5250 Keyboard Mapping from AIX TN3270

5250 Function Keys	Keystrokes for AIX on the PS/2 System
F1	F1
F2	F2
F3	F3
F4	F4
F5	F5
F6	F6
F7	F7 (This is the Roll Down or Page Up key)
F8	F8 (This is the Roll Up or Page Down key)
F9	F9
F10	F10
F11	F11
F12	F12
F13	Shift_F1 (Hold down the Shift key and press F1)
F14	Shift_F2 (Hold down the Shift key and press F2)
F15	Shift_F3 (Hold down the Shift key and press F3)
F16	Shift_F4 (Hold down the Shift key and press F4)
F17	Shift_F5 (Hold down the Shift key and press F5)
F18	Shift_F6 (Hold down the Shift key and press F6)
F19	Shift_F7 (Hold down the Shift key and press F7)
F20	Shift_F8 (Hold down the Shift key and press F8)
F21	Shift_F9 (Hold down the Shift key and press F9)
F22	Shift_F10 (Hold down the Shift key and press F10)
F23	Shift_F11 (Hold down the Shift key and press F11)
F24	Shift_F12 (Hold down the Shift key and press F12)
SysReq	Alt_F1+F1 (Hold down the Alt key and press F1, release both, then press F1)
Clear	Alt_F1+F3 (Hold down the Alt key and press F1, release both, then press F3)
Print Screen	Alt_F1+F4 (Hold down the Alt key and press F1, release both, then press F4)
Error Reset	Alt_F1+F5 (Hold down the Alt key and press F1, release both, then press F5)
Field Exit	KP_+Tab (Press the Keypad "-" release, then press Tab)

Table 73. 5250 Keyboard Mapping from VAX/MVS

5250 Function Keys	Keystrokes on the VT Terminal
F1	KP1 (Keypad 1)

Table 73. 5250 Keyboard Mapping from VAX/MVS (continued)

5250 Function	
Keys	Keystrokes on the VT Terminal
F2	KP2
F3	KP3
F4	KP4
F5	KP5
F6	KP6
F7	KP7 (This is the Roll Down or Page Up key)
F8	KP8 (This is the Roll Up or Page Down key)
F9	KP9
F10	Gold+KP0 (Press the Gold key, release, then press Keypad 0)
F11	Gold+KP1 (Press the Gold key, release, then press Keypad 1)
F12	Gold+KP2 (Press the Gold key, release, then press Keypad 2)
F13	Gold+KP3 (Press the Gold key, release, then press Keypad 3)
F14	Gold+KP4 (Press the Gold key, release, then press Keypad 4)
F15	Gold+KP5 (Press the Gold key, release, then press Keypad 5)
F16	Gold+KP6 (Press the Gold key, release, then press Keypad 6)
F17	Gold+KP7 (Press the Gold key, release, then press Keypad 7)
F18	Gold+KP8 (Press the Gold key, release, then press Keypad 8)
F19	Gold+KP9 (Press the Gold key, release, then press Keypad 9)
F20	Ctrl_p+KP0 (Hold down the Ctrl key and press p, release both, then press Keypad 0)
F21	Ctrl_p+KP1 (Hold down the Ctrl key and press p, release both, then press Keypad 1)
F22	Ctrl_p+KP2 (Hold down the Ctrl key and press p, release both, then press Keypad 2)
F23	Ctrl_p+KP3 (Hold down the Ctrl key and press p, release both, then press Keypad 3)
F24	Ctrl_p+KP4 (Hold down the Ctrl key and press p, release both, then press Keypad 4)
SysReq	Ctrl-a+KP1 (Hold down the Ctrl key and press a, release both, then press Keypad 1)
Clear	Ctrl-a (Hold down the Ctrl key and press a)
Print Screen	Ctrl-a+KP4 (Hold down the Ctrl key and press a, release both, then press Keypad 4)
Error Reset	Ctrl-a+KP5 (Hold down the Ctrl key and press a, release both, then press Keypad 5)
Field Exit	Ctrl_e+Tab (Hold down the Ctrl key and press e, release both, then press Tab)

Appendix E. TCP/IP Application Exit Points and Programs

Certain TCP/IP applications provide exit points that enable them to call customer-written exit programs. This appendix contains the following information:

- Conceptual information on TCP/IP exit points and programs
- General instructions on creating exit programs for TCP/IP applications
- Descriptions of the TCP/IP application exit point interfaces
- Specific instructions on how to prepare exit programs for each TCP/IP application exit point, with examples.

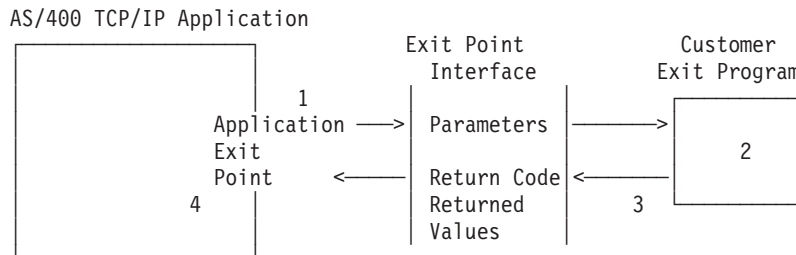
TCP/IP Exit Points and Exit Programs

An **exit point** is a specific point in the TCP/IP application program where control may be passed to an exit program. An **exit program** is a program to which the exit point passes control.

For each exit point, there is an associated programming interface, called an **exit point interface**. The exit point uses this interface to pass information between the TCP/IP application and the exit program. Each exit point has a unique name. Each exit point interface has an **exit point format name** that defines how information is passed between the TCP/IP application and the customer-written exit program.

Different exit points may share the same exit point interface. When this is the case, multiple exit points can call a single exit program.

Figure 293 shows how parameters and control are passed from the TCP/IP application program to the customer-written exit program and back again.



Processing flow:

- 1 TCP/IP application passes request parameters to the exit program
- 2 Exit program processes request parameters
- 3 Exit program returns information to the TCP/IP application
- 4 TCP/IP application performs operation based on exit program response.

Figure 293. TCP/IP Exit Point Processing

OS/400 Registration Facility

Exit points for TCP/IP applications are automatically registered when the parent product or option is installed, using the OS/400 **registration facility**. The registration facility contains a repository that allows customers to associate their exit programs with specific exit points. TCP/IP applications check the registration facility repository to determine which exit program to call for a particular exit point.

You must add your exit program to an exit point in the registration repository before a TCP/IP application can call it. Adding the exit program to the repository associates the exit program with a specific exit point.

For security exit programs, the TCP/IP application will typically request the exit program to indicate if a specified operation should be allowed. When no exit program has been added to an exit point, the TCP/IP application assumes that no additional security controls are to be applied.

You can use the Work with Registration Information (WRKREGINF) command to display a list of the exit points in the OS/400 registration facility. Use this list to display information about an exit point or to work with exit programs associated with an exit point. The Work with Registration Information display is shown in Figure 294 on page 538.

TCP/IP Application Exit Points

The following table lists the exit points provided for each TCP/IP application.

Note: If using Distributed Data Management (DDM), see the DDMACC parameter on CHGNETACMD in the *CL Reference (Abridged)* for more information.

Table 74. TCP/IP Application Exit Points

TCP/IP Application	Exit Point	Exit Point Format
FTP Client	QIBM_QTMF_CLIENT_REQ	VLRQ0100 ¹ (see page 547)
FTP Server	QIBM_QTMF_SERVER_REQ	VLRQ0100 ¹ (see page 547)
FTP Server	QIBM_QTMF_SVR_LOGON	TCPL0100 ² or TCP0200 (see page 551)
REXEC Server	QIBM_QTMX_SERVER_REQ	VLRQ0100 ¹ (see page 547)
REXEC Server	QIBM_QTMF_SVR_LOGON	TCPL0100 ² (see page 551)
REXEC Server	QIBM_QTMF_SVR_SELECT	RXCS0100 (see page 551)
TFTP Server	QIBM_QTOD_SERVER_REQ	VLRQ0100 ¹ (see page 547)
Workstation gateway (WSG) server	QIBM_QTMT_WSG	QAPP0100 (see page 569)
DHCP Server	QIBM_QTOD_DHCP_REQ	DHCV0100 ³
DHCP Server	QIBM_QTOD_DHCP_ABND	DHCA0100 ³
DHCP Server	QIBM_QTOD_DHCP_ARLS	DHCR0100 ³
TELNET Server	QIBM_QTG_DEVINIT	INIT0100 (see page 541)
TELNET Server	QIBM_QTG_DEVTERM	TERM0100 (see page 546)
Note:		
¹	The same interface format is used for request validation for the FTP client, FTP server, REXEC server, and TFTP server. This allows the use of one exit program for request validation of any combination of these applications.	
²	The same interface format is used for server log-on processing for the FTP server and REXEC server applications. This allows the use of one exit program to process log-on requests for both of these applications.	
³	For a detailed description of the DHCP exit points and how to use them, see the <i>System API Reference</i> , SC41-5801-03.	

Creating Exit Programs

There are several steps involved in designing and writing exit programs. They include:

1. Review the purpose of the exit point and the format of its interface
2. Define the scope and operation of your exit program
3. Design the exit program
4. Code the exit program
5. Add the exit program to the appropriate exit point in the registration facility. (See “Adding Your Exit Program to the Registration Facility” for instructions on how to do this.)

Note: Only users with both *SECADM and *ALLOBJ authority are allowed to add and remove TCP/IP application exit programs.

6. Test your exit program
 - Tests for each user ID
 - Tests for each operation

The most important step in establishing security exit programs is verifying that the exit program works. You must assure that the security wall works and does not have any weaknesses.

Notes:

1. If the exit program fails or returns an incorrect output parameter, the operation will not be allowed by the TCP/IP application.
2. To ensure the highest level of security, create the exit program in a library that has *PUBLIC authority of *EXCLUDE and give the exit program itself a *PUBLIC authority of *EXCLUDE. The TCP/IP application adopts authority when it is necessary to resolve and call the exit program.

Adding Your Exit Program to the Registration Facility

To add your exit program, run the Work with Registration Information (WRKREGINF) command. The following display is shown:

```

Work with Registration Information

Type options, press Enter.
  5=Display exit point  8=Work with exit programs

Exit
Opt  Exit Point      Exit Point      Registered  Text
     Point      Format
-----
QIBM_QRQ_SQL      RSQL0100    *YES      Original Remote SQL Server
QIBM_QSY_CHG_PROFILE CHGP0100    *YES      Change User Profile Exit Poin
QIBM_QSY_CRT_PROFILE CRTP0100    *YES      Create User Profile Exit Poin
QIBM_QSY_DLT_PROFILE DLTP0100    *YES      Delete User Profile Exit Poin
QIBM_QSY_DLT_PROFILE DLTPO200    *YES      Delete User Profile Exit Poin
QIBM_QSY_RST_PROFILE RSTP0100    *YES      Restore User Profile Exit Poi
QIBM_QTF_TRANSFER  TRAN0100    *YES      Original File Transfer Functi
QIBM_QTMF_CLIENT_REQ VLRQ0100    *YES      FTP Client Request Validation
QIBM_QTMF_SERVER_REQ VLRQ0100    *YES      FTP Server Request Validation
QIBM_QTMT_SVR_LOGON TCPL0100    *YES      FTP Server Logon
QIBM_QTMT_WSG      QAPP0100    *YES      WSG Server Sign-On Validation

More...

Command
===>
F3=Exit  F4=Prompt  F9=Retrieve  F12=Cancel

```

Figure 294. Work with Registration Information Display — Display 1

Step 1. Select your exit point

Type 8 next to the exit point to which you want to add an exit program. For example, to associate a program with the WSG server sign-on validation exit point, type an 8 next to this exit point, as shown.

```

QIBM_QSY_RST_PROFILE RSTP0100    *YES      Restore User Profile Exit Poi
QIBM_QTF_TRANSFER    TRAN0100    *YES      Original File Transfer Functi
QIBM_QTMF_CLIENT_REQ VLRQ0100    *YES      FTP Client Request Validation
QIBM_QTMF_SERVER_REQ VLRQ0100    *YES      FTP Server Request Validation
QIBM_QTMT_SVR_LOGON TCPL0100    *YES      FTP Server Logon
8 QIBM_QTMT_WSG      QAPP0100    *YES      WSG Server Sign-On Validation

More...

Command
===>
F3=Exit  F4=Prompt  F9=Retrieve  F12=Cancel

```

Figure 295. Work with Registration Information Display — Display 2

The Work with Exit Programs display is shown.

```

Work with Exit Programs
Exit point:  QIBM_QTMT_WSG          Format:  QAPP0100
Type options, press Enter.
  1=Add  4=Remove  5=Display  10=Replace

      Exit
      Program
Opt      Number  Exit
      Program      Library

(No exit programs found.)

```

Figure 296. Adding an Exit Program — Display 1

Step 2: Select the Add Exit Program option

Select the add option by typing a 1 (Add) in the Opt column as shown in Figure 297.

```

Work with Exit Programs
Exit point:  QIBM_QTMT_WSG          Format:  QAPP0100
Type options, press Enter.
  1=Add  4=Remove  5=Display  10=Replace

      Exit
      Program
Opt      Number  Exit
  1      Program      Library

(No exit programs found.)

```

Figure 297. Adding an Exit Program — Display 2

Step 3: Add your exit program

Fill in the exit program information as shown in Figure 298 on page 540 and Figure 299 on page 540, then press enter.

Notes:

1. You can bypass Steps 1 and 2 by using the Add Exit Program (ADDEXITPGM) command.
2. You must set the Program number parameter of the Add Exit Program (ADDEXITPGM) command to 1 when adding exit programs to FTP exit points.
3. When you add exit programs for FTP clients, these programs take effect as soon as you start additional sessions. Changes do not affect client sessions that are already running.

- When you add FTP server exit programs, end and restart the FTP servers to ensure that all servers are using the exit programs. See “Ending and Restarting FTP Server Jobs” on page 282 for instructions on how to do this.

When you add workstation gateway server exit programs, you do not need to end and restart the workstation gateway server. The WSG server checks for the exit program dynamically.

When you add REXEC server exit programs, you do not need to end and restart the REXEC server. The REXEC server checks for the exit programs dynamically.

```

Add Exit Program (ADDEXITPGM)

Type choices, press Enter.

Exit point . . . . . > QIBM_QTMT_WSG
Exit point format . . . . . > QAPP0100      Name
Program number . . . . . > 1                1-2147483647, *LOW, *HIGH
Program . . . . . > YOURPGM                Name
Library . . . . . > YOURLIB                Name, *CURLIB
Text 'description' . . . . . > 'Description of your exit program'

Additional Parameters

Replace existing entry . . . . . > *NO          *YES, *NO
Create exit point . . . . . > *NO            *YES, *NO

More...
F3=Exit  F4=Prompt  F5=Refresh  F12=Cancel  F13=How to use this display
F24=More keys

```

Figure 298. Adding an Exit Program — Display 3

```

Add Exit Program (ADDEXITPGM)

Type choices, press Enter.

Exit program data:
Coded character set ID . . . . . *JOB          Number, *NONE, *JOB
Length of data . . . . .                0-2048, *CALC
Program data . . . . .

...

```

Figure 299. Adding an Exit Program — Display 4

Removing Exit Programs

To remove an exit program from an exit point, do one of the following:

- Follow the steps for adding an exit point until the Work with Exit Programs display is shown. Select option 4 (Remove) to remove the exit program.
- Use the Remove Exit Program (RMVEXITPGM) command.

When you remove an exit program that performs a security-related operation, this operation is no longer performed. Remove security-related exit programs with caution.

TELNET Exit Points

The following sections describe information about TELNET exit points.

For additional information on TELNET exit programs as well as example code, see the AS/400 TCP/IP information at this URL:

http://www.as400.ibm.com/tstudio/tech_ref/tcpip/index.htm

For information on other AS/400 technical information, see the AS/400 Technical Studio home page at this URL:

<http://www.as400.ibm.com/techstudio>

The Technical Studio contains links to such things as workshops and technical reference, like TCP/IP and TELNET information. The TCP/IP URL listed above is located off the Technical Studio's link for Technical Reference.

The information at these URLs will be continuously updated and expanded.

Telnet Device Initialization Exit Program

The AS/400 TELNET server application includes exit points that allow you to hook into TELNET's sign-on and termination logic. You can use the AS/400 WRKREGINF (Work with Registration Information) or ADDEXITPGM (Add Exit Program) commands to associate your custom exit program to an exit point. If the TELNET server finds a program registered to one of the exit points for the server, it calls that program using parameters defined by the exit point. These parameters include things like IP address, user name, and virtual device name. Your custom exit program then processes the information, for example, logs a message and returns control to the TELNET server. On return, your exit program tells the server whether to accept or reject this client and any optional user or password overrides.

Each exit point has a name and an exit point interface. The exit point interface is a list of input and output parameters the TELNET server exchanges with your exit program. There are two exit points for the TELNET server:

- QIBM_QTG_DEVINIT
- QIBM_QTG_DEVTERM

Required Parameter Group:

1	User description information	I/O	Char(*)
2	Device description information	I/O	Char(*)
3	Connection description information	Input	Char(*)
4	Environment options	Input	Char(*)
5	Length of environment options	Input	Binary(4)
6	Allow connection	Output	Char(1)
7	Allow autosign-on	Output	Char(1)

QSYSINC Member Name: ETGDEVEX
Exit Point Name: QIBM_QTG_DEVINIT
Exit Point Format Name: INIT0100

The TELNET server will optionally provide for selecting or setting the device name to be used over the TELNET session, and allow for a TELNET client to bypass traditional device initialization. Administrators may control these new features through the use of a new exit program, which will optionally start just after client session establishment. Several parameters will be supplied to the exit program to be used in the decision process, and the exit program can set or change various parameters prior to returning to the TELNET server. You can optionally register a second exit program to start just prior to session termination. You can use this second exit program for session auditing or virtual device management.

Telnet Exit Point Format INIT0100: Required Parameter Group

User description information

I/O; CHAR(*) Information about the user that the system will use as part of the auto-signon process.

Device description information

I/O; CHAR(*) Information that the system will use to create or change the device that it uses for this TELNET session.

Connection description information

I/O; CHAR(*) Information about the client connection that the exit program can use.

Environment options

INPUT; CHAR(*) An array containing all the RFC 1572 environment options negotiated by the client. These will be in the exact format that they were in when received from the client and specified by RFC 1572. The array will, in general, consist of 1 or more pairs of environment variable names and associated values. The RFC specifies that each variable name will always be preceded by either an X'01' or X'03' depending on whether it is an RFC 1572 defined VAR, or an application specific defined USERVAR. If a value is to be associated with a VAR (or USERVAR), that value will appear next in the array preceded by the RFC 1572 defined VALUE character - X'01'. This sequence of VAR/VALUE pairs will be repeated up to a maximum of 1024 total bytes of negotiation data.

RFC 1572 and the more general TELNET negotiation RFCs also allow for control characters to appear within the VAR/USERVAR variable names or their associated values. This is allowed through the use of the ESC character X'02' and rules that apply when the ESC character itself or TELNET IAC control characters must appear in the negotiation sequence. Refer to RFC 1572 for a more complete description of control character escaping rules.

Length of environment options

INPUT; BINARY(4) The Internet Protocol (IP) address that was just released from its client host binding. This string is in dotted decimal format, left justified.

Allow connection

OUTPUT; CHAR(1) Applies to all devices and indicates to the TELNET server whether it should allow the client to connect. If the device type is DISPLAY and you have enabled auto-signon, then this client may also bypass the sign-on panel on the AS/400. The valid values are as follows:

0 Reject the request from the client

- 1 Accept the request from the client

Allow auto-signon

OUTPUT; CHAR(1) Applies to DISPLAY device types, and indicates to the TELNET server whether the auto-signon operation should be allowed to proceed for this particular client. If auto-signon is allowed, then this client can bypass the sign-on panel on the AS/400. The valid values are as follows:

- 0 Reject the application request from the client. The system will ignore the User profile, Current library, Program to call, Initial menu, and Device name output parameters.
- 1 Accept the application request from the client. The system may consider the User profile, Current library, Program to call, Initial menu, and Device Name output parameters valid if the exit program returns them.

INIT0100: Format of User Description Information

The auto-signon process will use the information about the user.

The following table shows the format of the user description information:

Table 75. Format of User Description Information

Offset		Type	Field
Dec	Hex		
0	0	INT(4)	Length of user description information
4	4	CHAR(10)	User profile
14	E	CHAR(10)	Current library
24	18	CHAR(10)	Program to call
34	22	CHAR(10)	Initial Menu

User Description Information Field Descriptions:

Current library

The name of the library that is to be made the current library if you enable the auto-signon flag. This parameter is optional, but if you supply it, you must make certain to left-justify it and pad it with blanks. Valid values are as follows:

***USRPRF**

Causes the values associated with the user profile that the system will use

library name

The name of the library that you would like the system to designate as the current library

Initial menu

The name of the initial menu to display if you have enabled the auto-signon flag. Valid values are as follows:

***USRPRF**

Causes the system to use the value associated with the user profile

menu name

The name of a menu to display

Length of user description information

Length of the user description information structure

Program to call

The name of a program that the system will call if you have enabled the auto-signon flag. This parameter is optional, but if you supply it you must left-justify it and pad it with blanks. Valid values are as follows:

*USRPRF

Causes the system to use the value associated with the user profile

program name

The name of a program that the system will start

User profile

The user profile that the system uses for the sign-on procedure if you have enabled the auto-signon flag. The system requires this parameter, and you must left-justify it and pad it with blanks.

INIT0100: Format of Device Description Information

Information that will be used to create or change the device used for this TELNET session.

The following table shows the format of the device description information, which describes the characteristics of the device to be associated with this session.

Table 76. Format of the Device Description Information

Offset		Type	Field
Dec	Hex		
0	0	CHAR(10)	Device name
10	A	CHAR(8)	Device format
18	12	CHAR(2)	Reserved
20	14	BINARY(4)	Offset to device attributes structure
24	18	BINARY(4)	Length of device attributes structure
28	1C	CHAR(*)	Device attributes structure

Device Description Information Field Descriptions:

Device name

The specific virtual device to be associated with this TELNET session. For DISPLAY devices, if the QAUTOVRT auto-create device system value allows for it, the device will be auto-created by the system if it does not already exist, and varied on. For PRINT devices, the system will auto-create the device if it does not already exist. If the exit program supplies no value, the TELNET server will default to using the traditional TELNET virtual device selection methods. Should be a valid DISPLAY or PRINT device description name and must adhere to standard OS/400 object naming conventions.

Device format

The specific virtual device type that is associated with this TELNET session. Currently only display devices that the system supports.

DSPD0100

Device is a display. The system returns display attributes.

Reserved

Reserved for future use.

Offset to device attributes structure

The offset from the start of the device description information to the start of the device attributes structure.

Length of device attributes structure

The length in the user space of the device attributes structure.

INIT0100: Format of Display Device Description Information (DSPD0100): The following table shows the format of the display device description information, which describes the characteristics of the device to be associated with this session.

Table 77. Format of Display Device Description Information (DSPD0100)

Offset		Type	Field
Dec	Hex		
0	0	CHAR(3)	Keyboard identifier
3	3	CHAR(1)	Reserved
4	4	BINARY(4)	Code page
8	8	BINARY(4)	Character set

DSPD0100 Field Descriptions:

Character set

Specifies the character set that the system is to use for this interactive job. You can find valid values in *National Language Support*. This field is identical to the Character set parameter of the Open Virtual Terminal Path QTVOPNVT API described in the *System API Reference* book.

Code page

Specifies the code page that the system is to use for this interactive job. You can find valid values in *National Language Support*. This field is identical to the Code page parameter of the Open Virtual Terminal Path QTVOPNVT API described in the *System API Reference* book.

Keyboard identifier

Specifies the 3 character keyboard identifier that the system is to use for this interactive job. The keyboard identifier implicitly specifies the code page and character set to be used, unless overridden as part of the Code page and Character set parameters. You can find valid identifiers in *National Language Support*. This field is identical to the keyboard Language type parameter of the Open Virtual Terminal Path QTVOPNVT API described in the *System API Reference* book.

Reserved

Reserved for future use

INIT0100: Format of Connection Description Information

Information about the client connection that the exit program can use.

The following table shows the format of the connection description information, which describes client and connection information for this session.

Table 78. Format of Connection Description Information

Offset		Type	Field
Dec	Hex		
0	0	INT(4)	Length of connection description information
4	4	CHAR(20)	Client internet address
24	18	CHAR(1)	Client password validated
25	19	CHAR(12)	Workstation type

Connection Description Information Field Descriptions:

Length of connection description information

Length of the connection description structure

Client internet address

This is the IP address (or type structure) of the requesting client, and is always provided to the exit program . The layout of the new fields are:

Table 79. Client IP Address Layout

Name	Size	Description
sin_len	CHAR(1)	Size of the sockaddr_in structure.
sin_family	CHAR(1)	Family or protocol. IP (Version 4) is hex 02, IPX is hex 06.
sin_port	CHAR(2)	16-bit unsigned port number.
sin_addr	CHAR(16)	4-byte unsigned

Client password validated

Specifies if TELNET validated the clients' encrypted password (if one was received). The system will set this value if Enhanced Clients or Client Access send the encrypted password for validation. The password will be checked using Client Access service functions calls. This allows the exit program to guarantee secure client sign-on process.

- 0** Client password not validated (or no password received)
- 1** Client clear-text password validated
- 2** Client encrypted password validated

Workstation type

The workstation type requested by the client, and will be one of the Internet Specifications listed in the 230 of this book; *TCP/IP Configuration and Reference*.

TELNET Device Termination Exit Program

The QIBM_QTG_DEVTERM exit point occurs when a TELNET client ends the TELNET session. This gives customers an opportunity to log session termination information and to perform device reset or cleanup operations.

The following shows the parameters for the QIBM_QTG_DEVTERM exit point.

1	Device name	Input	Char(10)
---	-------------	-------	----------

QSYSINC Member Name: NONE
Exit Point Name: QIBM_QTG_DEVTERM
Exit Point Format Name: TERM0100

Device name

The specific virtual device to be associated with this TELNET session.

The TELNET server will optionally provide for the stopping of the device, session auditing activities, and virtual device management related to the device associated with the ended TELNET session.

Required Parameter Group

Device name

Input; CHAR(10) The specific virtual device that is associated with this TELNET session.

Exit Point Interfaces for TCP/IP Application Exit Points

The exit point interfaces for TCP/IP application exit points are:

- TCP/IP application request validation exit point interface
- TCP/IP application server logon exit point interface (For information about this exit point, see *AS/400e Information Center*, SK3T-2027-01. For the Information Center URL, see “TCP/IP Topics in the Information Center” on page xv.
- TCP/IP remote execution server command processing selection exit point interface
- TCP/IP workstation gateway server sign-on exit point
- TCP/IP TELNET device initialization exit point interface
- TCP/IP TELNET device termination exit point interface

Note: For a detailed description of the DHCP exit points and how to use them, see the *System API Reference*, SC41-5801-03.

