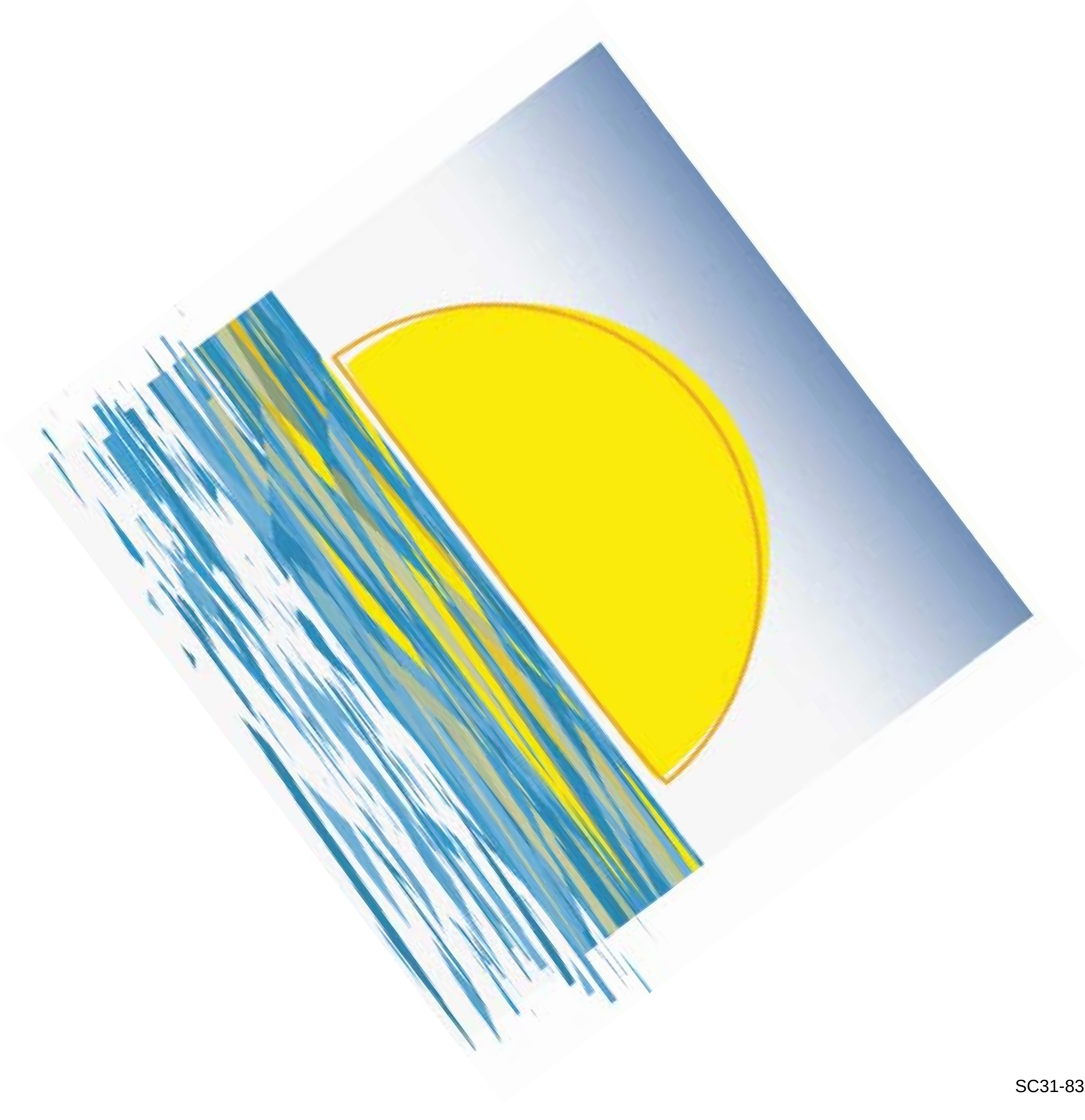
OS/390 TCP/IP OpenEdition

Planning and Release Guide



SC31-8303-00

OS/390 TCP/IP OpenEdition

Planning and Release Guide

SC31-8303-00

Note

Before using this information and the product it supports, be sure to read the general information under Appendix I, “Notices” on page 125.

First Edition (June 1997)

This edition applies to OS/390 (5645-001) and OS/390 TCP/IP OpenEdition. Make sure you are using the correct edition for the level of the product.

Order publications through your IBM representative or the IBM branch office serving your locality. Publications are not stocked at the address below.

IBM welcomes your comments. A form for readers' comments may be at the back of this publication. If the form has been removed, you may send your comments to the following address:

International Business Machines Corporation
Department CGMD
P.O. Box 12195
Research Triangle Park, North Carolina 27709
USA

If you prefer to send comments electronically, use one of the following methods:

Fax (USA and Canada):	1-800-227-5088
Internet e-mail:	usib2hpd@vnet.ibm.com
World Wide Web:	http://www.s390.ibm.com/os390
IBMLink:	CIBMORCF at RALVM13
IBM Mail Exchange:	USIB2HPD at IBMMAIL

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

Contents

Figures

Tables

About This Book

This book describes OS/390 TCP/IP OpenEdition, a feature available with OS/390 Release 3. The basic TCP/IP component supplied with OS/390 Release 3 is a V3R2 stack. In OS/390 Release 3, the classic (non-OpenEdition) applications run on the V3R2 stack; only the OpenEdition (OE) applications run on OS/390 TCP/IP OpenEdition stack.

This book is intended to provide planning information on the functions and the configurations possible with OS/390 TCP/IP OpenEdition.

OS/390 TCP/IP OpenEdition is an integral part of the OS/390 family of products. For an overview and mapping of the documentation available for OS/390, see the *OS/390 Information Roadmap*.

Who Should Use This Book

This book is used by planners, system programmers, and network administrators who are planning to install OS/390 TCP/IP OpenEdition on OS/390 Release 3. You should use this book to :

- Evaluate TCP/IP in OS/390
- Understand the differences between the previous release of TCP/IP and OS/390 TCP/IP OpenEdition
- Understand the differences between non-OpenEditon applications and OpenEdition applications
- Prepare for the coexistence of OS/390 TCP/IP OpenEdition and TCP/IP V3R2 for MVS

You should be familiar with OS/390 TCP/IP and the protocol commands to use this book.

What is Not in This Book

This book **does not** cover planning and migration for TCP/IP V3R2 for MVS. See the *TCP/IP for MVS: Planning and Migration Guide* for for TCP/IP V3R2 planning and migration information.

Where to Find Related Information on the Internet

You may find the following information helpful.

For current updates to the TCP/IP Version 3 Release 2 for MVS documentation described in “Bibliography” on page 131, check out the TCP/IP for MVS home page:

```
http://www.networking.ibm.com/tcm/