TCP/IP Application Request Validation Exit Point Interface

Required Parameter Group:

1	Application identifier	Input	Binary(4)
2	Operation identifier	Input	Binary(4)
3	User profile	Input	Char(10)
4	Remote IP address	Input	Char(*)
5	Length of remote IP address	Input	Binary(4)
6	Operation-specific information	Input	Char(*)
7	Length of operation-specific information	Input	Binary(4)
8	Allow operation	Output	Binary(4)

Exit Point Name: QIBM_QTMF_CLIENT_REQ
Exit Point Name: QIBM_QTMF_SERVER_REQ
Exit Point Name: QIBM_QTMX_SERVER_REQ
Exit Point Name: QIBM_QTOD_SERVER_REQ
Exit Point Format Name: VLRQ0100

The TCP/IP request validation exit point enables additional control for restricting an operation. Any restrictions that are imposed by the exit program are in addition to any validation that is performed by the application program, such as normal AS/400 object security. When an exit program is added to the exit point, it is called by the TCP/IP application to validate the requested action specified by the operation identifier and other input parameters in the required parameter group. The exit program sets the output parameter, Allow operation, to indicate if the TCP/IP application is to perform the operation.

Note: All character data passed to the exit program is in the coded character set ID (CCSID) of the job, or if the job CCSID is 65535, the default CCSID of the job.

Required Parameter Group

Application identifier

INPUT; BINARY(4) Identifies the application program from which the request is being made. The valid values are as follows:

- 0 FTP client program
- 1 FTP server program
- 2 REXEC server program
- 3 TFTP server program

Operation identifier

INPUT; BINARY(4) Indicates the operation that the user is attempting to perform. When the application identifier indicates the FTP client or FTP server program, the valid values are as follows:

- 0 Session initialization
- 1 Directory/library creation
- 2 Directory/library deletion
- 3 Set current directory
- 4 List files
- 5 File deletion
- 6 Sending file
- 7 Receiving file
- 8 Renaming file
- 9 Execute CL command

When the application identifier indicates the REXEC server program, valid values are as follows:

- 0 Session initialization
- 9 Perform CL command

When the application identifier indicated the TFTP server program, the valid values are as follows:

- 6 Sending file (RRQ)
- 7 Receiving file (WRQ)

User profile

INPUT; CHAR(10) The user profile under which the requested operation is run (if it is allowed).

Remote IP address

INPUT; CHAR(*) The Internet Protocol (IP) address of the remote host system. This string is in dotted decimal format, left justified. The remote host may be a client or a server based on the setting of the *application identifier* parameter.

Length of remote IP address

INPUT; BINARY(4) Indicates the length (in bytes) of the remote IP address.

Operation specific information

INPUT; CHAR(*) Information that describes the operation being attempted. The contents of this field are dependent on the value of the operation identifier.

For operation identifier 0 and application identifier 0, there is no operation-specific information. This field is blank.

For operation identifier 0 and application identifier 1, the operation-specific information contains the IP address that identifies the TCP/IP interface through which the connection to the local host (server) system is established. This string is in dotted decimal format, left justified.

For operation identifiers 1 through 3, the operation-specific information contains the name of the directory or library on which the operation is to be performed. The directory or library name is formatted as an absolute path name.

For operation identifiers 4 through 8, the operation-specific information contains the name of the file on which the operation is to be performed. The file name is formatted as an absolute path name.

For operation identifier 9, the operation-specific information contains the AS/400 Control Language (CL) command which is to be run at the user's request.

Note: See "Usage Notes" on page 550 for a summary of the operation-specific information that is required for each operation identifier.

Length of operation-specific information

INPUT; BINARY(4) Indicates the length (in bytes) of the operation-specific information, or 0 if no operation-specific information is provided.

Allow operation

OUTPUT; BINARY(4) Indicates whether the operation should be accepted or rejected. The valid values are as follows:

- 1 Never allow this operation identifier:
 - This operation identifier is to be unconditionally rejected for the remainder of the current session.
 - The exit program will not be called again for this operation identifier.
- 0 Reject the operation
- 1 Allow the operation
- 2 Always allow this operation identifier.
 - This operation identifier is to be allowed unconditionally for the remainder of the current session.
 - The exit program will not be called again with this operation identifier.

Usage Notes

For FTP, if the returned Allow operation output parameter is not valid, the FTP application will not allow the operation and the message “Data from exit program for exit point &1; is missing or not valid” will be issued to the job log.

For FTP, if any exception is encountered when calling the exit program, the FTP application will issue the message: Exception encountered for FTP exit program &1; in library &2; for exit point &3;

Two different exit points are provided for the FTP application. Exit point QIBM_QTMF_CLIENT_REQ is used to validate requests processed by the FTP client program. Exit point QIBM_QTMF_SERVER_REQ is used to validate requests processed by the FTP server program. If desired, the same exit program can be used to validate requests from both of these exit points.

Table 80 summarizes the operation-specific information required for each operation identifier.

Table 80. Application Request Validation Operation-Specific Information

Operation Identifier	Operation-Specific Information
0	NONE if application ID=0
0	Dotted decimal format IP address of client host when application ID=1 or 2
1-3	Absolute path name of library or directory /QSYS.LIB/QGPL.LIB ¹ /QOpenSys/DirA/DirAB/DirABC ²
4-8	Absolute path name of file /QSYS.LIB/MYLIB.LIB/MYFILE.FILE/MYMEMB.MBR ¹ /QOpenSys/DirA/DirAB/DirABC/FileA1 ²
9	CL command string
:	
¹	QSYS.LIB file system pathnames are always in uppercase
²	QOpenSys file system pathnames are case sensitive and may be in either upper or lower case.

Table 81 defines the FTP client and server subcommands that are associated with each operation identifier.

Table 81. FTP Client and Server Subcommands Associated with Operation Identifiers

Operation Identifier	Client Subcommands	Server Subcommands
0 - Initialize Session	OPEN	new connection ¹ on page 551
1 - Create Directory/Library		MKD, XMKD
2 - Delete directory/library		RMD, XRMD
3 - Set current directory	LCD	CWD, CDUP, XCWD, XCUP
4 - List directory/library		LIST, NLIST
5 - Delete files		DELE
6 - Send files	APPEND, PUT, MPUT ² on page 551	RETR

Table 81. FTP Client and Server Subcommands Associated with Operation Identifiers (continued)

Operation Identifier	Client Subcommands	Server Subcommands
7 - Receive files	GET, MGET ²	APPE, STOR, STOU
8 - Rename files		RNFR, RNT0
9 - Execute CL commands	SYSCMD ³	RCMD, ADDM, ADDV, CRTL, CRTP, CRTS, DLTF, DLT
:		
Notes:		
1. The exit program is called with this operation identifier each time the FTP server receives a connection request.		
2. For the MGET and MPUT subcommands, the exit program is called once for each file that is sent or retrieved.		
3. If an exit program is associated with exit point QIBM_QTMF_CLIENT_REQ, the F21 (CL command line) key is disabled and the user must use the System Command (SYSCMD) subcommand to run a CL command.		

The following notes apply to the REXEC server (application identifier 2):

1. The only valid values for the operation identifier are 0 and 9.
2. If the returned Allow operation output parameter is not valid, the REXEC server will not allow the operation and the message "Data from exit program for exit point &1 is missing or not valid" is issued to the job log.
3. If any exception is encountered when calling the exit program, the REXEC server will not allow the operation and the message "Exception encountered for REXEC exit program &1 in library &2 for exit point &3." is issued to the job log.

The following note applies to the TFTP server (application identifier 3):

1. For the TFTP server program, operation identifier 6 indicates the TFTP Read Request (RRQ) operation; operation code 7 indicates the TFTP Write Request (WRQ) operation.

TCP/IP Application Server Logon Exit Point Interface

For information about the TCP/IP application server logon exit point interface, see *AS/400e Information Center*, SK3T-2027-01. For the URL to access the Information Center online, see "TCP/IP Topics in the Information Center" on page xv.

Remote Execution Server Command Processing Selection Exit Point

The REXEC server command processing selection exit program enables you to select:

- Which command processor runs the command that the REXEC client user provides
- Whether the REXEC server converts data between ASCII and EBCDIC (for Qshell commands or spawn path names)

Required Parameter Group:

1	User profile	Input	Char(10)
2	Remote IP address	Input	Char(*)
3	Length of remote IP address	Input	Binary(4)
4	Command string	Input	Char(*)
5	Length of command string	Input	Binary(4)
6	Command processor identifier	Output	Binary(4)

7	Character conversion option	Output	Binary(4)
---	-----------------------------	--------	-----------

Exit Point Name: QIBM_QTMF_SVR_SELECT

Exit Point Format Name: RXCS0100

Note: Character data passes to the exit program in the coded character set identifier (CCSID) of the job. If the job CCSID is 65535, the server uses the default CCSID of the job.

Required Parameter Group

User profile

INPUT; CHAR(10) The user profile under which the requested operation is run.

Remote IP address

INPUT; CHAR(*) The Internet Protocol (IP) address of the REXEC client system. This string is in dotted decimal format, left justified.

Length of remote IP address

INPUT; BINARY(4) Indicates the length (in bytes) of the remote IP address.

Command string

INPUT; CHAR(*) The command to be run as specified by the REXEC client.

Length of command string

INPUT; BINARY(4) Indicates the length (in bytes) of the command string.

Command processor identifier

OUTPUT; BINARY(4) Indicates the command processor that you want the server to use for interpreting and running the command. The following values are valid:

- 0** AS/400 Control Language
The server processes the command as an AS/400 control language (CL) command. This is the default value.
- 1** Qshell command
The Qshell command interpreter processes the command. The server uses the spawn() application program interface (API) to call QShell as a child job.
- 2** Spawn path name
The server treats the command name as a path name and passes it to the spawn() application program interface (API), which runs as a child job.

Character conversion option

OUTPUT; BINARY(4) Indicates whether the REXEC server performs ASCII-EBCDIC character conversion for data that is passed on the stdin, stdout, and stderr streams. These values are valid:

- 0** Do not convert data. The server transfers all data on the stdin, stdout, and stderr streams without converting it.
- 1** Convert data.
 - The server converts data in the stdin stream from the ASCII CCSID that the CHGRXCA command specifies to the job CCSID. If the job CCSID is 65535, the server uses the default CCSID of the job.

- The server converts data in the stdout and stderr streams from the job CCSID to the ASCII CCSID that the CHGRXCA command specifies. If the job CCSID is 65535, the server uses the default CCSID of the job.

This is the default value.

Usage Notes

- If you add exit programs to both the QIBM_QTMX_SERVER_REQ and QIBM_QTMX_SVR_SELECT exit points, REXEC server first calls the exit program that you add to the QIBM_QTMX_SERVER_REQ exit point. If this program allows the operation, the server then calls the exit program that you add to the QIBM_QTMX_SVR_SELECT exit point.
- When you set the Command processor identifier parameter to 0 (AS/400 Control Language command), the conversion option is ignored. The server always performs character conversion for CL commands.
- When you set the command processor identifier to 1 (Qshell Command), the server sets these environment variables:
 - `TERMINAL_TYPE= REMOTE`
 - `PATH= /usr/bin:`
 - `LOGNAME= user` (where *user* is the user profile)
 - `HOME= homedir` (where *homedir* is the user's home directory)

If the Qshell Interpreter option of OS/400 is not installed, the REXEC client receives (in the stdout stream) a REXEC protocol diagnostic message that says "Qshell interpreter not installed".

- When you set the Command processor identifier parameter to 1 or 2:
 - The server maps the REXEC stdin, stderr, and stdout streams to file descriptors 0, 1, and 2, respectively.
 - The server sets the QIBM_USE_DESCRIPTOR_STDIO environment variable to Y.

Any other environment variables that the exit program sets are inherited by the child job.

- If you set the Command processor identifier parameter to 2 and the command string is not a valid path name for the `spawn()` API, the message "Incorrect command or path name specified" is returned to the REXEC client in the stderr stream.

File Transfer Protocol (FTP) Exit Points

AS/400 FTP provides three exit points:

- QIBM_QTMF_CLIENT_REQ — FTP Client Request Validation
- QIBM_QTMF_SERVER_REQ — FTP Server Request Validation
- QIBM_QTMF_SVR_LOGON — FTP Server Logon

You can use these exit points to set up additional security and validation controls for FTP. The FTP client and server request validation exit points are used to control the use of FTP subcommands. The server logon exit point authenticates the user who is trying to logon to the FTP server. Also, you can use the two server exit points to establish an anonymous FTP server site. (See "Anonymous FTP" on page 568.)

For information about the FTP exit point interface, see *AS/400e Information Center*, SK3T-2027-01. For the URL to access the Information Center online, see “TCP/IP Topics in the Information Center” on page xv.

Considerations and Recommendations for FTP Exit Programs

- The FTP server adopts authority when it is necessary to resolve and call the exit program. IBM strongly recommends that you create the exit program in a library with *PUBLIC authority set to *EXCLUDE, and give the exit program itself a *PUBLIC authority of *EXCLUDE.
- The various input parameters for the exit points enable you to tailor your operation validation exit program to meet your particular requirements. For example, you may restrict users to send files only to certain libraries, perform only certain system commands, and so on.
- If the FTP server at one site supports both anonymous FTP and other security restrictions, then the same exit program for each exit point must support both of these functions.

FTP Exit Program—Scenario

Figure 300 on page 555 shows an FTP client session for users that have restrictions imposed on them by exit programs. User ABC is not allowed to log on to the server. User XYZ is allowed to log on to the server, but is restricted to certain files and libraries on both the server and the client.

In this example, the server exit program does not permit user XYZ to get data from file FILEA in library LIB101 and the client exit program does not allow user XYZ to send data from file FILEC on the client system. In this way FTP exit programs may be used to restrict what files one can copy from an AS/400 system.

```
File Transfer Protocol

Previous FTP subcommands and messages:
  Connecting to host sysnam01.city.company.com at address 9.130.42.106 using
  21.
  220-QTCP at sysnam01.city.company.com.
  220 Connection will close if idle more than 5 minutes.
  OS/400 is the remote operating system. The TCP/IP version is "V3R2M0"
> abc
  331 Enter password.
  530 Log on attempt by user ABC rejected.
> user xyz
  331 Enter password.
  230 XYZ logged on.
  250 Now using naming format "0".
  257 "XYZLIB" is current library.
> cd LIB101
  250 Current library changed to LIB101.
> get FILEA.MEMBER1
  550 Request rejected.
> put FILEC.MEMBER5
  Operation not authorized.

Enter an FTP subcommand.
===> _____
_____
_____
F3=Exit      F6=Print      F9=Retrieve
F17=Top      F18=Bottom    F21=CL command line
```

Figure 300. FTP Client Session Showing Restricted Logon and Restricted FTP Operations

Sample FTP Server Logon Exit Program (C Language)

The sample program that follows provides additional control over the process of authenticating a user to a TCP/IP application server.

```

/* Module Description *****/
/*
/* Source File Name: qtmfsvrlgn.c
/*
/* Module Name: FTP Server Logon exit program.
/*
/* Service Program Name: n/a
/*
/* Source File Description:
/* This example exit program provides additional control over the
/* process of authenticating a user to a TCP/IP application server.*/
/* When installed, this example exit program would be called each
/* time a user attempts to log on to the server.
/*
/*
*****/
/* ** NOTE **
/* This material contains programming source code for your
/* consideration. These examples have not been thoroughly tested
/* under all conditions. IBM, therefore, cannot guarantee or imply
/* reliability, serviceability, performance or function of these
/* programs. All programs contained herein are provided "AS IS".
/* THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A
/* PARTICULAR PURPOSE ARE EXPRESSLY DISCLAIMED.
*****/

```

Figure 301. Sample FTP Logon Exit Program (Part 1 of 26)

```

/*
/* Function List: main - FTP Server Logon exit program main.
/* qtmfsvrlgn - FTP Server Logon exit function.
/* CheckClientAddress - Check originating sessions IP
/* address.
/*
/* End Module Description *****/
#define _QTMFSVRLGN_C

*****/
/* All file scoped includes go here
*****/
#ifndef __stdio_h
#include <stdio.h>
#endif

```

Figure 301. Sample FTP Logon Exit Program (Part 2 of 26)


```

#ifndef __ctype_h
#include <ctype.h>
#endif

#ifndef __string_h
#include <string.h>
#endif

#ifndef __stdlib_h
#include <stdlib.h>
#endif

#include "qusec.h"          /* Include for API error code structure */
#include "qsyrusri.h"      /* Include for User Information API    */

/*****
/* All file scoped Constants go here
*****/
#define EQ      ==
#define NEQ     !=
#define BLANK   ' '
#define FWIDTH  128      /* Width of one database file record */
#define FNAME   21       /* Qualified database file name width */

```

Figure 301. Sample FTP Logon Exit Program (Part 3 of 26)

```

/* Valid characters for Client IP address. The CheckClientAddress()
/* function will check the Client IP address input argument
/* (ClientIPAddr_p) to ensure it is in valid dotted-decimal format.
/* This is one example of an input validity check.
const char ValidChars[] = "0123456789.";
/*****
/* All file scoped type declarations go here
*****/

/*****
/* All file-scoped macro calls go here
*****/

/*****
/* All internal function prototypes go here
*****/
static void qtmfsvrlgn
(int,char *,int,char *,int,char *,int,int *,char *,char *,char *);

static int CheckClientAddress(char *, int);

```

Figure 301. Sample FTP Logon Exit Program (Part 4 of 26)

```

/*****/
/* All file scoped variable declarations go here */
/*****/

/*****/
/*          ** NOTE **          */
/* The following client IP address are for example purposes only. Any */
/* resemblance to actual system IP addresses is purely coincidental. */
/*****/
/* EXCLUSIVE system lists, ie - Logon attempts from any client IP */
/*          addresses NOT in one of these lists */
/*          are allowed to continue. */
/* Reject server logon attempts of users attempting to log in from */
/* these client systems (return code = 0) */
char Reject[] = "1.2.3.4 5.6.7.8";
/* Limit logon abilities of users attempting to log in as ANONYMOUS */
/* from these client systems (return code = 6). */
/* In this example program, the initial current library is set and */
/* returned as an output parameter for users attempting to log in */
/* as ANONYMOUS from these specific client systems. */

```

Figure 301. Sample FTP Logon Exit Program (Part 5 of 26)

```

char Limit[] = "9.8.7.6 4.3.2.1 8.7.6.5";

/* Function Specification *****/
/* */
/* Function Name: Main */
/* */
/* Descriptive Name: FTP Server Logon exit program main. */
/* */
/* This example exit program allows access to a TCP/IP server to */
/* be controlled by the address of the originating session, gives */
/* additional control over the initial current library to a user, */
/* and provides the capability to implement "anonymous" FTP. */
/* */
/* Notes: */
/* */
/* Dependencies: */
/* FTP Server Logon exit point QIBM_QTMF_SVR_LOGON was registered */
/* during FTP product installation. */
/* */
/* Restrictions: */
/* */
/* None */

```

Figure 301. Sample FTP Logon Exit Program (Part 6 of 26)

```

/*                                                     */
/* Messages:                                           */
/*                                                     */
/*     None                                           */
/*                                                     */
/* Side Effects:                                       */
/*                                                     */
/*     None                                           */
/*                                                     */
/* Functions/Macros called:                            */
/*                                                     */
/*     qtmfsvr1gn - Server Logon exit function.       */
/*                                                     */
/* Input:                                              */
/* int * argv[1]   - Identifies requesting application */
/*                  (FTP Client =0, FTP Server = 1).  */
/* char * argv[2]  - User identifier from client program. */
/*                  (For FTP server, this is user CMD data */
/* int * argv[3]   - Length (in bytes) of User ID string. */
/* char * argv[4]  - Authentication string from client.  */

```

Figure 301. Sample FTP Logon Exit Program (Part 7 of 26)

```

/*                                                     */
/*     (For FTP server, this is the password)         */
/* int * argv[5]   - Length (bytes) Authentication string. */
/* char * argv[6]  - Internet Protocol address from which */
/*                  the session originates.           */
/* int * argv[7]   - Length (in bytes) of IP address.  */
/* int * argv[8]   - Return code (received as 0).      */
/* char * argv[9]  - User profile (received as blanks). */
/* char * argv[10] - Password (received as blanks).    */
/* char * argv[11] - Initial current library (received as blanks)*/
/*                                                     */
/* Exit Normal: Return Return Code, User Profile, Password, Initial */
/*                  Current Library to server application. */
/*                                                     */
/* Exit Error: None */
/*                                                     */
/* End Function Specification *****/
void main(int argc, char *argv[])
{
    /*******/
    /* Code */
    /*******/

```

Figure 301. Sample FTP Logon Exit Program (Part 8 of 26)

```

/*****
/* Collect input arguments and call function to determine if client */
/* should be allowed to log in to an FTP server application.      */
/*****
qtmfsvrlnn(*(int *) (argv[1]), /* Application Identifier
(Input) */
          argv[2],             /* User Identifier      (Input) */
          (*(int *) (argv[3])), /* Length User of
Identifier(Input) */
          argv[4],             /* Authentication String (Input) */
          (*(int *) (argv[5])), /* Length of Authentication string */
(Input) */
          argv[6],             /* Client IP Address    (Input) */
          (*(int *) (argv[7])), /* Length of Client IP Address */
(Input) */
          (int *) (argv[8]),    /* Return Code          (Output)*/
          argv[9],             /* User Profile         (Output)*/
          argv[10],            /* Password             (Output)*/
          argv[11]);           /* Initial Current Library (Output)*/

```

Figure 301. Sample FTP Logon Exit Program (Part 9 of 26)

```

return;
}

/* Function Specification *****/
/* *****/
/* Function Name: qtmfsvrlnn *****/
/* *****/
/* Descriptive Name: Server Logon exit function. *****/
/* *****/
/* This exit function provides control over user authentication to *****/
/* an FTP server. *****/
/* *****/
/* Notes: *****/
/* *****/
/* Dependencies: *****/
/* *****/
/* FTP Server Logon exit point QIBM_QTMF_SVR_LOGON was *****/
/* registered during FTP product installation. *****/
/* *****/
/* Restrictions: *****/
/* *****/

```

Figure 301. Sample FTP Logon Exit Program (Part 10 of 26)

```

/*                                                     */
/*      None                                           */
/*                                                     */
/* Messages:                                           */
/*                                                     */
/*      None                                           */
/*                                                     */
/* Side Effects:                                       */
/*                                                     */
/*      None                                           */
/*                                                     */
/* Functions/Macros called:                            */
/*                                                     */
/*      CheckClientAddress - Check the ClientIPAddr_p input argument.*/
/*      memcpy - Copy bytes from source to destination. */
/*      memset - Set bytes to value.                  */
/*      strstr - Locate first occurrence of substring. */
/*      sprintf - Formatted print to buffer.          */
/*                                                     */
/* Input:                                              */
/*      int  ApplId          - Application Identifier (Server = 1). */
/*      char * UserId_p     - User identifier from client program. */
/*                          (For FTP server, USER subcommand data)*/

```

Figure 301. Sample FTP Logon Exit Program (Part 11 of 26)

```

/*      int  Lgth_UserId    - Length (in bytes) of user ID string. */
/*      char * AuthStr_p    - Authentication string from client.    */
/*                          (For FTP server, this is the password)*/
/*      int  Lgth_AuthStr   - Length (bytes) Authentication string. */
/*      char * ClientIPAddr_p - Internet Protocol address from which */
/*                          the session originates.                */
/*      int * Lgth_ClientIPAddr - Length (in bytes) of IP address. */
/*                                                     */
/* Output:                                             */
/*      int * ReturnCode: Indicates degree of success of operation: */
/*      ReturnCode = 0 - Reject logon.                    */
/*      ReturnCode = 1 - Continue logon; use initial current library*/
/*      ReturnCode = 2 - Continue logon; override initial current */
/*                      library                                  */
/*      ReturnCode = 3 - Continue logon; override user, password */
/*      ReturnCode = 4 - Continue logon; override user, password, */
/*                      current library                          */
/*      ReturnCode = 5 - Accept logon; override user profile    */
/*      ReturnCode = 6 - Accept logon; override user profile,  */
/*                      current library                          */
/*      char * UserProfile - User profile to use for this session */
/*      char * Password    - Password to use for this session    */
/*      char * Init_Cur_Lib - Initial current library for this session */

```

Figure 301. Sample FTP Logon Exit Program (Part 12 of 26)

```

/*                                                                    */
/* Exit Normal: (See OUTPUT)                                          */
/*                                                                    */
/* Exit Error: None                                                  */
/*                                                                    */
/* End Function Specification *****/
static void qtmfsvrIgn(int ApplId,                                     /* Entry point */
                      char *UserId_p,
                      int Lgth_UserId,
                      char *AuthStr_p,
                      int Lgth_AuthStr,
                      char *ClientIPAddr_p,
                      int Lgth_ClientIPAddr,
                      int *ReturnCode,
                      char *UserProfile_p,
                      char *Password_p,
                      char *InitCurrLib_p)
{
  /******
  /* Local Variables
  /******
  /* The following lists serve as an example of an additional layer
  /* of control over user authentication to an application server.
  /*

```

Figure 301. Sample FTP Logon Exit Program (Part 13 of 26)

```

/* Here, logon operations using the following user identifiers
/* will be allowed to continue, but the output parameters returned
/* by this example exit program will vary depending on which list
/* a user identifier (UserId_p) is found in.
/*
/* For example, attempts to logon as FTPUSR11 or FTPUSR2 will be
/* allowed, and this example exit will return the initial current
/* library as an output parameter along with a return code of 2.
/*
/******
/* Continue the logon operation, Return Code = 1
char Return1[] = "FTPUSR10 ";
/* Continue the logon operation, Return Code = 2
char Return2[] = "FTPUSR11 FTPUSR2 ";
/* Continue the logon operation, Return Code = 3
char Return3[] = "FTPUSR12 FTPUSR3 FTPUSR23 ";
/* Continue the logon operation, Return Code = 4
char Return4[] = "FTPUSER FTPUSR4 FTPUSR24 FTPUSR94 ";
int rc; /* Results of server logon request
Qsy_USRI0300_T Receiver_var; /* QSYRUSRI API Receiver variable
int Lgth_Receiver_var; /* Receiver variable length
char Format_Name[8]; /* Format name buffer
char User_Id[10]; /* User Identifier buffer

```

Figure 301. Sample FTP Logon Exit Program (Part 14 of 26)

```

Qus_EC_t error_code =          /* QSYRUSRI API error code structure: */
{
    sizeof(Qus_EC_t),          /* Set bytes provided          */
    0,                          /* Initialize bytes available */
    ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', /* Initialize Exception Id    */
};
char *pcTest_p;                /* Upper-case User Identifier pointer*/
int i;                          /* "For" loop counter variable    */

/*****
/* Code
*****/

/* Test validity of application ID input argument. */
if(1 NEQ ApplId)
{
    /* ERROR - Not FTP server application. */
    /* Return Code of 0 is used here to indicate */
    /* that an incorrect input argument was received. */
    /* The server logon operation will be rejected. */
    rc = 0;                      /* Application ID not valid */
}

```

Figure 301. Sample FTP Logon Exit Program (Part 15 of 26)

```

} /* End If the application identifier is NOT for FTP server */
else /* FTP server application identifier */
{
    /* Validate the client IP address input argument. */
    rc = CheckClientAddress(ClientIPAddr_p,
                            Lgth_ClientIPAddr);
    if(0 NEQ rc) /* Valid, acceptable client address */
    {
        /* Initialize User_Id; used to hold upper-cased user identifier */
        memset(User_Id, BLANK, sizeof(User_Id));

        /* Initialize pcTest_p to point to UserId_p input argument. */
        pcTest_p = UserId_p;

        /* Uppercase all of the user ID to compare for ANONYMOUS user. */
        for(i = 0; i < Lgth_UserId; i++)
        {
            User_Id[i] = (char)toupper(*pcTest_p);
            pcTest_p += 1;
        }
    }
}

```

Figure 301. Sample FTP Logon Exit Program (Part 16 of 26)

```

/* If user has logged in as ANONYMOUS. */
if(0 == memcmp("ANONYMOUS ", User_Id, 10))
{
/* Determine how to continue with ANONYMOUS logon attempt. */
if(NULL NEQ strstr(Limit, ClientIPAddr_p))
{
/* If users system IP address is found in the "Limit" list, */
/* return ReturnCode of 6, user profile and initial */
/* current library values as output parameters. */
memcpy(UserProfile_p, "USERA1 ", 10);
memcpy(InitCurrLib_p, "PUBLIC ", 10);
rc = 6;
}
}
else
{
/* Users system IP address is NOT found in the "Limit" list,*/
/* return ReturnCode of 5, user profile output parameter; */
/* use the initial current library that is specified by the */
/* user profile information. */
memcpy(UserProfile_p, "USERA1 ", 10);
rc = 5;
}
}

```

Figure 301. Sample FTP Logon Exit Program (Part 17 of 26)

```

} /* End If USER is ANONYMOUS */
else /* Else USER is not ANONYMOUS */
{
/* Set receiver variable length. */
Lgth_Receiver_var = sizeof(Qsy_USRI0300_T);
/* Set return information format. */
memcpy(Format_Name, "USRI0300", sizeof(Format_Name));
/* Set user identifier passed in. */
memset(User_Id, BLANK, sizeof(User_Id));
memcpy(User_Id, UserId_p, Lgth_UserId);
/* Call QSYRUSRI - Retrieve User Information API */
QSYRUSRI(&Receiver_var, /* Return Information receiver var */
Lgth_Receiver_var, /* Receiver variable length */
Format_Name, /* Return information format name */
User_Id, /* User ID seeking information */
&error_code); /* Error return information */
/* Check if an error occurred (byte_available not equal 0) */
if(0 NEQ error_code.Bytes_Available)
{
/* Return ReturnCode of 0 only (Reject logon); */
}
}
}

```

Figure 301. Sample FTP Logon Exit Program (Part 18 of 26)


```

rc = 0;                /* Reject the logon operation */
*ReturnCode = rc;     /* Assign result to ReturnCode */
}
else /* No error occurred from Retrieve User Info */
{ /* (Bytes_Available = 0) */
/* Set current library for user profile. */
memcpy(InitCurrLib_p, Receiver_var.Current_Library, 10);
if(NULL NEQ strstr("CRTDFT ",
Receiver_var.Current_Library))
{
memcpy(InitCurrLib_p, "FTPDEFAULT", 10);
}
}
else
{
if(NULL NEQ strstr(Return1, UserId_p))
{
/* Return ReturnCode of 1 (Continue logon); */
/* Also return user profile and password output */
/* parameters to endure they are ignored by the server.*/
memcpy(UserProfile_p, UserId_p, Lgth_UserId);
memcpy>Password_p, AuthStr_p, Lgth_AuthStr);
}
}
}

```

Figure 301. Sample FTP Logon Exit Program (Part 19 of 26)

```

rc = 1;                /* Continue the logon operation */
}
else
{
if(NULL NEQ strstr(Return2, UserId_p))
{
/* Return ReturnCode of 2, and initial current library*/
/* Also return user profile and password values */
/* even though they will be ignored by the server. */
memcpy(UserProfile_p, UserId_p, Lgth_UserId);
memcpy>Password_p, AuthStr_p, Lgth_AuthStr);
memcpy(InitCurrLib_p, "FTPEXT2",
strlen("FTPEXT2"));
rc = 2; /* Continue logon; return InitCurLib */
}
}
else
{
if(NULL NEQ strstr(Return3, UserId_p))
{
/* Return ReturnCode of 3, user profile, password. */
}
}
}

```

Figure 301. Sample FTP Logon Exit Program (Part 20 of 26)


```

/* Function Specification *****/
/*
/* Function Name: CheckClientAddress
/*
/* Descriptive Name: Check the IP address of the originating session
/*
/*           from the input argument (ClientIPAddr_p) to
/*           ensure it is in valid dotted-decimal format,
/*           and that the client system is allowed access.
/*           This is an example of an input validity check.
/*
/* Notes:
/*
/* Dependencies:
/*     None
/*
/* Restrictions:
/*     None
/*
/* Messages:
/*     None
/*
/* Side Effects:
/*     None

```

Figure 301. Sample FTP Logon Exit Program (Part 23 of 26)

```

/*
/* Functions/Macros called:
/*
/*     strstr - Search for first occurrence of a string.
/*
/* Input:
/* char * ClientIPAddr_p - Internet Protocol address from which
/*                       the session originates.
/* int * Lgth_ClientIPAddr - Length (in bytes) of IP address.
/*
/* Output:
/* int rc - Return code indicating validity of IP
/*         address from ClientIPAddr_p input.
/*         0 = Reject the logon operation.
/*           ClientIPAddr_p is one that is not
/*           allowed, or contains a character
/*           that is not valid.
/*         1 = Continue the logon operation.
/*
/* Exit Normal: (See OUTPUT)
/*
/* Exit Error: None.
/*

```

Figure 301. Sample FTP Logon Exit Program (Part 24 of 26)

```

/* End Function Specification *****/
static int CheckClientAddress(char *ClientIPAddr_p, /* Entry point */
                             int Lgth_ClientIPAddr)
{
  /******
  /* Local Variables
  /******
  int rc; /* Return code */

  /******
  /* Code
  /******
  /* Check that client IP address input argument is dotted-decimal
  /* format of minimum length, with no leading blanks or periods,
  /* and contains only valid characters.
  if((Lgth_ClientIPAddr < 7) || /* Minimum IP address size */
     (strspn(ClientIPAddr_p, ValidChars) < Lgth_ClientIPAddr)||
     (strspn(ClientIPAddr_p, ".") EQ 1)|| /* Leading '.' in IP
     (strspn(ClientIPAddr_p, " ") EQ 1)) /* Leading blank in IP
  {

```

Figure 301. Sample FTP Logon Exit Program (Part 25 of 26)

```

  /* Client's IP address not valid, or contains an incorrect character */
  rc = 0; /* Client IP address input argument not valid */
}
else
{
  /* Is client system allowed to log in to FTP server?
  if(NULL NEQ strstr(Reject, ClientIPAddr_p))
  {
    /* Return code = 0 - Reject the server logon operation, as the
    /* client IP address is found in the global
    /* "Reject" list.
    rc = 0; /* Reject the logon operation
  }
  else
  {
    /* Continue the server logon operation checks.
    rc = 1; /* Continue the logon operation
  }
}
return(rc);
}

#undef _QTMFSVRLGN_C

```

Figure 301. Sample FTP Logon Exit Program (Part 26 of 26)

Anonymous FTP

For information about anonymous FTP, see *AS/400e Information Center*, SK3T-2027-01. For the URL to access the Information Center online, see “TCP/IP Topics in the Information Center” on page xv.

Sample Scenario for Anonymous FTP

The following display shows an anonymous FTP logon sequence. It also shows the server reply that is returned when you attempt an operation that is restricted by the FTP request validation exit program.

```
File Transfer Protocol

Previous FTP subcommands and messages:
Connecting to host sysnam01.city.company.com at address 9.130.42.106 using
21.
220-QTCP at sysnam01.city.company.com.
220 Connection will close if idle more than 5 minutes.
  OS/400 is the remote operating system. The TCP/IP version is "V3R2M0"
> anonymous
331 Guest logon in process, send complete e-mail address as password.
230 Guest logon accepted, access restrictions apply.
250 Now using naming format "0".
257 "PUBLIC" is current library.
> cd qtcp
550 Request rejected.

Enter an FTP subcommand.
====> _____
_____
F3=Exit      F6=Print      F9=Retrieve
F17=Top      F18=Bottom    F21=CL command line
```

Workstation Gateway Server (WSG) Exit Point

There is one AS/400 exit point for workstation gateway server (WSG):
QIBM_QTMT_WSG — workstation gateway server sign-on exit point.

The WSG exit point format is:
QAPP0100

The workstation gateway server sign-on exit point enables client browsers to call applications directly without sending a user profile or password. This allows a client browser to bypass the AS/400 sign-on display and to call specific applications without requiring a sign-on. You can also use this exit point to control which user profile is used for a Web browser.

When an exit program is added to the exit point, the exit program is called each time there is an attempt to sign-on to the server. The exit program sets the **Allow operation output** parameter to indicate whether or not the server is to continue the sign-on operation. Also, different output parameters may be set to provide sign-on information to the workstation gateway server. The WSG server then uses these output parameters and performs the sign-on on behalf of the client browser.

Workstation Gateway Server Sign-on Exit Point Interface (QAPP0100) Required Parameters

Here are the required parameters for the workstation gateway server exit point interface (QAPP0100)

QAPP0100 Required Parameter Group:

1	Operation-specific parameter (optional)	Input	Char(*)
2	Length of operation-specific parameter content	Input	Binary(4)
3	Client IP address	Input	Char(15)
4	CCSID	Input	Binary(4)
5	Allow operation	Output	Char(1)
6	User profile	Output	Char(10)
7	Password	Output	Char(10)
8	Current library	Output	Char(10)
9	Program to call	Output	Char(10)
10	Initial menu	Output	Char(10)
11	Session end URL	Output	Char(300)

Descriptions of Required Parameters for the WSG Exit Point Interface (QAPP0100)

Here are the descriptions of the required parameters for the WSG exit point interface (QAPP0100).

Op_specific_parm

INPUT:CHAR(*) This is information for the exit program. It appears in the URL after the /WSG/QAPP0100? string, as follows:

`http://sysnam.location.company.com:port/ WSG/QAPP0100?op_specific_parm`

Anything that follows the /WSG/QAPP0100? string is passed as an operation-specific parameter. This may be any kind of character string.

If no information is appended, then it is set to a NULL pointer.

Length of operation-specific parameter content

INPUT:BINARY(4) The length, in bytes, of any operation-specific parameter content from the URL. The length is set to zero if no operation-specific parameter is provided.

Client IP address

INPUT:CHAR(15) The Internet Protocol (IP) (dotted decimal) address of the client, which is extracted from the connected socket information. Because this is extracted from the connected socket, this may actually be a firewall address, router, or other identifier, and not the actual client IP address.

CCSID

INPUT:BINARY(4) This is the EBCDIC CCSID of the data in parameter 1; it is always CCSID 500.

Allow operation

OUTPUT:BINARY(4) Indicates whether the application should be called on behalf of the client browser. The possible values are:

- 0** Reject the application request from the client browser. The WSG server ignores the user profile, password, current library, program to call, and initial menu output parameters and sends an error reply to the client.
- 1** Accept the application request from the client browser. The exit program supplies the user profile, password, current library, program to call and initial menu output parameters.

User profile

OUTPUT:CHAR(10) The user profile to be used for the sign-on procedure. This parameter is required. It must be left justified and padded with blanks.

Password

OUTPUT:CHAR(10) The password to be used for the sign-on procedure. This parameter is required. It must be left justified and padded with blanks.

Current library

OUTPUT:CHAR(10) The library in which the program to be called is located. This parameter is optional. It must be left justified and padded with blanks. If you do not supply a library, the WSG server uses the default of *USRPRF.

Program to call

OUTPUT:CHAR(10) The name of the program to be called. This parameter is optional. It must be left justified and padded with blanks. If you do not supply a program name, the WSG server uses the default of *USRPRF.

Initial menu

OUTPUT:CHAR(10) The name of the initial menu to display. This parameter is optional. It must be left justified and padded with blanks. If you do not supply a menu name, the WSG server uses the default of *USRPRF.

Session end URL

OUTPUT:CHAR(300) The URL link, which should be sent to the web browser when the WSG session is ended. The system strips all trailing blanks. If you do not specify a URL, then the default action is to post timing statistics.

See “Using a WSG exit program to bypass the AS/400 Sign-on Display” for more information on the workstation gateway server sign-on exit point and a sample program.

Using a WSG exit program to bypass the AS/400 Sign-on Display

The WSG Application Logon exit point allows you to bypass the AS/400 sign-on display and call an application directly without the client browser sending a user profile or password. This allows you to provide *any* application to client browsers without requiring a sign-on. To allow users to bypass the sign-on display, your program must authenticate the client request and provide sign-on information to the workstation gateway server. The workstation gateway server then uses the output of your exit program to perform the sign-on for the client.

You may pass information to the exit program by adding (concatenating) it to the end of the URL after “QAPP0100” as follows:

```
http://hostname:port/WSG/QAPP0100?op_specific_parm
```

where:

- http://hostname:port defines the protocol, host system and port number
- /WSG indicates that this is a request for the workstation gateway.
- /QAPP0100 tells the WSG server to call any registered exit program.
- ? is the delimiter between the QAPP0100 string and the actual parameter.
- op_specific_parm is information for the exit program.

When the exit program gets control, it must validate the request by using the supplied Internet Protocol address and any operation-specific information passed to it.

Sample WSG Server Logon Exit Program

```
/* Module Description *****/
/*
/* Source File Name: wsgexit.c
/*
/* Module Name: Workstation Gateway Server logon exit program.
/*
/* Service Program Name: n/a
/*
/* Source File Description:
/*
/* This module contains functions to allow a client browser to
/* bypass an AS/400 sign-on panel and invoke an application.
/*
/* OCO Source Materials
/*
/* The Source code for this program is not published or otherwise
/* divested of its trade secrets, irrespective of what has been
/* deposited with the U.S. Copyright Office
/*
/* 5763-TC1,5716-TC1,5769-TC1 (C) Copyright IBM Corp. 1987, 1998.
/*
```

Figure 302. Sample WSG Server Logon Exit Program (Part 1 of 50)

```
/* Change Activity:
/*
/* CFD List:
/*
/* $A0=Dxxxxx 3D20 960331 FINGARJM: New function.
/* $A1=Dxxxxx 4D20 970722 JSTEVENS: Updates, new functions.
/*
/* End CFD List.
/*
/* Additional notes about the Change Activity
/*
/* $A1 - Better logging scheme, flight recorder, new parm 11 added.
/* $A0 - Initial user exit program.
/*
/* End Change Activity.
/*
/* End Module Description *****/
```

Figure 302. Sample WSG Server Logon Exit Program (Part 2 of 50)


```

#pragma comment (copyright, \
"5763-TC1,5716-TC1,5769-TC1 \
(C) Copyright IBM Corp. 1996, 1998. \
All rights reserved. \
US Government Users Restricted Rights - \
Use, duplication or disclosure restricted \
by GSA ADP Schedule Contract with IBM Corp. \
Licensed Material-Property of IBM")

#define _WSGEXIT_C

/*****
/* All file scoped includes go here */
*****/
#include <ctype.h> /* isspace, toupper */
#include <except.h> /* _CNL_Hndlr_Parms_T structure */
#include <recio.h> /* _Ropen, _Rwrite, _Rreadf */
#include <signal.h> /* _GetExcData(), signal() */
#include <stdarg.h> /* va_start(), va_arg(), va_end() */
#include <stdio.h> /* sprintf, printf, fopen, fread, */
#include <stdlib.h> /* system, atoi, free, malloc, exit, */

```

Figure 302. Sample WSG Server Logon Exit Program (Part 3 of 50)

```

#include <string.h> /* strxxx, memxxx */
#include <time.h> /* time(), ctime() */
#include <xxfdbk.h> /* _Ropnfbk */
#ifdef __ILEC400__
#include <qusec.h> /* Include for API error code struct */
#include <qcapcmd.h> /* QCAPCMD - Process CL commands */
#include <qmhrtvm.h> /* QMHRTVM - retrieve program message*/
#include <qmhsndm.h> /* QMHSNDM - send non-program msg */
#include <qmhchgem.h> /* QMHCHGEM - change exception msg */
#include <qwcrdtaa.h> /* QWCRDTAA() - retrieve data area */
#include <qmhsndpm.h> /* QMHSNDPM - send program message */

```

Figure 302. Sample WSG Server Logon Exit Program (Part 4 of 50)

```

/*****
/* Make all API calls library qualified (valid only at V3R7 and up) */
/*****
#pragma map (QDCRDEVD, "QSYS/QDCRDEVD")
#pragma map (QCAPCMD, "QSYS/QCAPCMD")
#pragma map (QMHRTVM, "QSYS/QMHRTVM")
#pragma map (QMHSNDM, "QSYS/QMHSNDM")
#pragma map (QMHCHGEM, "QSYS/QMHCHGEM")
#pragma map (QWCRDTAA, "QSYS/QWCRDTAA")
#pragma map (QMHSNDPM, "QSYS/QMHSNDPM")
#endif

/*****
/* All file scoped Constants go here */
/*****
const int fTrue = 1;
const int fFalse = 0;

```

Figure 302. Sample WSG Server Logon Exit Program (Part 5 of 50)

```

/*****
/* All file scoped type declarations go here */
/*****
typedef struct _ERRSTRUCT {
    int Bytes_Provided;
    int Bytes_Available;
    char Exception_Id[7];
    char Reserved;
    char Exception_Data[256];
} ERRSTRUCT;

typedef ERRSTRUCT *PERRSTRUCT;

/*****
/* All file scoped Macro invocations go here */
/*****
#define MAX(a,b) (((a) > (b)) ? (a) : (b))
#define MIN(a,b) (((a) < (b)) ? (a) : (b))

```

Figure 302. Sample WSG Server Logon Exit Program (Part 6 of 50)

```

/*****
/* The following describe the log file */
/*****
#define RECORD_WIDTH 240

/*****
/* Some message API definitions */
/*****
#define MSG_DIAG      "*DIAG      "
#define MSG_ESCAPE    "*ESCAPE    "
#define MSG_INFO      "*INFO      "
#define MSG_INQ       "*INQ       "
#define MSG_COMP      "*COMP      "
#define MSG_NOTIFY    "*NOTIFY    "
#define MSG_RQS       "*RQS       "
#define MSG_STATUS    "*STATUS    "

```

Figure 302. Sample WSG Server Logon Exit Program (Part 7 of 50)

```

#define MSQ_Q_CUR_PROG "*"

#define QTCMSG      "QTCMSG  *LIBL  "      /* Stack MSGF */
#define QTCMSGF    "QTCMSGF *LIBL  "      /* Apps MSGF   */
#define QCPFMSG    "QCPFMSG  *LIBL  "      /* Most CPFxxx */
#define QCEEMSG    "QCEEMSG  *LIBL  "      /* CEE9901     */
#define QC2MSGF    "QC2MSGF  *LIBL  "      /* ILE Signals */

/*****
/* All internal function prototypes go here */
/*****
static void buffer(char *Buffer, int Length);
static void record(char *Format, ...);
void handler(int iSignal);
static void cl_command(char *Command);
static int Pad(char *pszString, int iLength);
static int Trim(char *pszString, int iLength);

```

Figure 302. Sample WSG Server Logon Exit Program (Part 8 of 50)

```

/*****
/* All file scoped variable declarations go here */
/*****
static int fRemoveEscapeMsg;
static int fException;
static int fDebug;          /* True means to log records to DEBUG */
static int fLogMsg;        /* True means send msg to QTCP msgq */
static int fLogFile;       /* True means log msg to SRCPF or PF */
static char acMsg[256];
static char acMsgKey[4];
static char acMsgReply[20];

char *pszLog      = "LOG";
char *pszDebug   = "DEBUG";
char *pszDtaara  = "WSGEXIT QTCP ";
char *pszDataArea = "QTCP/WSGEXIT";

```

Figure 302. Sample WSG Server Logon Exit Program (Part 9 of 50)

```

/* Function Specification *****/
/*
/* Function Name: Main */
/*
/* Descriptive Name: Application Logon exit program sample program. */
/*
/* This test exit program provides control over signon panels via */
/* the WSG server in the V3R2 release. */
/*
/* Notes: For V3R2 the "argv[]" parameters are "char *" by definition.*/
/* Reference integers as "*(int(argv[1]))", for example. */
/* Consider the method for passing them back to the caller. */
/*
/* Dependencies: */
/* WSG Applicaton Logon exit point QIBM_QTMT_WSG format QAPP0100 */
/* was registered during WSG V3R2 installation. */
/*

```

Figure 302. Sample WSG Server Logon Exit Program (Part 10 of 50)

```

/* Restrictions: */
/* None */
/* Messages: */
/* None */
/* Side Effects: */
/* None */
/* Functions/Macros called: */
/* None */

```

Figure 302. Sample WSG Server Logon Exit Program (Part 11 of 50)

```

/* Input: */
/* char * argv[1] - Operation specific information */
/* int argv[2] - Length of operation specific information */
/* char * argv[3] - IP address of the remote host system. */
/* int argv[4] - CCSID of the operaton specific info */
/* char * argv[5] - Allow operation '0'=No, '1'=Yes(output) */
/* char * argv[6] - User profile to be used (output) */
/* char * argv[7] - Password to be used (output) */
/* char * argv[8] - Program library to be used (output) */
/* char * argv[9] - Program name to be used (output) */
/* char * argv[10] - Menu panel to be used (output) */
/* char * argv[11] - Return URL when session closed (output) */
/* */
/* Exit Normal: Return AllowOper value to server application. */
/* */
/* Exit Error: None */
/* */
/* End Function Specification *****/
void main(int argc, char *argv[])
{

```

Figure 302. Sample WSG Server Logon Exit Program (Part 12 of 50)

```

/*****
/* Local variables
*****/
int iParms;          /* # required parameters */
int iMin;           /* Working variable */
int i;              /* Working variable */
char *Accept;
char *Reject;

char *OperSpecInfo_p;
int Lgth_OperSpecInfo;
char *ClientIPAddr;
int CCSID;
char *AllowOper;
char *UserProfile;
char *Password;
char *ProgramLib;
char *ProgramName;

```

Figure 302. Sample WSG Server Logon Exit Program (Part 13 of 50)

```

char *InitialMenu;
char *URL;

char pcOpenFile[RECORD_WIDTH]; /* _Ropen() file name */
char pcOpenParms[RECORD_WIDTH]; /* _Ropen() attributes */
FILE *pDebug; /* Debug file pointer */
FILE *pLog; /* Log file pointer */

/*****
/* Data area layout expected:
/* - required to have library and name of logging file
/* - if you want a connect msg to QTCP msgq, add *MSG string
/* - if you want debug msgs to DEBUG member, add *DEBUG string
/*
/* Example:
/* 'filelib/filename msgqflag debugflag'
/* 'QTCP/WSGEXIT *MSGQ *DEBUG'
*****/

```

Figure 302. Sample WSG Server Logon Exit Program (Part 14 of 50)

```

/*****
typedef struct _Dtaara {
    /*****
    /* typedef _Packed struct Qwc_Rdtaa_Data_Returned {
    /* int Bytes_Available;
    /* int Bytes_Returned;
    /* char Type_Value_Returned[10];
    /* char Library_Name[10];
    /* int Length_Value_Returned;
    /* int Number_Decimal_Positions;
    /* char Value[];           commented out
    /* } Qwc_Rdtaa_Data_Returned_t;
    /*****
    Qwc_Rdtaa_Data_Returned_t returned;
    char contents[257];
} Dtaara_t;
Dtaara_t dtaara;

ERRSTRUCT esErrCode;

```

Figure 302. Sample WSG Server Logon Exit Program (Part 15 of 50)

```

/*****
/* Code
/*****
fException      = fFalse;    /* Exception was trapped by signal() */
fRemoveEscapeMsg = fFalse;  /* Remove exception msg from joblog */
fDebug          = fFalse;    /* Log debug msgs to DEBUG member */
fLogMsg         = fFalse;    /* Log connect msg to QTCP msg queue */
fLogFile        = fFalse;    /* Log connect msg to SRCPF or PF */
signal(SIGALL, &handler);   /* Trap all signals with our handler */

/*****
/* Read fixed data area to see if debug logging is active. We
/* force exceptions to be signalled so we know if the data area
/* exists or not by checking the fException flag.
/*****

```

Figure 302. Sample WSG Server Logon Exit Program (Part 16 of 50)

```

/*****
memset(&esErrCode, 0x00, sizeof(esErrCode));
esErrCode.Bytes_Provided = 0L; /* Force exception to be signalled */
memset(&dtaara, 0x00, sizeof(dtaara));
dtaara.returned.Bytes_Available = sizeof(dtaara);
fException = fFalse;          /* Initialize - reset in handler */
fRemoveEscapeMsg = fTrue;     /* Initialize - want to hide errors */
QWCRDTAA(&dtaara, sizeof(dtaara), pszDtaara, -1,
        sizeof(dtaara.contents) - 1, &esErrCode);
fRemoveEscapeMsg = fFalse;    /* Reset - want to see errors */
/*****
/* If data area found and contents exist, process contents
/*****
if (!fException && dtaara.returned.Bytes_Returned) {
if (NULL != strstr(dtaara.contents, "MSGQ")) {
    /*****

```

Figure 302. Sample WSG Server Logon Exit Program (Part 17 of 50)

```

/* If find *MSGQ, means log connect msg to QTCP msg queue      */
/*****
fLogMsg = fTrue;
} /* endif */
if (NULL != strstr(dtaara.contents, "*FILE")) {
/*****
/* If find *FILE, means log connect msg to SRCPF or PF      */
/*****
fLogFile = fTrue;
} /* endif */
if (NULL != strstr(dtaara.contents, "*DEBUG")) {
/*****
/* If find *DEBUG, log flight records to map file member DEBUG */
/*****

```

Figure 302. Sample WSG Server Logon Exit Program (Part 18 of 50)

```

fDebug = fTrue;
} /* endif */
/*****
/* Map file is supposed to be the first string in the data area */
/*****
pszDataArea = strtok(dtaara.contents, " ");
} /* endif */

if (fDebug) {
/*****
/* record() output goes to standard out */
/*****

```

Figure 302. Sample WSG Server Logon Exit Program (Part 19 of 50)

```

sprintf(pcOpenFile, "%s(%s)", pszDataArea, pszDebug);
sprintf(pcOpenParms, "a+, lrecl=%d, recfm=v", RECORD_WIDTH);
pDebug = freopen(pcOpenFile, pcOpenParms, stdout);
} /* endif */

/*****
/* Initialize for QMHSNDM API calls later */
/*****
memset(acMsg, 0x00, sizeof(acMsg));
memset(acMsgKey, 0x00, sizeof(acMsgKey));

record("\n");
record("wsgexit: >>>> entry\n");
record("wsgexit: Signals -> handler\n");
signal(SIGALL, &handler); /* Trap all signals with our handler */
iParms = 11; /* Total of 12 parms on interface */
record("wsgexit: argc: %d iParms: %d\n", argc, iParms);

```

Figure 302. Sample WSG Server Logon Exit Program (Part 20 of 50)


```

if ((argc-1) != iParms) {
    record("wsgexit: Invalid number of parameters!\n");
    /******
    /* TCP7117 - Program &1 in library &2 received an invalid number */
    /* parameters. */
    /******
    sprintf(acMsg, "WSGEXIT *LIBL ");
    iMin = strlen(acMsg);
    memcpy(&acMsg[iMin], &iParms, sizeof(int));
    iMin += sizeof(int);
    QMHSNDPM("TCP7117", QTCPMSGF, acMsg, iMin, MSG_INFO,
            MSQ_Q_CUR_PROG, 0, acMsgKey, &esErrCode);
    record("wsgexit: <<<< exit\n\n");
    exit(0);
} /* endif */

record("wsgexit: OperSpecInfo      = >%s<\n", argv[1]);
record("wsgexit: Lgth_OperSpecInfo = %d\n", *((int *)argv[2]));
record("wsgexit: ClientIPAddr     = >%s<\n", argv[3]);

```

Figure 302. Sample WSG Server Logon Exit Program (Part 21 of 50)

```

record("wsgexit: CCSID              = %d\n", *((int *)argv[4]));
record("wsgexit: AllowOper         = '%c'\n", *argv[5]);
record("wsgexit: UserProfile       = >%.10s<\n", argv[6]);
record("wsgexit: Password         = >%.10s<\n", argv[7]);
record("wsgexit: ProgramLib       = >%.10s<\n", argv[8]);
record("wsgexit: ProgramName      = >%.10s<\n", argv[9]);
record("wsgexit: InitialMenu      = >%.10s<\n", argv[10]);
record("wsgexit: URL              = >%.300s<\n", argv[11]);

OperSpecInfo_p = argv[1];
Lgth_OperSpecInfo = *((int *)argv[2]);
ClientIPAddr = argv[3];
CCSID = *((int *)argv[4]);
AllowOper = argv[5];
UserProfile = argv[6];
Password = argv[7];
ProgramLib = argv[8];
ProgramName = argv[9];
InitialMenu = argv[10];
URL = argv[11];

```

Figure 302. Sample WSG Server Logon Exit Program (Part 22 of 50)

```

/*****
/* Define a return URL to use - echo any operation specific info */
/*****
if (Lgth_OperSpecInfo && !strncmp("http://", OperSpecInfo_p, 7)) {
    strcpy(URL, OperSpecInfo_p);
} else {
    strcpy(URL, "http://www.ibm.com");
} /* endif */
record("wsgexit: Set URL: >%s<\n", URL);

/*****
/* Accept the following client systems for AS/400 sign-on bypass. */
/*
/* AS400.IBM.COM (1.222.33.44)
/* AS401.IBM.COM (1.222.33.55)
/* AS402.IBM.COM (1.222.33.66)
/*
/*
/*****

```

Figure 302. Sample WSG Server Logon Exit Program (Part 23 of 50)

```

Reject = "1.222.33.77 1.222.33.88 1.222.33.99 ";

/*****
/* Validate Client IP address */
/*****
record("wsgexit: Validate IP address\n");
if (NULL != strstr(Accept, ClientIPAddr)) {
    /*****
    /* Parse the "OperSpecInfo_p" input string:
    /* Return AllowOper of '1' - Accept this clients request.
    /* Set return values for client browser:
    /*****
    memcpy(UserProfile, "SOMEUSER ", 10);
    memcpy>Password, "SOMEPW ", 10);
    memcpy(ProgramLib, " ", 10);
    memcpy(ProgramName, "QCMD ", 10);
    memcpy(InitialMenu, " ", 10);
    memcpy(AllowOper, "1", 1);
    record("wsgexit: Client is in Accept list: AllowOper='1'\n");
} else {
    /* Check if client is in "Reject" list
    if (NULL != strstr(Reject, ClientIPAddr)) {

```

Figure 302. Sample WSG Server Logon Exit Program (Part 24 of 50)

```

/*****
/* Return AllowOper of '0' - Reject this operation. */
/*****
record("wsgexit: Client is in Reject list: AllowOper='0'\n");
memcpy(AllowOper, "0", 1);
} else {
/*****
/* Return AllowOper of '0' - Invalid user for this operation */
/*****
record("wsgexit: Client not in either list: AllowOper='0'\n");
memcpy(AllowOper, "0", 1);
}
} /* End Check if client is in "Reject" list */

```

Figure 302. Sample WSG Server Logon Exit Program (Part 25 of 50)

```

/*****
/* Create message for logging */
/*****
sprintf(acMsg,
        "IP: >%s< Allow: '%c' Profile: >%s< OperLen: >%d< Oper: >%s<",
        ClientIPAddr,
        *AllowOper,
        UserProfile,
        Lgth_OperSpecInfo,
        OperSpecInfo_p);

record("wsgexit: Log record:\n");
buffer(acMsg, strlen(acMsg));

```

Figure 302. Sample WSG Server Logon Exit Program (Part 26 of 50)

```

/*****
/* If data area found and special string found, send connect msg */
/*****
if (fLogMsg) {
    memset(&esErrCode, 0x00, sizeof(esErrCode));
    esErrCode.Bytes_Provided = sizeof(esErrCode); /* Ignore exceptions */
    QMHSNDM("CPF9897", QCPFMSG, acMsg, strlen(acMsg), MSG_INFO,
            "QTCP *USER ", 1, acMsgReply, acMsgKey, &esErrCode);
} /* endif */

/*****
/* If data area found and special string found, log connect msg */
/*****
if (fLogFile) {

```

Figure 302. Sample WSG Server Logon Exit Program (Part 27 of 50)

```

/*****
/* Open database file for permanent log of test results.      */
/*****
sprintf(pcOpenFile, "%s(%s)", pszDataArea, pszLog);
sprintf(pcOpenParms, "a+, lrecl=%d, recfm=v", RECORD_WIDTH);
pLog=fopen(pcOpenFile, pcOpenParms);
fwrite(acMsg, 1, strlen(acMsg), pLog);
fclose(pLog);
} /* endif */

/*****
/* Exit                                                         */
/*****
record("wsgexit: <<<< exit\n");
exit(0);
}                                     /* End main */

```

Figure 302. Sample WSG Server Logon Exit Program (Part 28 of 50)

```

/*****
/*@function buffer()                                          */
/*                                                         */
/* Parameters:                                              */
/*                                                         */
/* char *buffer - points at buffer to dump                 */
/* int length - length of buffer to dump                  */
/*                                                         */
/* Description:                                             */
/*                                                         */
/* Dumps out a buffer in both hex and readable form.      */
/*                                                         */
/*           1934D8E3 D4E3E2D7 C3F0F0F2 40404040 |..QTM TSPC002 | */
/*           40404040 40404040 40404040 40404040 |                | */
/*           00000000 00000000 00000000 00000000 |.....|      */
/*****
void buffer(char *Buffer, int Length)
{
    int iRow      = 0;

```

Figure 302. Sample WSG Server Logon Exit Program (Part 29 of 50)

```

int iCol      = 0;
int iLast    = 0;
int iTotRows = 0;
int iLen;
int iBytes;
unsigned char c;
char aLine[20];
char Buff[512];
char *pBuff = Buff;

/*****
/* If debug is not active, nothing to do here...          */
/*****
if (!fDebug || (Buffer == (char *)NULL)) {
    record("buffer: Buffer is NULL\n");
    return;
} /* endif */

```

Figure 302. Sample WSG Server Logon Exit Program (Part 30 of 50)

```

/*****
/* Calculate total rows to fit all bytes                    */
/*****
iTotRows = (Length + 15) / 16;

/*****
/* Primary loop for rows                                   */
/*****
for (iRow = 0; iRow < iTotRows; iRow++) {
    /*****
    /* Offset into line by 4 blank characters              */
    /*****
    *pBuff = ' '; pBuff++;
    *pBuff = ' '; pBuff++;
    *pBuff = ' '; pBuff++;
    *pBuff = ' '; pBuff++;
    /*****
    /* Print 16 bytes of the buffer as the hexadecimal dump section */
    /* Primary loop for columns                             */
    /*****

```

Figure 302. Sample WSG Server Logon Exit Program (Part 31 of 50)

```

    for (iCol = 0; iCol < 16; iCol++) {
/*****
/* One printable character displays two hexadecimal characters */
/*****
if (iCol + (iRow * 16) >= Length) {
/*****
/* No more data - just fill rewsgeixiting positions with blanks */
/*****
*pBuff = ' '; pBuff++;
*pBuff = ' '; pBuff++;
} else {
/*****
/* Print actual data */
/*****
sprintf(pBuff, "%02X", Buffer[iCol + (iRow * 16)]);
pBuff += 2;
} /* endelse */

```

Figure 302. Sample WSG Server Logon Exit Program (Part 32 of 50)

```

/*****
/* Add a space separator every 4 hex bytes */
/*****
if (iCol % 4 == 3) {
    *pBuff = ' '; pBuff++;          /* Pad character */
} /* endif */
} /* endfor */
/*****
/* Print the same 16 bytes as the "readable" text section */
/*****
*pBuff = ' '; pBuff++;
*pBuff = '|'; pBuff++;          /* Left text bar */
/*****
/* Print 16 bytes of the buffer as readable text section */
/* Secondary loop for columns */
/*****
iLast = 0;          /* Bytes last line */
for (iCol = 0; iCol < 16; iCol++) {
    if (iCol + (iRow * 16) >= Length) {

```

Figure 302. Sample WSG Server Logon Exit Program (Part 33 of 50)

```

/*****
/* No more data - just fill positions with blanks */
/*****
*pBuff = ' '; pBuff++;
} else {
/*****
/* Print actual data */
/*****
iLast++; /* Actual data */
c = Buffer[iCol + (iRow * 16)];
if (isprint(c)) {
*pBuff = c; pBuff++; /* Actual char */
} else if (c == 0x40) {
*pBuff = ' '; pBuff++; /* Blank char */
} else {
*pBuff = '.'; pBuff++; /* Unprintable */
} /* endelse */
} /* endelse */
} /* endfor */
*pBuff = '|'; pBuff++; /* Right text bar */

```

Figure 302. Sample WSG Server Logon Exit Program (Part 34 of 50)

```

/*****
/* Build the finished output */
/*****
memset(acLine, 0x00, sizeof(acLine));
iBytes = (iRow * 16) + iLast;
sprintf(acLine, " Byte %d\n", iBytes); /* Byte count */
strcpy(pBuff, acLine);
pBuff = &Buff[0]; /* Reset pointer */
printf("%s", Buff);
} /* endfor */
return;
}

/*****
/*@function record() */
/* */
/* Parameters: */
/* */
/* variable arguments */
/* */
/* Log test result entry to standard out (normally console). This */
/* occurs only if DEBUG is active. If this occurs in a batch */
/* job, a spooled file is usually created that holds output. */
/*****

```

Figure 302. Sample WSG Server Logon Exit Program (Part 35 of 50)

```

void record(char *Format, ...)
{
    va_list arg_ptr;
    int iLen;
    char record[512];

    /*****
    /* If debug is not active, nothing to do here...
    *****/
    if (!fDebug) {
        return;
    } /* endif */

    va_start(arg_ptr, Format);
    iLen = vsprintf(record, Format, arg_ptr);
    va_end(arg_ptr);
    record[iLen] = 0x00; record[iLen] = 0x00;
    printf("%s", record);
    return;
}

```

Figure 302. Sample WSG Server Logon Exit Program (Part 36 of 50)

```

/*****
/*@function handler()
/*
/* Parameters:
/*
/* int iSignal - value of the signal that caused the handler to be
/*          invoked. SIGABRT, SIGTERM, etc.
*****/
void handler(int iSignal)
{
    ERRSTRUCT esErrCode;

    _INTRPT_Hndlr_Parms_T Signal;
    _INTRPT_Hndlr_Parms_T *pSignal = &Signal

    char *pszMsgFile = NULL;

    struct {
        int Bytes_Return;
        int Bytes_Available;
        int Length_Message_Returned; int Length_Message_Available;
        int Length_Help_Returned;
        int Length_Help_Available;
        char Message[256];
        char Message_Help[256];
    } rtvm0100;
}

```

Figure 302. Sample WSG Server Logon Exit Program (Part 37 of 50)


```

char *Signals[] = {
"SIGABRT", /* 1 Abnormal termination */
"SIGFPE", /* 2 Erroneous arithmetic operation */
"SIGILL", /* 3 Invalid hardware instruction */
"SIGINT", /* 4 Interactive attention signal */
"SIGSEGV", /* 5 Invalid memory reference */
"SIGTERM", /* 6 Termination signal */
"SIGUSR1", /* 7 Application defined signal 1 */
"SIGUSR2", /* 8 Application defined signal 2 */
"SIGIO", /* 9 I/O possible, or completed */
"SIGALL", /* 10 All signals */
"SIGOTHER", /* 11 ILE C/400 signal */
"SIGKILL", /* 12 Termination signal (cannot be caught, ignored) */
"SIGPIPE", /* 13 Write on a pipe with no readers */
"SIGALRM", /* 14 Timeout signal */
"SIGHUP", /* 15 Hangup detected on controlling terminal */
"SIGQUIT", /* 16 Interactive termination signal */
"SIGSTOP", /* 17 Stop signal (cannot be caught or ignored) */
"SIGTSTP", /* 18 Interactive stop signal */
"SIGCONT", /* 19 Continue if stopped */
"SIGCHLD", /* 20 Child process terminated or stopped */
"SIGTTIN", /* 21 Background read from controlling terminal */

```

Figure 302. Sample WSG Server Logon Exit Program (Part 38 of 50)

```

"SIGTTOU", /* 22 Background write to controlling terminal */
"SIGURG", /* 23 High bandwidth data is available at a socket */
"SIGPOLL", /* 24 Pollable event */
"SIG25", /* 25 Not defined */
"SIG26", /* 26 Not defined */
"SIG27", /* 27 Not defined */
"SIG28", /* 28 Not defined */
"SIG29", /* 29 Not defined */
"SIG30", /* 30 Not defined */
"SIG31", /* 31 Not defined */
"SIGBUS", /* 32 Bus error (specification exception) */
"SIGDANGER", /* 33 system crash imminent */
"SIGPRE", /* 34 programming exception */
"SIGSYS", /* 35 Bad system call */
"SIGTRAP", /* 36 Trace/Breakpoint trap */
"SIGPROF", /* 37 Profiling timer expired */
"SIGVTALRM", /* 38 Virtual timer expired */
"SIGXCPU", /* 39 CPU time limit exceeded */
"SIGXFSZ" /* 40 File size limit exceeded */
};

record("handler: Caught signal %s\n", Signals[iSignal-1]);

```

Figure 302. Sample WSG Server Logon Exit Program (Part 39 of 50)

```

/*****
/* Set file scoped flag so caller knows exception occurred */
/*****
fException = fTrue;

/*****
/* Try and pull out the message text */
/*****
_GetExcData(&Signal);
if (!memcmp(Signal.Msg_Id, "TCP", 3) ) {
    /*****
    /* TCP Apps (and Stack with recursive call) message file */
    /*****
    pszMsgFile = QTCPSMGF;
} else if (!memcmp(Signal.Msg_Id, "C2M16", 5) ) {

```

Figure 302. Sample WSG Server Logon Exit Program (Part 40 of 50)

```

/*****
/* ILE-C message file (primarily for signals, if use raise/abort) */
/*****
pszMsgFile = QC2MSGF;
} else if (!memcmp(Signal.Msg_Id, "CEE99", 5) ) {
    /*****
    /* ILE-C message file */
    /*****
    pszMsgFile = QCEEMSG;
} else {
    /*****
    /* Most CPFxxxx messages */
    /*****
    pszMsgFile = QCPFMSG;
} /* endif */

```

Figure 302. Sample WSG Server Logon Exit Program (Part 41 of 50)

```

memset(&esErrCode, 0x00, sizeof(esErrCode));
esErrCode.Bytes_Provided = sizeof(esErrCode); /* Ignore exceptions */
QMHRVTVM(&rtvm0100, /* Message information */
    sizeof(rtvm0100), /* Length of message information */
    "RTVM0100", /* Format name */
    Signal.Msg_Id, /* Message identifier */
    pszMsgFile, /* Qualified message file name */
    Signal.Ex_Data, /* Message data */
    sizeof(Signal.Ex_Data), /* Length of message data */
    "*YES ", /* Replace substitution values */
    "*NO ", /* Return format control */
    &esErrCode, /* Error Code */
    "*MSGID ", /* Retrieve option */
    0, /* Convert to CCSID */
    0); /* Message data CCSID */

if (esErrCode.Bytes_Available || !rtvm0100.Length_Message_Returned) {

```

Figure 302. Sample WSG Server Logon Exit Program (Part 42 of 50)

```

    record("handler: Escape message not found\n");
} else {
    record("handler: Escape message:\n");
    buffer(rtm0100.Message, rtm0100.Length_Message_Returned);
} /* endelse */

/*****
/* Delete message from job log if caller so desires (flag indicator)*/
*****/
if (fRemoveEscapeMsg) {
    record("handler: Remove escape message from job log\n");
    memset(&esErrCode, 0x00, sizeof(esErrCode));
    esErrCode.Bytes_Provided = sizeof(esErrCode);/* Ignore exceptions */
    QMCHGEM(&Signal.Target, 0, &Signal.Msg_Ref_Key, "REMOVE ",
            "", 0, &esErrCode);
} else {
    record("handler: Escape message not removed from job log\n");
    record("handler: Reset signals -> handler\n");
    signal(SIGALL, &handler);
    return;
}

```

Figure 302. Sample WSG Server Logon Exit Program (Part 43 of 50)

```

/* Function Specification *****/
/*
/* Function Name: cl_command
/*
/* Descriptive Name: run any CL command as if on a command line.
/*
/* char * Command - null terminated string
/*
/* End Function Specification *****/
void cl_command(char *Command) /* Entry point */
{
    /*****
    /* Local Variables
    *****/
    ERRSTRUCT esErrCode;

    Qca_PCMD_CPOP0100_t cpop0100;
    char cpop0100_out[512];
    int cpop0100_len;

```

Figure 302. Sample WSG Server Logon Exit Program (Part 44 of 50)

```

/*****
/* typedef _Packed struct Qca_PCMD_CPOP0100 {
/*     int  Command_Process_Type;
/*     char DBCS_Data_Handling;
/*     char Prompter_Action;
/*     char Command_String_Syntax;
/*     char Message_Key[4];
/*     char Reserved[9];
/* } Qca_PCMD_CPOP0100_t;
*****/
record("cl_command: Command:\n");
buffer(Command, strlen(Command));
memset(cpop0100_out, 0x00, sizeof(cpop0100_out));
memset(&cpop0100, 0x00, sizeof(Qca_PCMD_CPOP0100_t));
cpop0100.Command_Process_Type = 0;
cpop0100.DBCS_Data_Handling = '0';
cpop0100.Prompter_Action = '0';
cpop0100.Command_String_Syntax = '0';
memset(&esErrCode, 0x00, sizeof(esErrCode));

```

Figure 302. Sample WSG Server Logon Exit Program (Part 45 of 50)

```

esErrCode.Bytes_Provided = 0L; /* Force exception to be signalled */
QCAPCMD(Command, strlen(Command), &cpop0100,
        sizeof(Qca_PCMD_CPOP0100_t), "CPOP0100", cpop0100_out,
        sizeof(cpop0100_out) - 1, &cpop0100_len, &esErrCode);
return;
}

/* Function Specification *****/
/*
/* Function Name: Pad
/*
/* char *pszString - null terminated string, may or may not have
/*                 blanks
/*
/* int iLength - maximum length to pad the string (includes NULL
/*              terminator). A NULL terminator WILL be added!
/*
/*              iLength should be where you want the NULL term.
/*
/* Descriptive Name: pads a string with blanks
/*
/* End Function Specification *****/

```

Figure 302. Sample WSG Server Logon Exit Program (Part 46 of 50)

```

int Pad(char *pszString, int iLength)
{
    int iLen = strlen(pszString);

    if (iLength <= iLen) {
        return(iLen);
    } /* endif */
    /******
    /* Start at the end of the string and add blanks.          */
    /******
    while (iLen < iLength) {
        pszString[iLen] = ' ';
        iLen++;
    } /* endwhile */
    /******
    /* Add the null terminator to make it a 'C' string.        */
    /******

```

Figure 302. Sample WSG Server Logon Exit Program (Part 47 of 50)

```

    pszString[iLen] = 0x00;
    iLen = strlen(pszString);

    return(iLen);
}

/* Function Specification *****/
/*                               */
/* Function Name: Trim           */
/*                               */
/* char *pszString - null terminated string, may have trailing */
/*                   blanks                                             */
/*                               */
/* int iLength   - start position of the string to be trimmed. This */
/*                 need not be the length of the string. If you are */
/*                 trimming a 'C' string, specify the position of any */
/*                 NULL terminator.                                     */
/*                               */
/*                               */
/*                   iLength should be where you want the NULL term. */
/*                               */
/*                               */
/*                 Thus, iLength=10 means array position 10, not 9. */
/*                               */
/* Descriptive Name: pads a string with blanks                         */
/*                               */
/* End Function Specification *****/

```

Figure 302. Sample WSG Server Logon Exit Program (Part 48 of 50)

Notices

This information was developed for products and services offered in the U.S.A. IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described 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:

IBM Director of Licensing
IBM Corporation
500 Columbus Avenue
Thornwood, NY 10594
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation

Software Interoperability Coordinator
3605 Highway 52 N
Rochester, MN 55901-7829
U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this information and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement or any equivalent agreement between us.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrates programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. You may copy, modify, and distribute these sample programs in any form without payment to IBM for the purposes of developing, using, marketing, or distributing application programs conforming to IBM's application programming interfaces.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

Programming Interface Information

This publication is intended to help you to use the TCP/IP function with the IBM AS/400 system. This publication documents General-Use Programming Interface and Associated Guidance Information provided by TCP/IP Connectivity Utilities for AS/400 licensed program and the OS/400 licensed program.

General-Use programming interfaces allow the customer to write programs that obtain the services of the TCP/IP Utilities licensed program and the OS/400 licensed program.

Trademarks

The following terms are trademarks of the IBM Corporation in the United States, or other countries, or both:

400
Advanced Function Printing
AFP
AIX
AnyNet
Application System/400
APPN
AS/400
AS/400e
AT

C/400
CICS/400
Client Access
CT
DB2
Distributed Relational Database Architecture
DRDA
IBM
IBM Global Network
Integrated Language Environment
Intelligent Printer Data Stream
IPDS
Netfinity
Network Station
OfficeVision
OfficeVision/400
Operating System/400
OS/2
OS/400
Print Services Facility
Proprinter
RISC System/6000
RPG/400
RS/6000
S/390
SecureWay
SP
System/36
System/38
System/370
System/390
ThinkPad
WebExplorer

Microsoft, Windows, Windows NT, and the Windows logo are registered trademarks of Microsoft Corporation in the United States, other countries, or both.

Java and all Java-based trademarks and logos are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

UNIX is a registered trademark in the United States, other countries, or both and is licensed exclusively through X/Open Company Limited.

Lotus Notes is a registered trademark, and Notes and Domino are trademarks of Lotus Development Corporation in the United States, other countries, or both.

Other company, product, and service names may be trademarks or service marks of others.

Bibliography

The publications listed in this bibliography provide additional information about topics described or referred to in this guide. The following publications are listed with their full title, short title, and order number. When these manuals are referred to in text, a shortened version of the title is used.

Client Access Books

These books provide information for planning and installing Client Access, and configuring and diagnosing problems for individual Client Access users.

- *Client Access for Windows 95/NT - Setup*, SC41-3512-05.
- *Client Access Express for Windows Host Servers*, SC41-5740-03.

Communications Manuals

- *Communications Configuration*, SC41-5401-00
Cool Title About the AS/400 and the Internet, SG24-4815. Explains how to attach and use the Internet (or Intranet) from AS/400 by discussing Wide Area Network (WAN), Local Area Network (LAN), and both dial-out and dial-in SLIP connections. This book also describes some traditional TCP/IP applications, such as e-mail with MIME and POP3 enhancements, Telnet, FTP with new APIs to provide anonymous access, Gopher, as well as AS/400 security issues. It also provides implementation details and example programs for topics such as CGI programming. provides an APPC over TCP/IP configuration example.

Describes the objects, commands, and parameters used to configure OS/400 communications. It includes a general discussion of the objects and methods used to configure communications and detailed descriptions of all parameters that can be specified for the commands used to create the configuration objects.

- *Communications Management*, SC41-5406-02. Contains information on working with communications status, errors, performance, line speed, and storage requirements.
- *ICF Programming*, SC41-5442-00. Provides the information needed to write application programs that use AS/400 communications and the ICF file.
- *Internetwork Packet Exchange (IPX) Support*, SC41-5400-00. Tells you how to configure and use the Novell protocol suite with OS/400. These protocols include: Internetwork Packet Exchange (IPX), Sequenced Packet Exchange (SPX), Service Advertising Protocol (SAP), and Router Information Protocol (RIP). This book is written for the network administrator.
- *LAN, Frame-Relay and ATM Support*, SC41-5404-01. Describes the AS/400 support for IBM token-ring, distributed data interface (DDI), Ethernet, and wireless local area networks and frame relay wide area networks. This manual also includes information about AS/400 functions used for problem determination and management of local area networks.
- *OS/400 Network File System Support*, SC41-5714-01. Describes both the basic concepts of the Network File System as well as the specific applications of NFS on the AS/400. Included are descriptions of what NFS is, what NFS does, and explicit details of NFS procedures.
- *IBM Network Station Manager for AS/400*, SC41-0632-01. Provides information for installing and administering the IBM Network Station Manager for AS/400.
- *Printer Device Programming*, SC41-5713-03. Provides information to help you understand and control printing. It provides specific information on printing elements and concepts of the AS/400 system, printer file and print spooling support for printing operations, and printer connectivity.
- *Remote Work Station Support*, SC41-5402-00. Provides information for setting up and using remote work station support, such as display station pass-through, distributed host command facility, and 3270 remote attachment.
- *SNA Distribution Services*, SC41-5410-01. Provides the system operator or system administrator with information about configuring a network for Systems Network Architecture distribution services (SNADS) and VM/MVS bridge. Object distribution and the system distribution directory are also discussed.
- *Simple Network Management Protocol (SNMP) Support*, SC41-5412-00. Provides information for configuring an AS/400 system to use Simple Network Management Protocol (SNMP).

- *Sockets Programming*, SC41-5422-03. Provides programming information for using the sockets programming interface for the AS/400 system.
- *Workstation Customization Programming*, SC41-5605-00. Provides information for using the workstation customizing function.
- *X.25 Network Support*, SC41-5405-01. Provides information on how to use the AS/400 X.25 support. Descriptions of various connection methods, diagnostic information, and configuration examples are included.
- *3270 Device Emulation Support*, SC41-5408-00. Provides the display station operator or the programmer who uses the OS/400 binary synchronous communications (BSC) and Systems Network Architecture (SNA) 3270 device emulation with information on 3270 device emulation. Also included are the differences between the 5250 keyboard and the 3278 keyboard.

Integrated Netfinity Server Manuals

- *Integration Services for Integrated PC Server*, SC41-5123-00. This book explains how to install, configure, and use the Integration Services for Integrated Netfinity Server feature of OS/400. This support enables you to use the Integrated Netfinity Server as an Ethernet or token-ring LAN adapter. This support also allows you to install and run applications such as Lotus Notes, NetWare, Warp Server and Windows NT Server on the Integrated Netfinity Server.

Internet Connection Server Manuals

- *HTTP Server for AS/400 Quick Beginnings*, GC41-5433-01 explains how to plan for, install, and start your server by using the default configuration settings and how to stop your server. It also explains how to view online help and print online books.
- *HTTP Server for AS/400 Webmaster's Guide* explains how to change the default configuration settings to meet your needs, either by using the built-in configuration utility or by editing the configuration file. It also explains how to control and track user's access to your server, how to include dynamic information in

the files your server returns to users, and how to set up a secure environment for your users to conduct business.

- *Internet Connection Server for AS/400 Web Programming Guide* is available online at <http://www.ics.raleigh.ibm.com/pub/icswpg.htm>. See this article for information on writing external programs that interact with the Internet Connection Secure Server by way of the Common Gateway Interface (CGI). For example, you can write a CGI program to generate a dynamic response to user input.

Programming Manuals

- *DDS Reference*, SC41-5712-01. Provides information about using DDS to create and maintain displays for applications, creating and working with display files on the system.
- *Application Display Programming*, SC41-5715-00. Contains information about managing files, and creating job queues and output queues.
- *CL Programming*, SC41-5721-02. Provides the application programmer or programmer with a wide-ranging discussion of AS/400 programming topics, such as a general discussion of objects and libraries, control language (CL) programming, messages and message handling, user-defined commands and menus, and application testing.
- *CL Reference (Abridged)*, SC41-5722-03. Provides the application programmer or system programmer with a description of the AS/400 control language (CL) and its commands. Command descriptions include a syntax diagram, parameters, default values and keywords.
- *ILE C for AS/400 Programmer's Guide*, SC09-2712-01. This book documents general-use programming interfaces and associated guidance information provided by the Integrated Language Environment C for AS/400 compiler.
- *System API Reference*, SC41-5801-03. Provides information for the experienced programmer on how to use the application programming interfaces (APIs) to such OS/400 functions as:
 - Dynamic Screen Manager
 - Files (database, spooled, hierarchical)
 - Message handling
 - National language support

- Network management
- Objects
- Problem management
- Registration facility
- Security
- Software products
- Source debug
- UNIX-type
- User-defined communications
- User interface
- Work management
- *AFP Utilities/400 User's Guide*, SH18-2416
- *TCP/IP for OS/2 Version 2.0 User's Guide*, SC31-6076

Security

- *IBM SecureWay AS/400 and the Internet*, G325-6321. This booklet discusses security considerations when connecting AS/400 to the Internet.
- *Security - Basic*, SC41-5301-00. Provides information for the entry-level system administrator to plan and put basic security and system tailoring into effect.
- *IBM Firewall for AS/400 Administrator's Guide*
- *Getting Started with IBM Firewall for AS/400*, SC41-5424-02. Provides information on how to install, configure, and administer the IBM Firewall for AS/400.
- *Tips and Tools for Securing Your AS/400*, SC41-5300-03. Describes how to use AS/400 functions and security tools commands to protect your system. Includes a chapter on securing TCP/IP communications.
- *IBM 8209 LAN Bridge Customer Information*, SA21-9994. Describes how to set up and use the 8209 Local Area Network Bridge.
- *International Application Development*, SC41-5603-01. Provides information required to understand and use the national language support function on the AS/400 system. This manual prepares the AS/400 user for planning and using the national language support (NLS) and the multilingual support of the AS/400 system.
- *System Operation for New Users*, SC41-3200-00. Provides display station operators with information about how to sign on and off; send and receive messages, respond to keyboard error messages; use function keys; and control and manage their own jobs. Also included is a description of keyboard differences.
- *System Operation*, SC41-4203-00. Provides the system operator or system administrator with information on how to respond to error messages and process and manage jobs on the system. Processing jobs includes working with spooled files and finding your printer output.
- *Work Management*, SC41-5306-03. Provides programmers with information about how to effectively manage their system workload by changing work management objects to meet their needs. The publication provides guidelines for performance tuning; descriptions of system values; and information on collecting performance data, gathering system use data, using work entries, and scheduling batch jobs.

Systems Network Architecture (SNA) Display Stations

The following manual contains information about the SNA devices that communicate, as display stations, with the AS/400 system. Refer to this manual for specific information about SNA devices.

- *3270 Information Display System: 3274 Control Unit Description and Programmer's Guide*, GA23-0061

System Manuals

- *AS/400e Handbook*, GA19-5486-17. Provides information about all aspects of AS/400e. Includes descriptions of basic AS/400 system concepts, models, hardware components, and licensed programs.
- *Local Device Configuration*, SC41-5121-00. Provides the system operator or system administrator with information on how to do an initial hardware configuration and how to change that configuration. Also included is a description of the different keyboard language types. Keyboard language types are specified when using the TELNET function.

Request For Comments (RFC)

An RFC is a formal specification of a portion of TCP/IP. Research ideas and new protocols (mostly application protocols) are brought to the attention of the Internet community in the form of an RFC. Some protocols are so useful that they are recommended to be implemented in all future implementations of TCP/IP; that is, they become recommended protocols. Each RFC has a status attribute to indicate the acceptance and stage of evolution this idea has in the TCP/IP protocol suite.

RFCs are generally very detailed and technical documents. RFCs may be purchased from:

Government Systems, Inc.
Attn: Network Information Center
14200 Park Meadow Drive
Suite 200
Chantilly, VA 22021
1-800-365-3642

The following RFCs were considered during the development of TCP/IP for the AS/400 system:

- RFC 768, *User Datagram Protocol*
- RFC 791, *Internet Protocol*
- RFC 792, *Internet Control Message Protocol*
- RFC 793, *Transmission Control Protocol*
- RFC 813, *Window And Acknowledgement Strategy In TCP*
- RFC 815, *IP Datagram Reassembly Algorithms*
- RFC 816, *Fault Isolation and Recovery*
- RFC 821, *Simple Mail Transfer Protocol*
- RFC 822, *Standard For The Format Of ARPA Internet Text Messages*
- RFC 826, *An Ethernet Address Resolution Protocol*
- RFC 854, *TELNET Protocol Specification*
- RFC 855, *TELNET Option Specifications*
- RFC 879, *The TCP Maximum Segment Size and Related Topics*
- RFC 894, *A Standard for the Transmission of IP Datagrams over Ethernet Networks*
- RFC 917, *Internet Subnets*
- RFC 919, *IP Broadcast Datagrams*
- RFC 922, *Broadcasting Internet Datagrams in the Presence of Subnets*
- RFC 950, *Internet Standard Subnetting Procedure*

- RFC 959, *File Transfer Protocol (FTP)*
- RFC 974, *Mail Routing And The Domain System*
- RFC 1034, *Domain Names - Concepts And Facilities*
- RFC 1035, *Domain Names - Implementation And Specification*
- RFC 1042, *A Standard for the Transmission of IP Datagrams over IEEE 802 Networks*
- RFC 1055, *Nonstandard for transmission of IP datagrams over serial lines: SLIP*
- RFC 1122, *Requirements For Internet Hosts—Communication Layers*
- RFC 1123, *Requirements For Internet Hosts—Application And Support*
- RFC 1179, *Line Printer Daemon Protocol*
- RFC 1213, *Management Information Base for Network Management of TCP/IP-Based Internets MIB-II*
- RFC 1349, *Type Of Service In The Internet Protocol Suite*
- RFC 1349, *Type of Service in the Internet Protocol Suite*
- RFC 1572, *Telnet Environment Option*
- RFC 1600, *Internet Official Protocol Standards*
- RFC 1635, *How to Use Anonymous FTP*
- RFC 1661, *The Point-to-Point Protocol (PPP)*
- RFC 1662, *PPP in HDLC-like Framing*
- RFC 1700, *Assigned Numbers*
- RFC 1725, *Post Office Protocol - Version 3*
- RFC 1738, *Uniform Resource Locators (URL)*

Other Information

A good place to begin looking for online AS/400 information is the Information Center. See "TCP/IP Topics in the Information Center" on page xv for a list of TCP/IP-related topics that are in the Information Center and the URL to access it online.

Military Standards are approved for use by all departments and agencies of the United States Department of Defense.

You can address requests for Military Standards to:

Defense Communications Agency
Attn: J110
1860 Wiehle Avenue
Reston, Virginia 22090

The following is a list of Military Standards that may be of interest to you.

- MIL-STD-1778 for information about the TCP protocol
- MIL-STD-1777 for information about the IP protocol

Index

Special Characters

- *AIX format 73
- *ANS configuration profile 132
- *AS400 format 73
- *BASE pool size 425
- *CTLQ value 202
- *CTLS value 202
- *IOSYSCFG authority 30
- *NIC format 73
- *PPP, PPP/SLIP over 156
- *SYSGEN default value 30, 506
- (Transmission Control Protocol/Internet Protocol)
 - attributes 36
 - command authority for users
 - QPGMR 509
 - QSECOFR 509
 - QSRV 509
 - QSRVBAS 509
 - QSYSOPR 509
 - routing 12
 - software 12

Numerics

- 3270 full-screen mode
 - handling null characters 162
 - mapping table 523
 - messages 162
 - screen size 161
 - setting up 183
 - Start TCP/IP Telnet (STRTCPTELN) command
 - applicable parameters 160
 - Telnet server 186
- 3270 mapping tables
 - creating a source member
 - incoming data 521
 - outgoing data 522
 - used by TELNET 521
- 3278-2-E terminal type 195
- 5250 data streams
 - converting to HTML 319
- 5250 full-screen mode
 - screen size 159
 - setting up 183
 - Start TCP/IP Telnet (STRTCPTELN) command
 - applicable parameters 159
- 5250 keyboard mapping
 - from AIX TN3270 533
 - from DOS TN3270 (PC/TCP) 532
 - from OS/2 TN3270 (PMANT) 532
 - from VAX/MVS 533

A

- AA entry 506
- access log (ACCCLOG) parameter 330
- access logging (ACCCLOG) parameter 327

- accessing
 - Internet 3
 - remote system 152
 - SST 494
- ACCCLOG (access log) parameter 330
- ACCCLOG (access logging) parameter 327
- active jobs
 - working with 138
- adapter
 - how to find installed 128
- adapters
 - SLIP, supported 128
- Add Authorization List Entry (ADDAUTLE)
 - command 147
- Add TCP/IP Interface (ADDTCPIFC) command 394
- Add TCP/IP Remote System Information (ADDTCPRSI)
 - command 37
- Add TCP/IP Route (ADDTCPRTE) command 34
- Add Work Station Entry (ADDWSE) command 233
- ADDAUTLE (Add Authorization List Entry)
 - command 147
- adding
 - authorization list entry 147
 - configuration profile for SLIP
 - dial-in 133
 - dial-out 133
 - default routes 35
 - Network Stations to DHCP 420
 - TCP/IP remote system information 37
 - TCP/IP route 32
- address
 - broadcast 8
 - character restrictions 10
 - class A networks 4
 - class B networks 4
 - class C networks 4
 - class D networks 4
 - class E networks 4
 - extension to Internet addressing 6
 - fixed-length 19
 - hardware 19
 - internet 2
 - Internet 3
 - internet 18, 19
 - assigning 2
 - physical hardware 2
- address book, POP mail server 311
- cache 312
- configuring 294
- refresh interval 294
- address mapping
 - IP addresses 232
- Address Resolution Protocol (ARP) 19
- address types
 - supported by MAPI service providers 311
- ADDTCPIFC (Add TCP/IP Interface) command 394
- ADDTCPRSI (Add TCP/IP Remote System Information)
 - command 37

- ADDDTCPRTE (Add TCP/IP Route) command 34
- ADDWSE (Add Work Station Entry) command 233
- Advanced Function Printing data stream 350
- AFPDS
 - see Advanced Function Printing data stream 350
- AIX
 - configuring device and virtual printer 356
- AIX (UNIX) example
 - FTP client
 - example 254
 - logon process 254
 - naming conventions 255
- anonymous FTP 568
- ANSWERBACK (VT100 answerback feature)
 - parameter 166
- AnyNet/400 19
- API (application program interface) 15
 - Pascal 16
- APPC
 - accessing using IPX 20
 - accessing using TCP/IP 20
 - problem analysis 492
 - tracing 493
- application
 - exit points 536
 - exit point interface 547
 - exit programs 535, 551
- application program interface (API) 15
 - Pascal 16
- application request validation
 - exit point interface 547
- application server logon
 - exit point interface 551
- ARP (Address Resolution Protocol) 19
 - support in SLIP 144
- AS/400
 - adding Network Stations to DHCP 420
- AS/400 file pre-creation 264
- AS/400 sign-on panel 325
 - changing the display file 326
- ASCII character set
 - table 526
- ASCII client mail 299
- ASCII files
 - printing files converted to EBCDIC 373
- ASCII line drawing character set 530
- ASCII line mode considerations
 - keyboard mapping 215
 - setting up 183
 - Telnet server 210
- ASCII mapping table
 - message CPX8416 517
 - used by FTP 518
 - used by TELNET 518
- ASCII operating mode (ASCOPRMOD) parameter 166
- ASCII-to-EBCDIC conversion 328
 - POP mail server 313
- ASCII-to-EBCDIC mapping tables 519
- ASCOPRMOD (ASCII operating mode) parameter 166
- assigning addresses 2, 3

- asynchronous line
 - configuring for SLIP 152
 - description 151
- attribute
 - character 166
 - supported by VTxxx terminal 166
- TCP/IP
 - Change TCP/IP Attributes (CHGTCPA)
 - command 36
 - IP datagram forwarding 36
- authority 190, 199, 213
 - *IOSYSCFG 30
 - for users
 - QPGMR authority 509
 - QSECOFR authority 509
 - QSRV authority 509
 - QSRVBAS authority 509
 - QSYSOPR authority 509
 - putting spooled files on output queues 372
- authorization list
 - creating 146, 147
 - SLIP 146
- authorization list entry
 - adding 147
- automatic configuration
 - creating controllers 505
 - virtual devices
 - 3270 full-screen mode 190
 - ASCII line mode 213
 - VTxxx full-screen mode 199
- automatic wrap 207
- automatically create virtual (devices) (QAUTOVRT)
 - parameter
 - 3270 full-screen mode 188
 - ASCII line mode 212
 - VTxxx full-screen mode 197
- AUTOSTART parameter 320, 378, 383, 392, 400, 415

B

- batch
 - running FTP in 235, 279
- batch job, FTP 269
- BDATA (broadcast data) packet 389
- binding, route-to-interface 58
- blksize option 387, 388
- block# BDATA packet field 389
- BLOCK parameter 396
- books
 - Client Access 599
 - communications 599
 - Integrated Netfinity Server 600
 - internet connection server 600
 - programming 600
 - security 601
 - system 601
- BOOTP (Bootstrap Protocol) server
 - accessing through Operations Navigator 377
 - adding IBM Network Stations 379
 - automatically starting 378
 - changing attributes 379
 - CL commands 511

BOOTP (Bootstrap Protocol) server *(continued)*

- configuring 377
 - definition 13
 - ending 378
 - overview 377
 - starting 378
 - working with BOOTP table 379
- BOOTP (Bootstrap Protocol) table 379
- Bootstrap Protocol (BOOTP) server
- accessing through Operations Navigator 377
 - adding IBM Network Stations 379
 - automatically starting 378
 - changing attributes 379
 - CL commands 511
 - configuring 378
 - definition 13
 - ending 378
 - overview 377
 - starting 378
 - working with BOOTP table 379
- BOTBNRURL (bottom banner URL) parameter 327
- bottom banner URL 327
- bottom banner URL (BOTBNRURL) parameter 327
- broadcast address
- directed 8
 - limited 8
- broadcast data (BDATA) packet 389
- broadcast host ID 8
- browser
- online help 338
- buttons
- WSG server 339

C

- calculating FFT values 301
- CCSID (coded character set identifier) 517
- code page tagging
 - when transferring files using FTP 264
 - description 316
 - FTP client 265
 - Latin-1 CCSID 266, 280
 - parameter 166
 - selecting character mapping for Telnet 179
- CCSID (coded character set identifier) parameter 328
- CFGTCP (Configure TCP/IP) command 27
- CFGTCPAPP (Configure TCP/IP Applications) command 378, 392
- CFGTCPBP (Configure TCP/IP BOOTP) command 378
- CFGTCPLPD (Configure TCP/IP Line Printer Daemon) command 366
- CFGTCPRTD (Configure TCP/IP RouteD) command 392, 397
- CFGTCPWSG (Configure TCP/IP Workstation Gateway Server) command 321, 323
- CFGWSF (Configure Work Station Function) command 161
- Change BOOTP Attributes (CHGBPA) command 379
- Change DHCP Attributes (CHGDHCPA) command 416
- AUTOSTART parameter 416
- Change Group Attributes (CHGGRPA) command 228
- change how nulls are handled (NULLS) parameter 160
- Change Keyboard Map (CHGKBDMAP) command 193
- Change Keyboard Mapping (CHGKBDMAP) command 531
- Change LPD Attributes (CHGLPDA) command 363, 366
- Change Message Queue (CHGMSGQ) command 195
- Change POP Attributes (CHGPOPA) command 293
- Change Printer File (CHGPRTF) command 368
- Change REXEC Attributes (CHGRXCA) command 400
- Change RouteD Attributes (CHGRTDA) command 392, 397
- Change Subsystem Description (CHGSBSD) command 326
- Change System Value (CHGSYSVAL) command 322
- 3270 full-screen mode 188
 - ASCII line mode 212
 - VTxxx full-screen mode 197
- Change TCP/IP Attributes (CHGTCPA) command 36
- Change TCP/IP TFTP Attributes (CHGTFTPA) command 384
- Change VT Keyboard Map (CHGVMTAP) command 206
- Change Workstation Gateway Attributes (CHGWSGA) command 320, 323, 337
- changing
- 5250 data streams to Hypertext Markup Language (HTML) 319
 - DHCP server attributes 416
 - group attributes 228
 - keyboard map 193
 - keyboard style 161
 - line printer daemon attributes 363
 - LPD attributes 366
 - mapping tables 523
 - message queue 195
 - system value
 - 3270 full-screen mode 188
 - ASCII line mode 212
 - VTxxx full-screen mode 197
 - TCP/IP attributes 36
 - VTxxx keyboard map 206
 - workstation gateway attributes 323
 - workstation gateway server (WSG) 337
 - WSG (workstation gateway server) 337
- character data 172
- character restriction 10
- character set
- ASCII 526
 - ASCII-to-EBCDIC mapping 529
 - EBCDIC 525
 - EBCDIC-to-ASCII mapping 527
- Check Communications Trace (CHKCMNTRC) command 493
- checking
- communications trace 493
- CHGBPA (Change BOOTP Attributes) command 379
- CHGDHCPA (Change DHCP Attributes) command 416
- AUTOSTART parameter 416
- CHGGRPA (Change Group Attributes) command 228

CHGKBDMAP (Change Keyboard Map)
 command 193, 531

CHGLPDA (Change LPD Attributes) command 363, 366

CHGMSGQ (Change Message Queue) command 195

CHGPOPA (Change POP Attributes) command 293

CHGPRTF (Change Printer File) command 368

CHGRTDA (Change Routed Attributes) command 392, 397

CHGRXCA (Change REXEC Attributes) command 400

CHGSBSD (Change Subsystem Description)
 command 326

CHGSYSVAL (Change System Value) command
 3270 full-screen mode 188
 ASCII line mode 212
 VTxxx full-screen mode 197

CHGSYSVAL (Change System Values) command 322

CHGTCPA (Change TCP/IP Attributes) command 36

CHGTFTP (Change TCP/IP TFTP Attributes)
 command 384

CHGVTMAP (Change VT Keyboard Map)
 command 206

CHGWSGA (Change Workstation Gateway
 Attributes) 320

CHGWSGA (Change Workstation Gateway Attributes)
 command 323, 337

China
 CCSID when MIME page not supported for WSG
 server 328, 337

CHKCMNTRC (Check Communications Trace)
 command 493

class A networks 4

class B networks 4

class C networks 4

class D networks 4

class E networks 4

classes of networks 4

client
 Telnet
 introduction 159

Client Access
 books 599

Client Access-based
 mail clients, setting up 293
 mail users
 configuring POP for 294

Client Access for Windows 95/NT
 POP server, capabilities with 287
 POP server, installing for use with 288

client port
 TFTP 385

coded character set identifier (CCSID) 517
 description 316
 FTP client 265
 Latin-1 CCSID 266, 280
 parameter 166
 selecting character mapping for Telnet 179

coded character set identifier (CCSID) parameter 328

collecting
 communications trace 493

command, CL
 Add Authorization List Entry (ADDAUTLE) 147
 Add TCP/IP Interface (ADDTCPIFC) 394
 Add TCP/IP Remote System Information
 (ADDTCPRSI) 37
 Add TCP/IP Route (ADDTCPRTE) 34
 Add Work Station Entry (ADDWSE) 233
 ADDAUTLE (Add Authorization List Entry) 147
 ADDTCPIFC (Add TCP/IP Interface) 394
 ADDTCPRSI (Add TCP/IP Remote System
 Information) 37
 ADDTCPRTE (Add TCP/IP Route) 34
 ADDWSE (Add Work Station Entry) 233
 BOOTP 511
 CFGTCPAPP (Configure TCP/IP Applications) 378,
 392
 CFGTCPBP (Configure TCP/IP BOOTP) 378
 CFGTCPLPD (Change TCP/IP LPD) 366
 CFGTCPRTD (Configure TCP/IP Routed) 392, 397
 CFGTCPWSG (Configure TCP/IP Workstation
 Gateway Server) 321, 323
 Change BOOTP Attributes (CHGBPA) 379
 Change DHCP Attributes (CHGDHCPA) 416
 Change Group Attributes (CHGGRPA) 228
 Change Keyboard Map (CHGKBDMAP) 193, 531
 Change LPD Attributes (CHGLPDA) 363, 366
 Change Message Queue (CHGMSGQ) 195
 Change Printer File (CHGPRTF) 368
 Change REXEC Attributes (CHGRXCA) 400
 Change Routed Attributes (CHGRTDA) 392, 397
 Change Subsystem Description (CHGSBSD) 326
 Change System Value (CHGSYSVAL) 322
 3270 full-screen mode 188
 ASCII line mode 212
 VTxxx full-screen mode 197
 Change TCP/IP Attributes (CHGTCPA) 36
 Change TCP/IP LPD (CFGTCPLPD) 366
 Change TCP/IP TFTP Attributes (CHGTFTP) 384
 Change VT Keyboard Map (CHGVTMAP) 206
 Change Workstation Gateway Attributes
 (CHGWSGA) 320, 323, 337
 Check Communications Trace (CHKCMNTRC) 493
 CHGBPA (Change BOOTP Attributes) 379
 CHGDHCPA (Change DHCP Attributes) 416
 CHGGRPA (Change Group Attributes) 228
 CHGKBDMAP (Change Keyboard Map) 193, 531
 CHGLPDA (Change LPD Attributes) 363, 366
 CHGMSGQ (Change Message Queue) 195
 CHGPRTF (Change Printer File) 368
 CHGRTDA (Change Routed Attributes) 392, 397
 CHGRXCA (Change REXEC Attributes) 400
 CHGSBSD (Change Subsystem Description) 326
 CHGSYSVAL (Change System Value) 322
 3270 full-screen mode 188
 ASCII line mode 212
 VTxxx full-screen mode 197
 CHGTCPA (Change TCP/IP Attributes) 36
 CHGTFTP (Change TCP/IP TFTP Attributes) 384
 CHGVTMAP (Change VT Keyboard Map) 206
 CHGWSGA (Change Workstation Gateway
 Attributes) 320, 323, 337

command, CL *(continued)*

CHKCMNTRC (Check Communications Trace) 147
 CL commands 509
 Configure TCP/IP Applications (CFGTCPAPP) 378, 392
 Configure TCP/IP BOOTP (CFGTCPBP) 378
 Configure TCP/IP RouteD (CFGTCPRTD) 392, 397
 Configure TCP/IP Workstation Gateway Server (CFGTCPWSG) 323
 Configure TCP/IP Workstation Gateway Server (CFGTCPWSG) command 321
 Create Authorization List (CRTAUTL) 146, 147
 Create Controller Description (Network) (CRTCTLNET) 505
 Create Controller Description (Virtual Work Station) (CRTCTLVWS) 215
 Create Device Description (Display) (CRTDEVDSP) 215
 Create Device Description (Network) 505
 Create Output Queue (CRTOUTQ) 352
 Create Table (CRTTBL) 518, 521
 Create User Profile (CRTUSRPRF)
 3270 full-screen mode 191
 ASCII line mode 215
 VTxxx full-screen mode 201
 CRTAUTL (Create Authorization List) 146, 147
 CRTCTLNET (Create Controller Description (Network)) 505
 CRTCTLVWS (Create Controller Description (Virtual Work Station)) 215
 CRTDEVDSP (Create Device Description (Display)) 215
 CRTDEVNET (Create Device Description (Network)) 505
 CRTOUTQ (Create Output Queue) 352
 CRTTBL (Create Table) 518, 521
 CRTUSRPRF (Create User Profile)
 3270 full-screen mode 191
 ASCII line mode 215
 VTxxx full-screen mode 201
 Delete Communications Trace (DLTCMNTRC) 493
 Delete Table (DLTTBL) 523
 Display Keyboard Map (DSPKBDMAP) 192
 Display VT Keyboard Map (DSPVTMAP) 205
 DLTCMNTRC (Delete Communications Trace) 493
 DLTTBL (Delete Table) 523
 DSPKBDMAP (Display Keyboard Map) 192
 DSPVTMAP (Display VT Keyboard Map) 205
 End Communications Trace (ENDCMNTRC) 493
 End Group Job (ENDGRPJOB) 229
 End TCP/IP (ENDTCP) 46
 End TCP/IP Connections (ENDTCPCNN) 63
 End TCP/IP Server (ENDTCPSVR) 320, 367, 378, 384, 392, 400, 416
 ENDCMNTRC (End Communications Trace) 493
 ENDGRPJOB (End Group Job) 229
 ENDTCP (End TCP/IP) 46
 ENDTCPCNN (End TCP/IP Connections) 63
 ENDTCPSVR (End TCP/IP Server) 320, 367, 378, 384, 392, 400, 416
 FTP 510

command, CL *(continued)*

Grant Object Authority (GRTOBJAUT)
 3270 full-screen mode 189
 ASCII line mode 213
 VTxxx full-screen mode 199
 GRTOBJAUT (Grant Object Authority)
 3270 full-screen mode 189
 ASCII line mode 213
 VTxxx full-screen mode 199
 HTTP Server 512
 LPD 511
 LPR 345
 NETSTAT (Network Status) 436
 Network Status (NETSTAT) 436
 POP Mail Server 511
 Print Communications Trace (PRTCMNTRC) 493
 PRTCMNTRC (Print Communications Trace) 493
 Remove Exit Program (RMVEXITPGM) 540
 Revoke Object Authority (RVKOBJAUT) 514
 REXEC 512
 RMVEXITPGM (Remove Exit Program) 540
 RouteD 512
 RVKOBJAUT (Revoke Object Authority) 514
 Send TCP/IP Spooled File (SNDTCPSPLF) 345
 Set Keyboard Map (SETKBDMAP) 193
 Set VT Keyboard Map (SETVTMAP) 206
 Set VT Mapping Tables (SETVTTBL) 207
 SETKBDMAP (Set Keyboard Map) 193
 SETVTMAP (Set VT Keyboard Map) 206
 SETVTTBL (Set VT Mapping Tables) 207
 SMTP 511
 SNDTCPSPLF (Send TCP/IP Spooled File) 345
 Start Communications Trace (STRCMNTRC) 493
 Start PDM (STRPDM) 330
 Start Remote Writer (STRRMTWTR) 355
 Start Source Entry Utility (STRSEU) 518
 Start TCP/IP (STRTCP) 43, 337
 Start TCP/IP Server (STRTCPSVR) 320, 366, 378, 383, 391, 399, 416
 Start TCP/IP Telnet (STRTCPTLN) 220
 STRCMNTRC (Start Communications Trace) 493
 STRPDM (Start PDM) 330
 STRRMTWTR (Start Remote Writer) 355
 STRSEU (Start Source Entry Utility) 518
 STRTCP (Start TCP/IP) 43, 337
 STRTCPSVR (Start TCP/IP Server) 320, 366, 378, 383, 391, 399, 416
 STRTCPTLN (Start TCP/IP Telnet) 220
 TELNET 510
 TFRGRPJOB (Transfer to Group Job) 228
 TFTP 512
 Trace TCP/IP Application (TRCTCPAPP) 455
 Transfer to Group Job (TFRGRPJOB) 228
 TRCTCPAPP (Trace TCP/IP Application) 455
 Verify TCP/IP Connection (VFYTCPCNN) 47
 VFYTCPCNN (Verify TCP/IP Connection) 47
 Work with Active Jobs (WRKACTJOB) 138, 363, 434
 Work with BOOTP Table (WRKBPTBL) 379
 Work with Configuration Status (WRKCFGSTS) 229

- command, CL *(continued)*
 - Work with Hardware Resources (WRKHDWRSC) 147
 - Work with Registration Information (WRKREGINF) 325, 334, 537
 - Work with Routed Configuration (WRKRTDCFG) 392
 - Work with Server Table (WRKSVRTBLE) 333
 - Work with Spooled Files (WRKSPLF) 139
 - Work with Tables (WRKTBL) 530
 - Work with TCP/IP Network Status (WRKTCPSTS) 55
 - Work with TCP/IP Point-to-Point (WRKTCPPTP) 135, 137
 - Work with TCP/IP Status (WRKTCPSTS) 436
 - workstation gateway server 512
 - WRKACTJOB (Work with Active Jobs) 138, 363, 434
 - WRKBPTBL (Work with BOOTP Table) 379
 - WRKCFGSTS (Work with Configuration Status) 229
 - WRKHDWRSC (Work with Hardware Resources) 128
 - WRKREGINF (Work with Registration Information) 325, 334, 537
 - WRKRTDCFG (Work with Routed Configuration) 392
 - WRKSPLF (Work with Spooled Files) 139
 - WRKSVRTBLE (Work with Server Table) 333
 - WRKTBL (Work with Tables) 530
 - WRKTCPPTP (Work with TCP/IP Point-to-Point) 135, 137
 - WRKTCPSTS (Work with TCP/IP Network Status) 55
 - WRKTCPSTS (Work with TCP/IP Status) 436
- command, Client Access
 - CFGWSF (Configure Work Station Function) 161
 - Configure Work Station Function (CFGWSF) 161
 - Work Station Function Keys (WSFKEYS) 161
 - WSFKEYS (Work Station Function Keys) 161
- command, TCP/IP
 - Network Status (NETSTAT) 55
 - Packet Internet Groper (PING) 47
 - Telnet 159
- command processing selection
 - exit point interface 551
- communications manuals 599
- communications trace
 - checking 493
 - collecting 493
 - deleting 493
 - display 495
 - ending 493
 - formatting 499
 - printing 493
 - saving 499
 - starting 493, 495
 - stopping 498
 - verifying contents 501
- compression, header
 - SLIP dial-out profile 149
- configuration 509
 - configuration 190 *(continued)*
 - 3270 full-screen mode 190
 - ASCII line mode 213
 - VTxxx full-screen mode 199
 - configuration profile
 - SLIP 132
 - SLIP, dial-out 147
 - configuration status
 - working with 229
 - Configure TCP/IP (CFGTCP) command 27
 - Configure TCP/IP Applications (CFGTCPAPP) command 378, 392
 - Configure TCP/IP BOOTP (CFGTCPBP) command 378
 - Configure TCP/IP Routed (CFGTCPRTD) command 392, 397
 - Configure TCP/IP Workstation Gateway Server (CFGTCPWSG) command 321, 323
 - Configure Work Station Function (CFGWSF) command 161
 - configuring
 - asynchronous line for SLIP 152
 - BOOTP server 378
 - configuration profile for SLIP 133
 - Configure TCP/IP menu 27
 - default routes 35
 - device and virtual printer for AIX printing 356
 - host table 38
 - interface 30
 - IP connections 294
 - IP datagram forwarding 36
 - IPX connections 294
 - line 30
 - line description 506
 - line printer daemon 363
 - local domain and host name 42
 - LPD for a RISC System/6000 system 356
 - maximum transmission unit (MTU) 33
 - modems for SLIP 130
 - multiple-network example 22
 - multiple systems 75
 - next hop 32
 - planning for X.25 22
 - POP for Client Access-based mail users 294
 - POP mail server address book 294
 - ports 85
 - print services facility/6000 function 360
 - remote system (X.25) information 36
 - Routed scenario 393
 - Routed server 392, 393
 - routes 32
 - single-network example 29
 - SLIP planning 127
 - SNA connections 294
 - subnet mask 32
 - TCP/IP Administration menu 22
 - TCP/IP attributes 36
 - TCP/IP interfaces 30
 - TCP/IP menu 27
 - TFTP server 389
 - workstation gateway server (WSG) 321, 323

- configuring (*continued*)
 - WSG (workstation gateway server) 152, 323
- connection
 - alternatives 111
 - display TCP/IP connections 70
 - displaying totals 71
 - End TCP/IP Connections (ENDTCPCNN)
 - command 63
 - status, work with TCP/IP connection 60
 - verify TCP/IP connection 47
- connection-oriented sockets
 - SOCK_STREAM 91
- connection profiles
 - configuring connection profiles 96
 - configuring SLIP 115
 - PPP 96
- connection script 152, 156
 - dial-out
 - example 123
 - how to create 122, 125
 - location of default 121
 - purpose 125
 - SLIP 145
 - dial in 121, 124
 - dial out 121
- connection type
 - Ethernet 30
 - fiber distributed data interface (FDDI) 30
 - frame relay 30
 - shielded twisted pair distributed data interface (SDDI) 30
 - token-ring 30
 - wireless LAN 30
 - X.25 PVC 30
 - X.25 SVC 30
- considerations
 - file naming 266
 - file structure 267
 - FTP client file naming 266
 - path 267
 - security 509
 - Telnet 3270 full-screen mode
 - client 159
 - server 186
 - Telnet 5250 full-screen mode
 - server 183
 - Telnet ASCII line mode
 - server 210
 - Telnet printer pass-through mode 216
 - Telnet VTxxx full-screen mode
 - client 163
 - server 196
- control characters (CTLCHAR) parameter 166
- control file
 - overview 369
- control key keywords 172
- controller description
 - SLIP
 - automatically creating 144, 151
 - virtual workstation 215
- conventions for domain names and host names 10
- conversion
 - ASCII-to-EBCDIC 328
 - long line 299
- converting
 - data for FTP 266, 280
 - files 76
 - TCP3C14 message 266, 280
- CPF87D7 message 201
- CPF87DF message
 - 3270 full-screen mode 188
 - 5250 full-screen mode 322
 - ASCII line mode 212
 - VTxxx full-screen mode 198
- CPF8940 message
 - 3270 full-screen mode 188
 - 5250 full-screen mode 322
 - ASCII line mode 212
 - VTxxx full-screen mode 198
- CPX8416 message ID 517
- Create Authorization List (CRAUTL) command 146, 147
- Create Controller Description (Network) (CRTCTLNET) command 505
- Create Controller Description (Virtual Work Station) (CRTCTLVWS) command 215
- Create Device Description (Display) (CRTDEVDSPP) command 215
- Create Device Description (Network) (CRTDEVNET) command 505
- Create Output Queue (CRTOUTQ) command 352
- Create Table (CRTTBL) command 518, 521
- Create User Profile (CRTUSRPRF)
 - 3270 full-screen mode 191
 - ASCII line mode 215
 - VTxxx full-screen mode 201
- creating
 - ASCII mapping tables 518
 - authorization list 146, 147
 - controller description
 - virtual workstation 215
 - device description
 - display 215
 - display device description 215
 - EBCDIC mapping tables 518
 - exit point programs 537
 - line description 506
 - mapping table from source members 518
 - source members for 3270 mapping tables 521, 522
 - user-defined mapping tables 521
 - user profile
 - 3270 full-screen mode 191
 - ASCII line mode 215
 - VTxxx full-screen mode 201
 - virtual workstation controller description 215
- CRAUTL (Create Authorization List) command 146, 147
- CRTCTLNET (Create Controller Description (Network)) command 505
- CRTCTLVWS (Create Controller Description (Virtual Work Station)) command 215

CRTDEVDS (Create Device Description (Display))
 command 215
 CRTDEVNET (Create Device Description (Network))
 command 505
 CRTOUTQ (Create Output Queue) command 352
 CRTTBL (Create Table) command 518, 521
 CRTUSRPRF (Create User Profile)
 3270 full-screen mode 191
 ASCII line mode 215
 VTxxx full-screen mode 201
 CSLIP 149
 CSRSLT (cursor select key) parameter 160
 CTLCHAR (control characters) parameter 166
 cursor select key (CSRSLT) parameter 160

D

data area values 299
 data BDATA packet field 389
 data request timeout 325
 data request timeout (DTARQSTIMO) parameter 325
 data stream
 transforming 350
 data transfer methods, FTP 258
 datagram
 definition 18
 forwarding 36, 146
 SLIP 145
 User Datagram Protocol (UDP) 17
 datagram forwarding
 SLIP 145
 datagram size
 determining the maximum 507
 DDI (distributed data interface)
 fiber distributed data interface (FDDI) 22
 shielded twisted pair distributed data interface
 (SDDI) 22
 DDN (Defense Data Network) conversion algorithm 37
 DDS applications
 converting to HTML 336
 dead gateway processing 78
 default route
 adding 35
 SLIP 144, 150
 definition
 IP subnets 5
 subnetwork 6
 Delete Communications Trace (DLTCMNTRC)
 command 493
 Delete Table (DLTTBL) command 523
 deleting
 communications trace 493
 mapping table 523
 determining
 device description 229
 emulation package 229
 keyboard mapping 229
 virtual display device 229
 device description
 display 215
 SLIP
 automatically creating 144, 151

device description (*continued*)
 SLIP (*continued*)
 remote location name 144, 151
 virtual
 definition 211
 DHCP (Dynamic Host Configuration Protocol) server
 accessing through Operations Navigator 414
 adding clients 420
 adding Network Stations 420
 automatically starting 415
 changing attributes 416
 configuring through Operations Navigator
 examples 417
 creating a scoped network 409
 defining scoped statements 410
 ending 416
 exit points 417
 introduction 405
 migrating BOOTP 419
 network configuration 408
 creating a scoped network 409
 defining scoped statements 410
 options
 architected 413
 specifying 412
 user-defined 413
 overview 405
 planning 406
 relay agent 420
 starting 415
 DHCP server
 definition 15
 dial-in profile
 maximum transmission unit 143
 remote interface address 143
 SLIP, configuring 140
 dial-in support 156
 dial-on-demand
 example 98
 dial-out connection script 122
 dial-out support 156
 DIR (Directory) subcommand 283
 direct routing 12
 directed broadcast address 8
 Directory (DIR) subcommand 283
 display
 current library content 283
 format trace data 499
 start a service tool for SST 494
 start trace 496
 system service tools (SST) 494
 TCP/IP connections 70
 work with communications traces 495
 display character attributes (DSPCHRATTR)
 parameter 166
 Display Keyboard Map (DSPKBDMAP) command 192
 display sign-on panel (DSPSGN) 325
 Display VT Keyboard Map (DSPVTMAP)
 command 205
 displaying
 communications trace 501

- displaying (*continued*)
 - keyboard map 501
 - keyboard style 161
 - minus sign 161
 - QATMTLOG member 330
 - system name and address 39
 - VTxxx keyboard map 205
- DIST_ROUTES_IN parameter 395
- distributed data interface (DDI)
 - fiber distributed data interface (FDDI) 22
 - shielded twisted pair distributed data interface (SDDI) 22
- distributions received by QSECOFR
 - example 465
- DLTCMNTRC (Delete Communications Trace)
 - command 493
- DLTTBL (Delete Table) command 523
- DNS (Domain Name System) server 9
 - concepts 421
 - definition 76
 - documentation 422
 - introduction 421
 - problem analysis 482
- DNS server
 - definition 14
- domain name 42
 - character restrictions 10
 - concatenation 437
 - definition 9
- Domain Name System (DNS) server 9
 - concepts 421
 - definition 76
 - documentation 422
 - introduction 421
 - problem analysis 482
- domain names and host names 10
- DSPCHRATTR (display character attributes)
 - parameter 166
- DSPKBDMAP (Display Keyboard Map) command 192
- DSPSGN (display sign-on panel) 325
- DSPVTMAP (Display VT Keyboard Map)
 - command 205
- DTARQSTIMO (data request timeout) parameter 325
- dynamic application printing with TCP/IP 231
- Dynamic Host Configuration Protocol
 - see DHCP (Dynamic Host Configuration Protocol)
 - server 405

E

- EBCDIC
 - character set 524
 - mapping tables
 - message CPX8416 517
 - used by FTP 518
 - used by TELNET 518
- emulation package 229
- End Communications Trace (ENDCMNTRC)
 - command 493
- End Group Job (ENDGRPJOB) command 229
- end system
 - definition 220
- End TCP/IP Connections (ENDTCPCNN) command 63
- End TCP/IP Interfaces (ENDTCPIFC) command 58
- End TCP/IP Server (ENDTCPSVR) command 320, 367, 378, 384, 392, 400, 416
- ENDCMNTRC (End Communications Trace)
 - command 493
- ENDGRPJOB (End Group Job) command 229
- ending
 - BOOTP server 378
 - communications trace 493
 - DHCP server 416
 - group job 229
 - interfaces, route-to-interface binding 58
 - POP mail server 296
 - Routed server 392
 - TCP/IP connections 63
 - Telnet server session 220
- ENDTCPCNN (End TCP/IP Connections) command 63
- ENDTCPIFC (End TCP/IP Interfaces) command 58
- ENDTCPSVR (End TCP/IP Server) command 320, 367, 378, 384, 392, 400, 416
- error log
 - problem analysis 503
- error messages 438, 486
- Ethernet
 - *SYSGEN default value 506
 - AA entry 506
 - Version 2 506
- example
 - ASCII-to-EBCDIC mapping 524
 - batch FTP 269
 - CCSID when MIME page not supported for WSG server 337
 - changing keyboard mapping 193, 206
 - changing number of WSG clients 337
 - configuring DHCP for a LAN with a router 417
 - configuring DHCP through Operations Navigator 417
 - dial-on-demand 98
 - distributions received by QSECOFR 465
 - EBCDIC-to-ASCII mapping 523
 - FTP client
 - AS/400-to-AIX (UNIX) 254
 - AS/400-to-AS/400 245
 - AS/400-to-OS/2 256
 - AS/400-to-VAX/Wollongong 251
 - save file transfer from VM to AS/400 259
 - incoming data (TBLIN) 522
 - IP address mapping scenarios 232
 - multihoming 79, 80, 81, 82
 - multiple network configuration 22
 - network delay with WSG server 338
 - office-to-office scenario 100
 - outgoing data (TBLOUT) 522
 - partly successful host table merge 75
 - querying for distributions for QSECOFR 464
 - single network configuration 29
- SLIP
 - connection script (dial in) 121, 124

- example (*continued*)
 - connection script (dial out) 524
 - SLIP connection scripts
 - how to create 122
 - SLIP dial-out connection scripts 123
 - SLIP spooled file output 140
 - starting WSG server automatically 337
 - successful host table merge 75
 - Trace TCP/IP Application (TRCTCPAPP)
 - command 455
 - using DHCP to configure clients attached to a twinaxial workstation controller 418
 - using server mapping tables 337
 - using Telnet virtual device names 234
 - using X.25 permanent virtual circuit (PVC) 91
 - verifying connections
 - host name 49
 - Internet address 50
 - PING LOOPBACK 46
 - WSG server configuration 337
- exit point
 - application 536
 - description 535
 - DHCP 417
 - FTP 553
 - interface 535
 - performance 233
 - programs, creating 537
 - registration facility 535
 - REXEC server 551
 - WSG 569
- exit point interface 547
 - application request validation 547
 - application server logon 551
 - command processing selection 551
- exit program 329, 535, 551
 - adding to registration facility 537
 - application validation 548
 - description 535
 - File Transfer Protocol (FTP)
 - request validation 548
 - removing from exit point 540
 - scenario, FTP 554
- extension
 - TFTP subnet broadcast option 385
 - TFTP transfer size option 385

F

- FAX
 - support for 311
- FFDC (first failure data capture) 480
- FFT Values, calculating 301
- fiber distributed data interface (FDDI) 22
- field
 - definition 235
- file 279
 - spooled 182
- file naming considerations, FTP client 266
- file pre-creation 264

- file structure
 - FTP considerations 267
- file system
 - QLANSrv file system 235
 - QOpenSys file system 235
 - Root file system 235
- file transfer
 - of files containing packed decimal data 259
 - QFileSvr.400 file system, using 261
 - QLANSrv 261
 - QOpenSys 261
 - QSYS.LIB 263
 - root 261
- File Transfer Protocol (FTP)
 - anonymous 568
 - application protocol 13
 - batch job 269
 - CL commands 510
 - definition 13
 - exit points 553
 - exit program 535, 548
 - scenario 554
 - logon ID 239
 - overview 235
 - problem analysis 450
 - relationship between client and server 235
 - reporting problems
 - materials required 452
 - request validation exit program 548
 - security for IBM-written programs 513
 - starting
 - to remote system 76
- File Transfer Protocol (FTP) client 261
- AIX (UNIX) example
 - logon process 254
- AS/400 file compatibility 235
- batch job 235
- considerations
 - AS/400 file pre-creation 264
 - batch examples 269
 - file naming 266
 - file pre-creation 264
 - file structure 262, 267
 - job wait time 266
 - level identifier 264
 - national language support 265
 - path 267
 - record structure 262
 - sequencing 264
 - source files 264
 - text files 262
 - time stamp 264
- data transfer methods 258
- file compatibility, AS/400 235
- file naming considerations 235
- file structure considerations 235
- file transfer
 - QSYS.LIB 263
- functions not supported 235
- functions supported 235
- HFS files, transferring 260

- File Transfer Protocol (FTP) client 254 (*continued*)
 - job names 254
 - NAMEFMT, naming format 241
 - naming format indicator 241
 - OS/2 example
 - PS/2 running 256
 - Put process 258
 - server messages 257
 - path considerations 235
 - QDLS documents, transferring 261
 - QFileSvr.400 file system
 - transferring files 261
 - QLANSrv file
 - CCSID code page tagging 264
 - transfer 261
 - QOpenSys file
 - CCSID code page tagging 264
 - transfer 261
 - QSYS.LIB
 - file transfer 263
 - root file
 - CCSID code page tagging 264
 - transfer 261
 - running FTP in batch 235
 - save file transfer 259
 - starting
 - alternative commands 237
 - tracing 456
 - transferring files
 - QFileSvr.400 file system, using 261
 - QLANSrv 261
 - QOpenSys 261
 - root 261
 - transferring HFS files 260
 - transferring QDLS documents 261
 - VAX/Wollongong example
 - logon process 251, 252
 - naming conventions 254
- File Transfer Protocol (FTP) server
 - AIX (UNIX) 282
 - AS/400 file compatibility 280
 - batch job 279
 - ending 281
 - file compatibility, AS/400 280
 - file naming considerations 280
 - file structure considerations 280
 - FTP considerations 282
 - functions not supported 280
 - functions supported 279
 - getting copy of job log 456
 - inactivity time-out value 268
 - job names 280
 - logon exit program (C language) 555
 - NAMEFMT value 283
 - path considerations 280
 - restarting 282
 - running FTP in batch 279
 - sample program, C language
 - server logon exit 555
 - tracing 453
 - VAX/Wollongong 282
 - firewalls 233
 - first failure data capture (FFDC) 480
 - format
 - *AIX 73
 - *AS400 73
 - *NIC 73
 - format communications trace data display 499
 - formatting
 - communications trace 499
 - FORWARD.COND parameter 397
 - forward datagrams 145
 - FORWARD parameter 397
 - frame relay 22
 - FTP (File Transfer Protocol)
 - anonymous 568
 - application protocol 13
 - batch job 269
 - CL commands 510
 - definition 13
 - exit points 553
 - exit program 535, 548
 - scenario 554
 - logon ID 239
 - overview 235
 - problem analysis 450
 - relationship between client and server 235
 - reporting problems
 - materials required 452
 - request validation exit program 548
 - security for IBM-written programs 513
 - starting
 - to remote system 76
 - FTP (File Transfer Protocol) client 259
 - AIX (UNIX) example
 - logon process 254
 - AS/400 file pre-creation 264
 - batch job 235
 - considerations
 - AS/400 file pre-creation 264
 - file naming 266
 - file pre-creation 264
 - file structure 262, 267
 - job wait time 266
 - level identifier 264
 - national language support 265
 - path 267
 - record structure 262
 - sequencing 264
 - source files 264
 - text files 262
 - time stamp 264
 - data transfer methods 258
 - file pre-creation, AS/400 264
 - functions not supported 235
 - functions supported 235
 - HFS files, transferring 260
 - NAMEFMT, naming format 241
 - naming format indicator 241
 - OS/2 example
 - PS/2 running 256
 - Put process 258

- FTP (File Transfer Protocol) client 256 *(continued)*
 - server messages 254
 - QDLS documents, transferring 261
 - QLANSrv file
 - CCSID code page tagging 264
 - QOpenSys file
 - CCSID code page tagging 264
 - root file
 - CCSID code page tagging 264
 - running FTP in batch 235
 - starting
 - alternative commands 237
 - tracing 456
 - transferring HFS files 260
 - transferring QDLS documents 261
 - VAX/Wollongong example
 - logon process 251, 252
 - naming conventions 254
- FTP (File Transfer Protocol) server
 - batch job 279
 - DEBUG subcommand 453
 - FTP considerations 282
 - functions not supported 280
 - functions supported 279
 - getting copy of job log 456
 - inactivity time-out value 268
 - logon exit program (C language) 555
 - NAMEFMT value 283
 - QUOTE DEBUG subcommand 453
 - running FTP in batch 279
 - sample program, C language
 - server logon exit 555
 - tracing 453
- functions not supported by FTP
 - client 235
 - server 280
- functions supported by FTP
 - client 235
 - server 279

G

- gateway
 - processing dead 78
 - proxy gateway 144
- getting started
 - working with communications trace 493
- Government Systems, Inc. 602
- Grant Object Authority (GRTOBJAUT) command
 - 3270 full-screen mode 189
 - ASCII line mode 213
 - VTxxx full-screen mode 199
- group attribute
 - changing 228
- group job
 - definition 228
 - ending 229
 - transferring to 228
- GRTOBJAUT (Grant Object Authority) command
 - 3270 full-screen mode 189
 - ASCII line mode 213

- GRTOBJAUT (Grant Object Authority) command *(continued)*
 - VTxxx full-screen mode 189

H

- hardware resources
 - working with 128
- HCF (Host Command Facility) feature
 - definition 194
- header compression
 - SLIP dial-out profile 149
- help panel URL 327
- help panel URL (HLPPNLURL) parameter 327
- hexadecimal data 172
- HFS files, transferring 260
- HLPPNLURL (help panel URL) parameter 327
- home system
 - definition 220
- hops 33
- host
 - definition 18
 - multihomed 79
- Host Command Facility (HCF) feature
 - definition 194
- host file
 - sending to remote system 76
- host group
 - datagrams 19
- host information 73
- host name
 - configuring 42
 - definition 9, 42
- host print transform 350
- host system
 - definition 19
- host table
 - adding entry 39
 - before using STRTCP command 45
 - configuring 38
 - conversion 45
 - creating 76
 - displaying 39
 - duplicate host names 74
 - listing 39
 - managing 73
 - merge file 76
 - merging 75
 - more than four host names 74
 - performance when merging 427
 - sending host file to remote system 76
 - sending information 73
 - sharing with multiple systems 75
 - socket applications 88
 - starting FTP to remote system 76
 - successful merge 75
 - unsuccessful merge 75
 - using *AIX files 73
 - using *NIC files 74
- HTML
 - converting 5250 data stream to 319

HTML (*continued*)
 converting DDS applications 319
 transformation rules for WSG server 335
HTTP (Hypertext Transfer Protocol) server
 CL commands 512
Hypertext Transfer Protocol (HTTP) server
 CL commands 512

I

IBM Network Station
 BOOTP 379
 TFTP 385
 TFTP server 389
ICMP (Internet Control Message Protocol) redirect
 message 76
identifying
 table object 181
 Work with Object (WRKOBJ) command 181
inactivity time-out value
 FTP server 268
inactivity timeout (INACTTIMO) parameter 325
INACTTIMO (inactivity timeout) parameter 325
incoming 3270 translation table (TBL3270IN)
 parameter 160
incoming ASCII translation table (TBLVTIN)
 parameter 166
incoming data
 5250-to-3270 mapping 521
 ASCII-to-EBCDIC mapping 519
indirect routing 12
information 601
input-inhibited light 195
Integrated Netfinity Server manuals 600
Intelligent Printer Data Stream 350
interface
 binding with routes 58
 definition 30
 End TCP/IP Interfaces (ENDTCPIFC) command 58
 exit point 535
 Start TCP/IP Interfaces (STRTCPIFC) command 57
intermediate system
 definition 220
Internet
 accessing 3
internet
 addressing 2
Internet
 addressing 5
internet
 addressing 18, 19
Internet
 definition 1
internet
 definition 2
Internet
 routing 12
Internet address
 displaying 39
 listing 39
Internet connection server 335

Internet Control Message Protocol (ICMP) 19
Internet Control Message Protocol (ICMP) redirect
 message 76
Internet Group Management Protocol (IGMP) 19
Internet Protocol (IP) 18
 definition 12
 over IPX 20
 over SNA 20
Internet Service Provider (ISP) 4
interoperability
 definition 2
INZWAIT (timeout wait for host) parameter
 3270 full-screen mode 160
 5250 full-screen mode 159
 VTxxx full-screen mode 166
IP (Internet Protocol)
 definition 12
 over IPX 20
 over SNA 20
IP address mapping 232
IP address mapping scenarios 232
IP connections
 Client Access for Windows 95/NT, configuring 294
IP datagram forwarding 36, 146
 SLIP 145
IP multicasting
 host group
 multicast capable routers 91
 subnet 91
IP Routing and Internet Control Message Protocol
 (ICMP) redirecting 76
IP security 4
IPDS
 see Intelligent Printer Data Stream 350
IProtocol (Internet Protocol) 18
IPX connections
 Client Access for Windows 95/NT, configuring 294
ISP (Internet Service Provider) 4

J

job log
 associated with host table merge 75
 working with 438
job logs
 creating REXEC server spooled logs 403
job names
 PPP 136
job status
 SLIP jobs 136
jobs
 displaying SLIP 136
 TCP/IP 45, 425
journal entry types 471

K

KBDTYPE (keyboard language type) parameter 159,
160
keyboard language type (KBDTYPE) parameter 159,
160
keyboard mapping
 ASCII line mode 215

- keyboard mapping (*continued*)
 - changing VTxxx 215
 - from AIX TN3270, 5250 533
 - from DOS TN3270 (PC/TCP) 532
 - from OS/2 TN3270 (PMANT), 5250 532
 - from VAX/MVS, 5250 533
 - TELNET 3270 program 531
 - Telnet server
 - 3270-to-5250 192
 - changing 193, 206
 - displaying 192, 205
 - VTxxx 201
- Korea
 - CCSID when MIME page not supported for WSG server 328, 337

L

- Latin-1 CCSID 266, 280
- library
 - put a file 245
- library lists
 - user profile 367
- limit security officer (QLMTSECOFR) parameter
 - 3270 full-screen mode 189
 - ASCII line mode 213
 - VTxxx full-screen mode 199
- limited broadcast address 8
- line description
 - configuring 506
 - creating 506
 - definition 30
 - maximum transmission unit 507
 - SLIP 144
 - SLIP dial-out profile 151
 - types supported 505
- line mode
 - ASCII 159
- line pools 95
- line printer daemon (LPD)
 - application protocol 13
 - changing attributes 363
 - CL commands 511
 - commands supported for control file printing 369
 - commands supported on AS/400 368
 - configuring 363
 - control file options 369
 - definition 13
 - ending 367
 - overview 363
 - printing ASCII files converted to EBCDIC 373
 - problem analysis 487
 - reporting problems
 - materials required 489
 - RS/6000 example 374
 - selecting an output queue 370
 - starting 366
 - System/370 example 374
- line printer requester (LPR)
 - application protocol 13
 - command considerations 486
- line printer requester (LPR) (*continued*)
 - configuration requirements 13
 - definition 13
 - overview 345
 - problem analysis 486
 - reporting problems
 - materials required 487
 - security for IBM-written programs 514
 - sending a spooled file 346
 - support of PostScript printers 351
- local control functions 173
- local domain and host name
 - configuring 42
 - definition 42
- local domain name 42
- local host name 42
- local interface address
 - SLIP dial-in profile 142
 - SLIP dial-out profile 149
- local network 1
- LOCALHOST host name 40
- log key 332
- logon, FTP
 - exit point interface 551
 - server logon exit
 - sample program, C language 555
- logon ID 239
- long line conversion 299
- LPD (line printer daemon)
 - changing attributes 363
 - CL commands 511
 - commands supported for control file printing 369
 - commands supported on AS/400 368
 - configuring 363
 - configuring device and virtual printer for AIX 356
 - configuring for a RISC System/6000 system 356
 - control file options 369
 - ending 367
 - overview 363
 - printing ASCII files converted to EBCDIC 373
 - problem analysis 487
 - reporting problems
 - materials required 489
 - RS/6000 example 374
 - selecting an output queue 370
 - starting 366
 - System/370 example 374
 - verifying on RISC System/6000 358
- LPR (line printer requester)
 - command considerations 486
 - configuration requirements 346
 - overview 345
 - problem analysis 486
 - reporting problems
 - materials required 487
 - security for IBM-written programs 514
 - sending a spooled file 346
 - support of PostScript printers 351
- LPR command 345

M

mail

- adding users to system distribution directory 289
- delivery problems 476
- MIME 285
- POP mail server 285
- removing users 295
- sending from MIME to OfficeVision 298
- sending OfficeVision mail to POP clients 305
- setting up MIME headers 307
- tracing distributions 469

mail delivery

- problems with 476

mail server framework

- POP mail server 297

managing

- host tables 73
- multiple systems 75

managing SLIP jobs 136

manuals

- communications 599
- Integrated Netfinity Server 600
- internet connection server 600
- programming 600
- security 601
- SNA 601
- system 601
- Systems Network Architecture 601

MAPI service providers

- address book, POP mail server 311
- FAX, support for 311
- supported address types 311

mapping

- IP addresses 232

mapping table 267

- 3270 full-screen mode 523
- ASCII 518, 520
- ASCII character set 524
- ASCII line drawing character set 530
- ASCII mapping tables in FTP command 267
- ASCII-to-EBCDIC 529
- changing 523
- creating
 - ASCII 518
 - Create Table (CRTTBL) command 520
 - EBCDIC 518
 - incoming data (5250-to-3270 mapping) 521
 - new table 523
 - outgoing data (3270-to-5250 mapping) 521
 - using 3270 521
- definition 517
- EBCDIC 518, 520
- EBCDIC character set 524
- EBCDIC-to-ASCII 527
- example 337
- message CPX8416 517
- object type 517
- QTCPEBL object 529
- reading 523
- reading table 523
- sample mappings 523

mapping table 207 (continued)

- Set VT Mapping Tables (SETVTTBL) 523
 - setting 207
 - SETVTTBL (Set VT Mapping Tables) 207
 - summary of specifications 518
 - system-supplied table object 527, 529
 - user-defined 514, 521
- ### mapping tables 328
- maximum sign-on attempts allowed (QMAXSIGN)
 - 3270 full-screen mode 189
 - ASCII full-screen mode 212
 - VTxxx full-screen mode 198
 - maximum transmission unit (MTU)
 - configuring 33
 - definition 33
 - parameter description 507
 - SLIP dial-in profile 143
 - SLIP dial-out profile 150

menu

- System Request 220
- TCP/IP Administration (TCPADM) 237

menu boxes

- WSG server 340

merging host tables 75

- performance 427

message

- CPF87D7
 - 3270 full-screen mode 188
 - 5250 full-screen mode 322
 - ASCII line mode 212
 - virtual device failure 201
 - VTxxx full-screen mode 198
- CPF8940
 - 3270 full-screen mode 188
 - 5250 full-screen mode 322
 - ASCII line mode 212
 - VTxxx full-screen mode 198
- CPX8416 517
- TCP2504
 - 3270 full-screen mode 188
 - 5250 full-screen mode 322
 - ASCII line mode 212
 - VTxxx full-screen mode 198
- TCP3C14 266, 280
- TN3270
 - 3270 full-screen mode 162

message queue

- changing 195

metric

- RouteD server 395

migrating BOOTP 419

Military Standards 602

MIME

- content types 302, 305
- definition 285
- header 328
- setting up 307
- mail
 - converting 317
 - sending across SNADS 308
 - sending to OfficeVision 298

- modem
 - configuring for SLIP 130
 - information
 - SLIP 145, 151
 - SLIP, supported 128
- MPTN (Multiprotocol Transport Networking) architecture
 - application protocol 16
- MTU (maximum transmission unit)
 - configuring 33
 - definition 33
 - parameter description 507
 - SLIP dial-in profile 143
 - SLIP dial-out profile 150
- multicast application programming information 91
- multicast datagrams 91
- multicast restrictions 92
- multihoming 79
- multinational character set
 - definition 179
- multiple hosts 79
- multiple logical interfaces 79
- multiple routes
 - description 84
- multiple systems
 - configuring 75
- Multiprotocol Transport Networking (MPTN) architecture
 - application protocol 16
- Multipurpose Internet Mail Extensions (MIME) 285

N

- name server
 - SLIP dial-out profile 150
- NAMEFMT, naming format 241
- NAMEFMT value, FTP server 283
- naming conventions
 - AIX 255
 - relationship between names and addresses 9
 - VAX 254
- naming format indicator 241
- national character set
 - definition 179
- national language support (NLS)
 - coded character set identifier (CCSID) 517
 - FTP client
 - coded character set identifier (CCSID) 265
 - systems with different primary languages 265
 - using FTP in ASCII mode between two EBCDIC systems 265
 - using FTP in binary mode 265
 - multinational character set 179
 - national character set 179
 - URL format for 334
- NBRCLT (number of clients per server) parameter 324
- NETSTAT (Network Status) command 55, 436
- network
 - class A 4
 - class B 4
 - class C 4
 - class D 4
 - class E 4
 - classes 4

- network (*continued*)
 - local 4
 - protocols 12
 - remote
 - definition 1
 - routers 1
 - software 12
 - network configuration
 - object security 513
 - Network File System (NFS) 15
 - network interface
 - types supported 505
 - Network Status (NETSTAT) command 55, 436
 - network status for TCP/IP
 - connections
 - display 62, 70
 - display totals 71
 - end 62
 - general information 60
 - ending interfaces 58
 - interface
 - display associated routes 66
 - work with configuration status 63
 - interface status
 - display 65
 - work with 56
 - route information, display 59, 68
 - starting interfaces 57
 - work with 55
 - network virtual terminal (NVT)
 - definition 210
 - next hop 33
 - definition 33
 - NFS (Network File System) 15
 - NLS (national language support)
 - coded character set identifier (CCSID) 517
 - considerations
 - SLIP 125
 - FTP client
 - coded character set identifier (CCSID) 265
 - systems with different primary languages 265
 - using FTP in ASCII mode between two EBCDIC systems 265
 - using FTP in binary mode 265
 - multinational character set 179
 - national character set 179
 - URL format for 334
 - NOFORWARD parameter 397
 - Notices 595
 - NULLS (change how nulls are handled) parameter 160
 - number of clients per server (NBRCLT) parameter 324
 - numeric lock keyboard (NUMLCK) parameter 160
 - NUMLCK (numeric lock keyboard) parameter 160
 - NVT (network virtual terminal)
 - definition 210

O

- OACK options 388
- object authority
 - 3270 full-screen mode 190

- object authority (*continued*)
 - ASCII line mode 190
 - VTxxx full-screen mode 199
- object security
 - network configuration 513
- object type for mapping tables 517
- OfficeVision 299
 - exchanging mail with POP 297
 - mail, sending to POP clients 305
- online help 338
- operating modes
 - Telnet client 159
 - Telnet server 183
- Operations Navigator 96
 - accessing BOOTP functions 377
 - accessing DHCP functions 414
 - accessing REXEC functions 399
 - accessing Routed functions 391
 - accessing SOCKS functions 423
 - accessing TFTP functions 383
 - accessing WSG functions 319
 - adding Network Stations 380
 - migrating BOOTP 419
- option acknowledgment
 - see OACK options 388
- options
 - DHCP
 - architected 413
 - specifying 412
 - user-defined 413
- OS/2 example
 - FTP client
 - example 256
 - Put process 258
 - server 257
- OS/400 Network File System (NFS) 15
- outgoing 3270 translation table (TBL3270OUT)
 - parameter 160
- outgoing ASCII translation table (TBLVTOUT)
 - parameter 166
- outgoing data
 - 3270-to-5250 mapping 521
 - EBCDIC-to-ASCII mapping 519
- output queue
 - creating 352

P

- packet
 - definition 2
- packet forwarding 36
- Packet Internet Groper (PING) command 47
 - security for IBM-written programs 514
- packet routing 36
- page down [roll up] key (PAGEDOWN) parameter 160
- page up [roll down] key (PAGEUP) parameter 160
- PAGEDOWN (page down [roll up] key) parameter 160
- PAGESCROLL (VT100 page scroll feature)
 - parameter 166
- PAGEUP (page up [roll down] key) parameter 160

- parameter
 - access log (ACCLOG) parameter 330
 - access logging (ACCLOG) 327
 - ACCLOG (access log) parameter 330
 - ACCLOG (access logging) 327
 - ANSWERBACK (VT100 answerback feature) 166
 - ASCII operating mode (ASCOPRMOD) 166
 - ASCOPRMOD (ASCII operating mode) 166
 - automatically create virtual (devices) (QAUTOVRT)
 - 3270 full-screen mode 188
 - ASCII line mode 212
 - VTxxx full-screen mode 197
 - AUTOSTART 320, 378, 383, 392, 400, 415
 - BOTBNRURL (bottom banner URL) 327
 - bottom banner URL (BOTBNRURL) 327
 - CCSID (coded character set identifier) 166, 328
 - change how nulls are handled (NULLS) 160
 - coded character set identifier (CCSID) 166, 328
 - control characters (CTLCHAR) 166
 - CSRSLT (cursor select key) 160
 - CTLCHAR (control characters) 166
 - cursor select key (CSRSLT) 160
 - display character attributes (DSPCHRATTR) 166
 - display sign-on panel (DSPSGN) 325
 - DSPCHRATTR (display character attributes) 166
 - DSPSGN (display sign-on panel) 325
 - DTARQSTIMO (data request timeout) 325
 - help panel URL (HLPPNLURL) 327
 - HLPPNLURL (help panel URL) 327
 - inactivity timeout (INACTTIMO) parameter 325
 - INACTTIMO (inactivity timeout) 325
 - incoming 3270 translation table (TBL3270IN) 160
 - incoming ASCII translation table (TBLVTIN) 166
 - INZWAIT (timeout wait for host)
 - 3270 full-screen mode 160
 - 5250 full-screen mode 159
 - VTxxx full-screen mode 166
 - KBDDTYPE (keyboard language type) 159, 160
 - keyboard language type (KBDDTYPE) 159, 160
 - limit security officer (QLMTSECOFR)
 - 3270 full-screen mode 189
 - ASCII line mode 213
 - VTxxx full-screen mode 199
 - maximum sign-on attempts allowed (QMAXSIGN)
 - 3270 full-screen mode 189
 - ASCII full-screen mode 212
 - VTxxx full-screen mode 198
 - maximum transmission unit (MTU) 507
 - MTU (maximum transmission unit) 507
 - NBRCLT (number of clients per server) 324
 - NULLS (change how nulls are handled) 160
 - number of clients per server (NBRCLT)
 - parameter 324
 - numeric lock keyboard (NUMLCK) 160
 - NUMLCK (numeric lock keyboard) 160
 - outgoing 3270 translation table (TBL3270OUT) 160
 - outgoing ASCII translation table (TBLVTOUT) 166
 - page down [roll up] key (PAGEDOWN) 160
 - page up [roll down] key (PAGEUP) 160
 - PAGEDOWN (page down [roll up] key) 160
 - PAGESCROLL (VT100 page scroll feature) 166

parameter (*continued*)

- PAGEUP (page up (roll down) key) 330
- PORT (port number of the remote host server application) 160, 166
- port number of the remote host server application (PORT) 160, 166
- QAUTOVRT (automatically create virtual (devices))
 - 3270 full-screen mode 188
 - ASCII line mode 212
 - VTxxx full-screen mode 197
- QLMTSECOFR (limit security officer)
 - 3270 full-screen mode 189
 - ASCII line mode 213
 - VTxxx full-screen mode 199
- QMAXSIGN (maximum sign-on attempts allowed)
 - 3270 full-screen mode 189
 - ASCII full-screen mode 212
 - VTxxx full-screen mode 198
- TABSTOP (VT100 tab stops) 166
- TBL3270IN (incoming 3270 translation table) 160
- TBL328-OUT (outgoing 3270 translation table) 160
- TBLVTDRWI (VT100 special table in) 166
- TBLVTDRWO (VT100 special table out) 166
- TBLVTIN (incoming ASCII translation table) 166
- TBLVTOUT (outgoing ASCII translation table) 166
- TBLWSGIN 328
- TBLWSGOUT 328
- TCPONLY 427
- TELNET timemark timeout (TIMMRKTIMO)
 - 3270 full-screen mode 190, 199, 214
 - Telnet printer pass-through mode 218
- timeout wait for host (INZWAIT)
 - 3270 full-screen mode 160
 - 5250 full-screen mode 159
 - VTxxx full-screen mode 166
- TIMMRKTIMO (TELNET timemark timeout)
 - 3270 full-screen mode 190, 199, 214
 - Telnet printer pass-through mode 218
- TOPBNRURL (top banner URL) 327
- VT100 answerback feature (ANSWERBACK) 166
- VT100 options selected (VTOPT) 166
- VT100 page scroll feature (PAGESCROLL) 166
- VT100 special table in (TBLVTDRWI) 166
- VT100 special table out (TBLVTDRWO) 166
- VT100 tab stops (TABSTOP) 166
- VTOPT (VT100 options selected) 166

Pascal API

- application protocol 16

PASSIVE supply value 395

path considerations 267

performance

- *BASE pool size 425
- file pre-creation 264
- merging host tables 427
- TCP/IP jobs 425
- Telnet server 233

permanent virtual circuit (PVC)

- example 91
- obtaining a network address 22

physical line 505

PING (Packet Internet Groper) command 47

PING (Packet Internet Groper) command 437 (*continued*)

- considerations 437
- security for IBM-written programs 514

planning

- TCP/IP installation and configuration 22
- work with communications trace 493

Point-to-Point Protocol (PPP)

- description 17

Point-to-Point TCP/IP 17, 126

pool size 425

POP (Post Office Protocol) mail server 285, 287

- *ANY support 297
- adding users to system distribution directory 289
- address book 311
- cache 312
- ASCII to EBCDIC conversion 313
- CL commands 511
- Client Access-based clients, setting up 293
- Client Access-based mail users, configuring for 294
- configuring 293
- content types supported 303
- ending 296
- exchanging mail with OfficeVision 297
- mail delivery problems 476
- mail server framework 297
- national language support 313
- overview 285
- problem analysis 476
- removing users 295
- SNADS tunneling 308
- standard POP clients, setting up 292
- starting 296
- verbs supported 296

POP mail server

- definition 14

port

- definition 85
- restriction 85
- TFTP client 385
- TFTP server 385
- well-known 17

PORT (port number of the remote host server application) parameter 160, 166

port number of the remote host server application (PORT) parameter 160, 166

Post Office Protocol (POP) mail server 285, 287

- *ANY support 297
- adding users to system distribution directory 289
- address book 311
- cache 312
- ASCII to EBCDIC conversion 313
- CL commands 511
- Client Access-based clients, setting up 293
- Client Access-based mail users, configuring for 294
- configuring 293
- content types supported 303
- ending 296
- exchanging mail with OfficeVision 297
- mail delivery problems 476
- mail server framework 297

- Post Office Protocol (POP) mail server 297, 287
 - (continued)
 - national language support 297
 - overview 285
 - problem analysis 476
 - removing users 295
 - SNADS tunneling 308
 - standard POP clients, setting up 292
 - starting 296
 - verbs supported 296
- PostScript printer 351
- PPP
 - checking existing connection profiles 96
 - configuring office scenarios 101
 - configuring remote scenarios 103
 - job names 136
 - office example scenarios 100
 - scenario definitions 100
- PPP (Point-to-Point Protocol)
 - description 17
- PPP job
 - status indicators, explanation 136
 - working with active 137
- PPP server
 - definition 15
- PPP/SLIP over *PPP 156
- preferred binding interface 34
- Print Communications Trace (PRTCMNTRC)
 - command 493
- Print key
 - start the printing function 182
- print services facility/6000 function 360
 - configuring 360
- printer
 - changing file default values 368
 - LPR support of PostScript 351
 - supported by LPR 352
- printer pass-through
 - overview 353
 - setup 354
 - starting 355
- printing 231
 - communications trace 493
 - problems 487
 - spooled file 363
 - Telnet system functions 182
- problem analysis
 - APPC over TCP/IP 492
 - domain name system (DNS) server 482
 - FTP 450
 - line printer daemon (LPD) 487
 - line printer requester (LPR) 486
 - POP mail server 476
 - protocol layer problems 492
 - remote execution server (REXEC) 489
 - SLIP 439
 - SLIC trace 134
 - SMTP 457
 - TCP/IP 429
 - TELNET server 443
 - tools 482
- problem analysis (continued)
 - using error log 492
 - using NETSTAT command 436
 - verifying local TCP/IP operation 431
 - verifying remote TCP/IP operation 433
 - workstation gateway server (WSG) 480
- product activity log 503
- profile
 - user 87
- program security 513, 514
- programming
 - examples
 - batch FTP example 269
 - changing keyboard mapping 193, 206
 - reply code monitoring in batch job 269
 - manuals 600
- protocol
 - application
 - Multiprotocol Transport Networking
 - architecture 16
 - Pascal API 16
 - sockets interface 15
 - definition 12
 - multicast routers
 - Internet Group Management Protocol (IGMP) 19
 - network
 - Address Resolution Protocol 19
 - definition 12
 - File Transfer Protocol (FTP) definition 13
 - Internet Control Message Protocol (ICMP) 19
 - Internet Protocol (IP) 18
 - line printer daemon (LPD) 13
 - line printer requester (LPR) 13
 - Simple Mail Transfer Protocol (SMTP)
 - definition 13
 - Simple Network Management Protocol (SNMP) 14
 - TELNET 14
 - Transmission Control Protocol (TCP) 16
 - User Datagram Protocol (UDP) 17
- protocol layer problems, tracing 492
- proxy ARP
 - definition 143
 - restrictions 144
 - SLIP 143
- PRTCMNTRC (Print Communications Trace)
 - command 493
- PSF/6000 function
 - see print services facility/6000 function 360
- public authority 514
- PVC (permanent virtual circuit)
 - example 91
 - obtaining a network address 22

Q

- QAPP0100 537, 571
- QAPP0100 exit point 329
- QAPPCTCP server job 45
- QATMTLOG 327
- QATMTLOG file 330

- QATMTLOG member
 - displaying 330
- QATOCPPSCR 121
- QAUTOVRT (automatically create virtual (devices))
 - parameter
 - 3270 full-screen mode 188
 - ASCII line mode 212
 - VTxxx full-screen mode 197
- QAUTOVRT system value 322
- QDCRDEVD API
 - API, QDCRDEVD 232
- QDLS documents, transferring 261
- QFileSvr.400 file system
 - transferring files 261
- QIBM_QTG_DEVINIT 233
- QLANSrv file
 - CCSID code page tagging 264
 - transfer 261
- QLANSrv file system 235
- QLMTSECOFR (limit security officer) parameter
 - 3270 full-screen mode 189
 - ASCII line mode 213
 - VTxxx full-screen mode 199
- QMAXSIGN (maximum sign-on attempts allowed)
 - 3270 full-screen mode 189
 - ASCII full-screen mode 212
 - VTxxx full-screen mode 198
- QOpenSys file
 - CCSID code page tagging 264
 - transfer 261
- QOpenSys file system 235
- QPGMR authority 509
- QSECOFR authority 509
- QSFWERRLOG system value 452
- QSNMPSA server job 45
- QSRV authority 509
- QSRVBAS authority 509
- QSYS.LIB
 - file transfer 263
- QSYSOPR authority 509
- QSYSWRK subsystem 45
- QTCASC object 527
- QTCPEBL table object 529
- QTCPIP server job 45
- QTFTPxxxxx server job 45
- QTGTELNETS server job 45
- QTLPDxxxxx server job 45
- QTMLPLD default user profile 376
- QTMSNMP server job 45
- QTMSNMPCV server job 45
- QTMTICINT program object 514
- QTPPANSnnn jobs 136
- QTPPDIALnn jobs 136
- QTSMTPBRCV job 45
- QTSMTPBRSR job 45
- QTSMTPLCNT job 45
- QTSMTPSRVR job 45
- querying for distributions for QSECOFR
 - example 464
- QUIT (End FTP Session) subcommand 240

R

- RCF (Receive control file) subcommand 365
- RCFF (Receive control files first) subcommand 365
- RDF (Receive data files) subcommand 365
- RDFUL (Receive data filed with unspecified length)
 - subcommand 365
- read request options 387
- reading
 - mapping table 523
- Receive control file (RCF) subcommand 365
- Receive control files first (RCFF) subcommand 365
- Receive data filed with unspecified length (RDFUL)
 - subcommand 365
- Receive data files (RDF) subcommand 365
- receiving
 - File Transfer Protocol (FTP)
 - text files 262
 - FTP (File Transfer Protocol)
 - text files 262
- record
 - definition 235
- registration facility
 - adding exit program to 537
 - exit point 535
- registration information 325, 334
- related tables 88
- relay agent 420
- Remote Execution (REXEC) server
 - accessing through Operations Navigator 399
 - changing attributes 400
 - CL commands 512
 - client considerations 402
 - command considerations 400
 - connection usage 401
 - creating server spooled job logs 403
 - ending 400
 - exit points
 - choosing command processor 403
 - server security 403
 - exit programs 551
 - getting a copy of job log 491
 - job log
 - getting a copy 491
 - job names 402
 - overview 399
 - problem analysis 489
 - reporting problems
 - materials required 491
 - security for IBM-written programs 514
 - server jobs 402
 - spooled output considerations 402
 - starting 399
 - starting automatically 400
 - tracing 491
- remote interface address
 - SLIP dial-in profile 143
 - SLIP dial-out profile 149
- remote network
 - definition 1
- remote system
 - definition 22

- remote system *(continued)*
 - determining address on X.25 network 22
- remote system access information 152
 - SLIP dial-out profile 152
- remote writer
 - starting 355
- Remove Exit Program (RMVEXITPGM) command 540
- removing
 - exit program 540
- Request for Comments (RFC) 95
 - 1725 285
 - list 602
- request header compression
 - SLIP dial-out profile 149
- request validation
 - application exit point interface 547
 - exit program, FTP 548
- requirements
 - adapters supported 128
 - SLIP 126
 - SLIP protocol 128
- resolving names 10
- Revoke Object Authority (RVKOBJAUT) command 514
- REXEC
 - definition 13
- REXEC (Remote Execution) server
 - accessing through Operations Navigator 399
 - changing attributes 400
 - CL commands 512
 - client considerations 402
 - command considerations 400
 - connection usage 401
 - creating server spooled job logs 403
 - ending 400
 - exit points
 - choosing command processor 403
 - server security 403
 - getting a copy of job log 491
 - job log
 - getting a copy 491
 - job names 402
 - overview 399
- REXEC (remote execution) server
 - problem analysis 489
- REXEC (Remote Execution) server
 - reporting problems
 - materials required 491
- REXEC (remote execution) server
 - security for IBM-written programs 514
- REXEC (Remote Execution) server
 - server jobs 402
 - spooled output considerations 402
 - starting 399
 - starting automatically 400
 - tracing 491
- REXEC (Remote Execution) server)
 - exit programs 551
- RFC (Request for Comments)
 - list 602
- RIP_INTERFACE statement
 - BLOCK 396
- RIP_INTERFACE statement *(continued)*
 - community 396
 - DIST_ROUTES_IN 395
 - FORWARD 397
 - FORWARD.COND 397
 - metric 395
 - NOFORWARD 397
 - overview 394
 - supply values
 - PASSIVE 395
 - SUPPLY OFF 395
 - SUPPLY RIP1 395
 - SUPPLY RIP2 395
- RISC System/6000
 - configuring device and virtual printer for AIX 356
 - configuring print services facility/6000 function 360
 - print services facility/6000 function 360
 - verifying LPD 358
- RISC System/6000 system
 - configuring LPD 356
- RMVEXITPGM (Remove Exit Program) command 540
- root file
 - CCSID code page tagging 264
 - transfer 261
- root file system 235
- route
 - default 35
 - TCP/IP route information, display 59, 68
- Route Daemon
 - see RouteD server 391
- route destination
 - definition 33
- route-to-interface binding, ending interfaces 58
- RouteD server
 - accessing through Operations Navigator 391
 - automatic starting 392
 - BLOCK 396
 - changing attributes 397
 - CL commands 512
 - community 396
 - configuration scenario 393
 - configuring 392, 393
 - definition 13
 - DIST_ROUTES_IN 395
 - ending 392
 - FORWARD 397
 - FORWARD.COND 397
 - metric 395
 - NOFORWARD 397
 - overview 391
 - RIP_INTERFACE statement 394
 - starting 391
- router
 - definition 1
 - linking networks 1
 - performing protocol conversion 1
- routing
 - definition 12
 - direct 12
 - indirect 12
 - office-to-office 102

routing (*continued*)
 remote-to-office 12
running
 FTP in batch 235, 279
RVKOBJAUT (Revoke Object Authority) command 514

S

sample program
 FTP server logon exit 555
 workstation gateway server logon exit program 572
 WSG server logon exit program 572
save files
 transferring 259
saving
 communications trace 499
sbid BDATA packet field 389
sbroadcast option 387, 388
scenario
 configuring LPD for a RISC System/6000
 system 356
 RouteD configuration 393
screen size
 3270 full-screen mode 161
 5250 full-screen mode 159
 VTxxx full-screen mode 166
script
 connection 145
 for SLIP 152
 connection, for SLIP 156
script source information
 SLIP dial-out profile 152
SCS
 see Systems Network Architecture character
 string 350
secured sockets 335
security 4, 601
 authority for users
 QPGMR 509
 QSECOFR 509
 QSRV 509
 QSRVBAS 509
 QSYSOPR 509
 considerations 509
 customer-written programs 514
 File Transfer Protocol 513
 IBM-written programs
 File Transfer Protocol (FTP) 513
 line printer requester (LPR) 513
 PING 513
 Simple Mail Transfer Protocol (SMTP) 513
 TELNET 513
 line printer requester (LPR) 514
 mapping tables
 create with public authority 514
 network configuration 513
 PING 514
 remote execution (REXEC) server 514
 Revoke Object Authority (RVKOBJAUT)
 command 514
 REXEC (remote execution) server 514
 SLIP 156

security 514, 601 (*continued*)
 SMTP 509
 TELNET 514
 user exit programs 513
 workstation gateway server 334
 WSG server 334
Send MIME Mail API 16
Send TCP/IP Spooled File (SNDTCPSPLF)
 command 345
sending
 host file to remote system 76
 host information
 using *AIX format 73
 using *AS400 format 73
 using *NIC format 73
server
 BOOTP 377
 DHCP 405
 line printer daemon 363
 LPD 363
 mapping tables 328
 example 337
 POP mail 285
 Remote Execution (REXEC) 399
 REXEC (Remote Execution) 399
 RouteD 391
 SOCKS 423
 TFTP (Trivial File Transfer Protocol) 383
 WSG 319
server job
 FTP
 ending 281
 restarting 282
 QAPPCTCP 45
 QSNMPSA 45
 QTCPIP 45
 QTFTPxxxxx 45
 QTGTELNETS 45
 QTLDPxxxxx 45
 QTMSNMP 45
 QTMSNMPCRV 45
 QTSMTPBRCV 45
 QTSMTPBRSR 45
 QTSMTPLCNT 45
 QTSMTPSRVR 45
 TCP/IP 45
server port
 TFTP 385
server session
 ending 220
Set Keyboard Map (SETKBDMAP) command 193
Set VT Keyboard Map (SETVTMAP) command 206
Set VT Mapping Tables (SETVTMAP) command 207
SETKBDMAP (Set Keyboard Map) command 193
setsockopt() function 91
setting
 mapping table 207
setting up
 timemark time-out parameter
 3270 full-screen mode 190, 199, 214
SETVTMAP (Set VT Keyboard Map) command 206

- SETVTBL (Set VT Mapping Tables) command 207
- SEU (source entry utility) 518, 519
- shielded twisted pair distributed data interface (SDDI) 22
- signature
 - workstation gateway server session 333
 - WSG server session 333
- Simple Mail Transfer Protocol (SMTP)
 - application protocol 13
 - CL commands 511
 - cleaning up unprocessed distributions 475
 - definition 13
 - distributions
 - journal entry types 471
 - performance 425
 - problem analysis 457
 - when using OfficeVision 462
 - problems analysis
 - without using OfficeVision 464
 - reporting problems
 - materials required 475
 - security for IBM-written programs 514
 - tracing distributions 469
- Simple Network Management Protocol (SNMP)
 - application protocol 14
- SLIP 126
 - asynchronous line for 153, 155
 - authorization list 146
 - configuration profile 132, 133
 - configuring 126, 127
 - configuring dial-in profile 140
 - connection script 145, 152, 156
 - how to create 125
 - connection scripts
 - dial out, how to create 122
 - how to create 122
 - controller and device
 - automatically creating 144, 151
 - datagram forwarding 145
 - default connection scripts
 - location 121
 - default route 144, 150
 - description 17
 - device description
 - remote location name 145, 151
 - dial-in profile
 - configuring 140
 - local interface address 142
 - maximum transmission unit 143
 - remote interface address 143
 - dial-out configuration profile 147
 - dial-out connection script example 123
 - dial-out profile
 - additional name server 150
 - line description 151
 - local interface address 149
 - maximum transmission unit 150
 - name server 150
 - remote interface address 149
 - remote system access information 152
 - request header compression 149
- SLIP 152 (*continued*)
 - dial-out profile (*continued*)
 - script source information 150
 - hardware requirements 128
 - IP datagram forwarding 145
 - job
 - status indicators, explanation 136
 - working with 135, 136
 - working with active 137
 - line description 144
 - modem configuration 130
 - modem information 145, 151
 - monitoring activity 134
 - NLS considerations 125
 - planning for configuration 127
 - problem analysis 439
 - proxy ARP 143
 - remote system access information 152
 - requirements for 126
 - SLIC trace 134
 - spooled file output example 140
 - system access authorization list 146
 - using a direct connection 155
 - using a modem 153
- SLIP problems
 - reporting 442
- SMTP (Simple Mail Transfer Protocol)
 - application protocol 13
 - CL commands 511
 - cleaning up unprocessed distributions 475
 - definition 13
 - distributions
 - journal entry types 471
 - performance 425
 - problem analysis 457
 - when using OfficeVision 462
 - problems analysis
 - without using OfficeVision 464
 - reporting problems
 - materials required 475
 - security for IBM-written programs 514
 - tracing distributions 469
- SNA (Systems Network Architecture)
 - connections
 - Client Access for Windows 95/NT, configuring 294
 - manuals 601
 - over IP 20
 - over IPX 20
 - servers
 - number of, configuring 295
- SNADS tunneling 308
 - configuring clients 308
 - how it works 308
- SDTCPSPFL (Send TCP/IP Spooled File)
 - command 345
- SNMP (Simple Network Management Protocol)
 - application protocol 14
- sockets
 - accessing using IPX 20
 - accessing using SNA 20

- sockets (*continued*)
 - SOCK_DGRAM 20
- sockets interface
 - application protocol 15
- SOCKS server
 - accessing through Operations Navigator 423
 - overview 423
- software, network 12
- source entry utility (SEU) 518, 519
- source service access point (SSAP)
 - AA entry 506
 - Ethernet line 506
- special authority, *IOSYSCFG 30
- specifications 602
- spooled file 182
 - AS/400 receiving from AS/400 364
 - AS/400 receiving from non-AS/400 365
 - attributes 367
 - authority for putting on output queues 372
 - authority required 375
 - locating 346
 - naming 366
 - ownership 370
 - printing 363
 - sending 345, 346
 - sending from AS/400 to AS/400 348
 - sending from AS/400 to non-AS/400 349
 - sending large 353
 - starting the transfer 347
 - tips 353
 - transforming 350
- spooled files
 - working with 139
- SSAP (source service access point)
 - AA entry 506
 - Ethernet line 506
- SST (system service tools)
 - accessing 494
- start
 - communications trace display 496
- Start Communications Trace (STRCMNTRC)
 - command 493
- Start PDM (STRPDM) command 330
- Start Remote Writer (STRRTWTR) command 355
- Start Source Entry Utility (STRSEU) command 518
- Start TCP/IP (STRTCP) command 337
- Start TCP/IP Interfaces (STRTCPIFC) command
 - TCP/IP interfaces 57
- Start TCP/IP Server (STRTCPSVR) command 320, 366, 378, 383, 391, 399, 416
- Start TCP/IP Telnet (STRTCPTELN) command 220
- starting
 - BOOTP server 378
 - communications trace 493, 495
 - DHCP server 415
 - LPD (line printer daemon) 366
 - service tool display for SST 494
 - Source Entry Utility 518
 - TCP/IP 337
 - TCP/IP and TCP/IP servers 43
 - TFTP (Trivial File Transfer Protocol) server 383
- starting (*continued*)
 - Trivial File Transfer Protocol (TFTP) server 378
 - work with communications trace 493
- stopping
 - communications trace 498
- STRCMNTRC (Start Communications Trace)
 - command 493
- STRPDM (Start PDM) command 330
- STRRTWTR (Start Remote Writer) command 355
- STRSEU (Start Source Entry Utility) command 518
- STRTCP (Start TCP/IP) command 337
- STRTCPIFC (Start TCP/IP Interfaces) command 57
- STRTCPSVR (Start TCP/IP Server) command 320, 366, 378, 383, 391, 399, 416
- STRTCPTELN (Start TCP/IP Telnet) command 220
- subcommand
 - RCF 365
 - RCFF 365
 - RDF 365
 - RDFUL 365
- subnet
 - addressing
 - definition 33
 - definition 5
 - extension to Internet addressing 6
 - mask 6
 - definition 33
 - routing
 - definition 33
- subnet broadcast option 385
- subnetting
 - definition 33
- subnetwork
 - definition 6
- subsystem routing and device name selection
 - routing and device name selection
 - device name selection 233
- SUPPLY OFF supply value 395
- SUPPLY RIP1 supply value 395
- SUPPLY RIP2 supply value 395
- SVC (switched virtual circuit)
 - obtaining a network address 22
- switched virtual circuit (SVC)
 - obtaining a network address 22
- system
 - POP mail users, removing 295
- system access authorization list
 - SLIP 146
- system activity
 - monitoring for SLIP 134
- System API Enhancement
 - API Enhancement, System 231
- system distribution directory
 - adding POP mail users 289
 - address book, refreshing 294
 - description 289
 - SNADS tunneling, configuring 308
- System Licensed Internal Code (SLIC) trace 134
- system manuals 601
- system name
 - character restrictions 10

- system name *(continued)*
 - displaying 10
 - listing 39
 - resolving names 10
- system service tools (SST)
 - accessing 494
 - display 494
- system value
 - QAUTOVRT 322
 - QSFWERRLOG 452
- Systems Network Architecture
 - character string 350
- Systems Network Architecture (SNA)
 - over IP 20
- Systems Network Architecture (SNA) manuals 601

T

- table
 - BOOTP 379
- table object 181
- tables
 - required for socket applications 88
 - working with 530
- TABSTOP (VT100 tab stops) parameter 166
- Taiwan
 - CCSID when MIME page not supported for WSG server 328, 337
- TBL3270IN (incoming 3270 translation table)
 - parameter 160
- TBL3270OUT (outgoing 3270 translation table)
 - parameter 160
- TBLVTDRAWI (VT100 special table in) parameter 166
- TBLVTDRAWO (VT100 special table out)
 - parameter 166
- TBLVTIN (incoming ASCII translation table)
 - parameter 166
- TBLVTOU (outgoing ASCII translation table)
 - parameter 166
- TBLWSGIN parameter 328
- TBLWSGOUT parameter 328
- TCP (Transmission Control Protocol) 16
 - definition 12
- TCP/IP (Transmission Control Protocol/Internet Protocol)
 - attributes 36
 - ending
 - End TCP/IP (ENDTCP) command 46
 - installation and configuration
 - planning 22
 - interface
 - entry 30
 - introduction 1
 - job
 - description 425
 - QAPPCTCP 45
 - QSNMPSA 45
 - QTCPIP 45
 - QFTFPxxxxx 45
 - QTGTELNETS 45
 - QTLPDxxxxx 45
 - QTMSNMP 45
 - QTMSNMPRCV 45

- TCP/IP (Transmission Control Protocol/Internet Protocol) *(continued)*
 - job *(continued)*
 - QTSMTPBRCCL 425
 - QTSMTPBRSR 45
 - QTSMTPLCNT 45
 - QTSMTPSRVR 45
 - planning for installation and configuration 22
 - problem analysis 429
 - QSYSWRK subsystem 45
 - starting 43, 337
- TCP/IP Administration (TCPADM) menu 22, 237
- TCP/IP File Server Support/400 15
- TCP/IP servers
 - starting 43
- TCP2504 message
 - 3270 full-screen mode 188
 - 5250 full-screen mode 322
 - ASCII line mode 212
 - VTxxx full-screen mode 198
- TCP3C14 message 266, 280
- TCPADM (TCP/IP Administration) menu 237
- TCPONLY parameter 427
- TELNET
 - application protocol 14
 - CL commands 510
- Telnet
 - client 3270 full-screen mode 159
 - ending
 - server session 220
 - operation
 - differences between VTxxx and 5250 terminals 163
- TELNET
 - security for IBM-written programs 514
- Telnet
 - server 3270 full-screen mode 186
 - server 5250 full-screen mode 183
 - server ASCII line mode 210
 - server printer pass-through mode 216
 - server session
 - ending 220
 - server VTxxx full-screen mode 196
 - starting 160
 - system functions
 - printing 182
- TELNET
 - using 3270 full-screen mode 523
 - using 3270 mapping tables 521
- Telnet
 - VTxxx full-screen mode 163
 - workstation type negotiations 229
- Telnet client
 - 3270 full-screen mode considerations
 - 3270 keyboards 160
 - 5250 keyboards 160
 - character sets 160
 - cursor select key 162
 - display station 160
 - handling null characters 162
 - keyboard language types 160

- Telnet client (*continued*)
 - messages 160
 - personal computer keyboards 161
- TELNET client
 - device termination exit program 546
 - exit program
 - device termination 546
- Telnet client
 - introduction 159
- TELNET client
 - keyboard mapping 531
- Telnet client
 - operating modes 159
 - VTxxx full-screen mode 163
- Telnet Printer Pass-Through mode 217
- Telnet printer pass-through mode
 - security considerations 218
- Telnet server
 - 3270 full-screen mode considerations 186
 - 3270-to-5250 keyboard mapping 192
 - changing keyboard map 193
 - defining device capabilities 195
 - displaying keyboard map 192, 205
 - input-inhibited light 195
 - messages 195
 - 5250 full-screen mode considerations 183
 - configuring virtual controllers and displays 183
 - ASCII line mode considerations 210
 - ending session 220
 - introduction 183
 - Printer pass-through mode considerations 216
 - response time 233
 - setting up
 - 3270 full-screen mode 183
 - 5250 full-screen mode 183
 - ASCII line mode 183
 - VTxxx full-screen mode 183
 - Telnet printer mode considerations 216
 - VTxxx
 - changing keyboard map 206
- TELNET timemark timeout (TIMMRKTIMO) parameter
 - 3270 full-screen mode 190, 199, 214
 - Telnet printer pass-through mode 218
- text file 262
- TFRGRPJOB (Transfer to Group Job) command 228
- TFTP (Trivial File Transfer Protocol) server
 - accessing through Operations Navigator 383
 - automatically starting 383
 - blksize option 387, 388
 - broadcast data (BDATA) packets 389
 - changing attributes 384
 - CL commands 512
 - configuring non-IBM Network Station clients 389
 - ending 384
 - extensions 385
 - option acknowledgment (OACK) 388
 - overview 383
 - read request (RRQ) options 387
 - sbroadcast option 387, 388
 - server and client ports 385
 - starting 383
- TFTP (Trivial File Transfer Protocol) server (*continued*)
 - subnet broadcast option 383
 - transfer size option 385
 - tsize option 387, 389
- TFTP server
 - definition 13
- timeout wait for host (INZWAIT) parameter
 - 3270 full-screen mode 160
 - 5250 full-screen mode 159
 - VTxxx full-screen mode 166
- TIMMRKTIMO (TELNET timemark timeout) parameter
 - 3270 full-screen mode 190, 199, 214
- Tips
 - enabling on a browser 329
- top banner URL 327
- top banner URL (TOPBNRURL) parameter 327
- TOPBNRURL (top banner URL) parameter 327
- trace
 - communications
 - formatting 499
 - saving 499
 - starting communications 495
 - stopping communications 498
 - verifying contents of communications 501
 - working with communications 493
- Trace TCP/IP Application (TRCTCPAPP)
 - command 455
- tracing
 - FTP server 453
 - protocol layer problems 492
- transfer size option 385
- Transfer to Group Job (TFRGRPJOB) command 228
- transferring
 - data
 - methods 258
 - files containing packed decimal data 259
 - HFS files 260
 - QDLS documents 261
 - QSYS.LIB files 263
 - save files 259
 - to group job 228
- translation tables 267
- Transmission Control Protocol (TCP) 16
 - definition 16
- Transmission Control Protocol/Internet Protocol (TCP/IP)
 - attributes 36
 - command authority for users
 - QPGMR 509
 - QSECOFR 509
 - QSRV 509
 - QSRVBAS 509
 - QSYSOPR 509
 - interface
 - entry 30
 - introduction 1
 - job
 - description 425
 - QAPPCTCP 45
 - QSNMPA 45
 - QTCPIP 45

- Transmission Control Protocol/Internet Protocol (TCP/IP) *(continued)*
 - QFTFPxxxx 36
 - QTGTELNETS 45
 - QTLPDxxxx 45
 - QTMSNMP 45
 - QTMSNMPRCV 45
 - QTSMPBRCL 45
 - QTSMPBRSR 45
 - QTSMPCLNT 45
 - QTSMPSRVR 45
 - QSYSWRK subsystem 45
 - routing 12
 - software 12
 - starting 43
- TRCTCPAPP (Trace TCP/IP Application)
 - command 455
- Trivial File Transfer Protocol (TFTP) server
 - accessing through Operations Navigator 383
 - automatically starting 383
 - blksize option 387, 388
 - broadcast data (BDATA) packets 389
 - changing attributes 384
 - CL commands 512
 - configuring non-IBM Network Station clients 389
 - ending 384
 - extensions 385
 - option acknowledgment (OACK) 388
 - overview 383
 - read request (RRQ) options 387
 - sbroadcast option 387, 388
 - server and client ports 385
 - starting 383
 - subnet broadcast option 385
 - transfer size option 385
 - tsize option 387, 389
- tsize option 387, 389
- tunneling
 - configuring system distribution directory entries 308
 - SNADS 308

U

- UDP (User Datagram Protocol)
 - definition 17
- URL (uniform resource locator)
 - format
 - national language support (NLS) 334
 - format supported by WSG server 334
 - WSG information 319
- USA Standard 7-bit ASCII Character Set
 - table 526
- user authority
 - QPGMR 509
 - QSECOFR 509
 - QSRV 509
 - QSRVBAS 509
 - QSYSOPR 509
- User Datagram Protocol (UDP)
 - definition 17

- user-defined data stream (UDDS)
 - definition 211
- user-defined mapping table 521
- user profile
 - creating a default 376
 - definition 87
 - library lists 367

V

- validation list 101
- VAX/Wollongong example
 - FTP client
 - example 251
 - logon process 252
 - naming conventions 254
 - verbs, POP mail server 296
- Verify TCP/IP Connection (VFYTCPCNN)
 - command 47
 - verifying
 - contents of communications trace 501
 - TCP/IP connections 47
- VFYTCPCNN (Verify TCP/IP Connection)
 - command 47
- virtual device 322
 - definition 183, 211
 - description 183, 211
- virtual workstation controller description 215
- VT100 answerback feature (ANSWERBACK)
 - parameter 166
- VT100 full-screen mode considerations
 - automatic wrap 207
 - control character keywords 174
 - keyboard indicator 166
 - keyboard mapping 201
 - server overview 196
 - setting keyboard map 206
- VT100 options selected (VTOPT) parameter 166
- VT100 page scroll feature (PAGE_SCROLL)
 - parameter 166
- VT100 special table in (TBLVTDRWI) parameter 166
- VT100 special table out (TBLVTDRWO)
 - parameter 166
- VT100 tab stops (TABSTOP) parameter 166
- VT220 full-screen mode considerations
 - control character keywords 174
 - national language support 179
- VTOPT (VT100 options selected) parameter 166
- VTxxx full-screen mode
 - screen size 166
- VTxxx full-screen mode considerations
 - 5250 cursor movement keys 164
 - character attributes 166
 - character data 172
 - control key keywords 172
 - display screen 208
 - error condition 207
 - hexadecimal data 172
 - local control functions 173
 - mapping table 207
 - operational differences 163

- VTxxx full-screen mode considerations (*continued*)
 - Set VT Mapping Tables (SETVTTBL) command 164
 - setting up 183
 - SETVTTBL (Set VT Mapping Tables) command 207
 - Start TCP/IP Telnet (STRTCPTLN) command
 - applicable parameters 166
 - supported attributes 208
 - system request 207

W

- Web browser
 - enabling 329
 - MIME header 328
 - starting a session to an AS/400 system 332
- WebConnection 319
- well-known port
 - definition 17
- Windows 95
 - address book, POP mail server 311
- wireless LAN 22
- work management 233
- Work Station Function Keys (WSFKEYS)
 - command 161
- Work with Active Jobs (WRKACTJOB) command 138, 363, 434
- Work with BOOTP Table (WRKBPTBL) command 379
- Work with Configuration Status (WRKCFGSTS)
 - command 229
- Work with Hardware Resources (WRKHDWRSC)
 - command 128
- Work with Registration Information (WRKREGINF)
 - command 325, 334, 537
- Work with Routed Configuration (WRKRTDCFG)
 - command 392
- Work with Server Table (WRKSVRTBLE)
 - command 333
- Work with SLIP jobs 135, 137
- Work with Spooled Files (WRKSPLF) command 139
- Work with Tables (WRKTBL) command 530
- work with TCP/IP connection status 60
- Work with TCP/IP Network Status (WRKTCPSTS)
 - command 55
- Work with TCP/IP Point-to-Point (WRKTCPPPT)
 - command 135, 137
- Work with TCP/IP Status (WRKTCPSTS)
 - command 436
- working with
 - active PPP jobs 138
 - communications trace 493
 - configuration status 229
 - hardware resources 128
 - job log 438
 - message queue 438
 - SLIP job 135
 - active 137
 - spooled files 139
 - tables 530
- workstation gateway server (WSG)
 - accessing through Operations Navigator 319
 - AS/400 menus 343

- workstation gateway server (WSG) (*continued*)
 - bottom banner URL 319
 - buttons 339
 - changing 337
 - CL commands 512
 - clients per server 325
 - coded character set identifier (CCSID) 328
 - configuration examples 337
 - configuring 321, 323
 - customizing online help information 330
 - definition 14
 - ending 320
 - example exit program for sign-on display 571
 - exit point 569
 - exit point interface (QAPP0100) 569, 570
 - exit program 329
 - exit program for sign-on display 571
 - FAQs 342
 - help panel URL 327
 - HTML transformation rules 335
 - logging access 327, 330
 - managing virtual devices 322
 - menu boxes 340
 - online help 329, 338
 - overview 319
 - problem analysis 480
 - QAPP0100 parameter descriptoins 570
 - QAPP0100 required parameters 569
 - sample display 336
 - sample logon exit program 572
 - secured sockets 335
 - security 334
 - server mapping tables 328
 - session identifier 333
 - session signature 333
 - starting 320
 - starting a client session 332
 - starting automatically 320
 - supported URL formats 334
 - top banner URL (TOPBNRURL) 327
 - URL 319
- workstation type negotiations 229
- wrapping, automatic 207
- WRKACTJOB (Work with Active Jobs) command 138, 363, 434
- WRKBPTBL (Work with BOOTP Table) command 379
- WRKCFGSTS (Work with Configuration Status)
 - command 229
- WRKHDWRSC (Work with Hardware Resources)
 - command 128
- WRKREGINF (Work with Registration Information)
 - command 325, 334, 537
- WRKRTDCFG (Work with Routed Configuration)
 - command 392
- WRKSPLF (Work with Spooled Files) command 139
- WRKSVRTBLE (Work with Server Table)
 - command 333
- WRKTBL (Work with Tables) command 530
- WRKTCPPPT (Work with TCP/IP Point-to-Point)
 - command 135, 137

- WRKTCPSTS (Work with TCP/IP Network Status)
 - command 55, 436
- WSFKEYS (Work Station Function Keys)
 - command 161
- WSG (workstation gateway server)
 - accessing through Operations Navigator 319
 - AS/400 menus 343
 - bottom banner URL 327
 - buttons 339
 - changing 337
 - CL commands 512
 - clients per server 325
 - coded character set identifier (CCSID) 328
 - configuration examples 337
 - configuring 321, 323
 - customizing online help information 330
 - ending 320
 - example exit program for sign-on display 571
 - exit point 569
 - exit point interface (QAPP0100) 569, 570
 - exit program 329
 - exit program for sign-on display 571
 - FAQs 342
 - help panel URL 327
 - HTML transformation rules 335
 - logging access 327, 330
 - managing virtual devices 322
 - menu boxes 340
 - online help 329, 338
 - overview 319
 - problem analysis 480
 - QAPP0100 parameter descriptions 570
 - QAPP0100 required parameters 569
 - sample display 336
 - sample logon exit program 572
 - secured sockets 335
 - security 334
 - server mapping tables 328
 - session identifier 333
 - session signature 333
 - starting 320
 - starting a client session 332
 - starting automatically 320
 - supported URL formats 334
 - top banner URL (TOPBNRURL) 327
 - URL 319

X

- X.25
 - configuring X.25 remote system information 36
 - DDN conversion algorithm 37
 - obtaining network addresses 22
 - permanent virtual circuit (PVC) 22
 - switched virtual circuit (SVC) 22

Readers' Comments — We'd Like to Hear from You

AS/400e
TCP/IP Configuration and Reference
Version 4

Publication No. SC41-5420-03

Overall, how satisfied are you with the information in this book?

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Overall satisfaction	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

How satisfied are you that the information in this book is:

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Accurate	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Complete	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Easy to find	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Easy to understand	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Well organized	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Applicable to your tasks	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Please tell us how we can improve this book:

Thank you for your responses. May we contact you? Yes No

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you.

Name

Address

Company or Organization

Phone No.



Cut or Fold
Along Line

Fold and Tape

Please do not staple

Fold and Tape



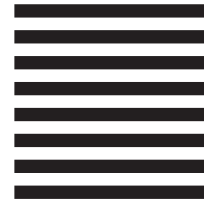
NO POSTAGE
NECESSARY
IF MAILED IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST-CLASS MAIL PERMIT NO. 40 ARMONK, NEW YORK

POSTAGE WILL BE PAID BY ADDRESSEE

IBM CORPORATION
ATTN DEPT 542
3605 HWY 52 N
ROCHESTER MN 55901-7829



Fold and Tape

Please do not staple

Fold and Tape

Cut or Fold
Along Line



Printed in the United States of America
on recycled paper containing 10%
recovered post-consumer fiber.

SC41-5420-03



Spine information:



AS/400e

OS/400 TCP/IP Configuration and Reference V4R4

Version 4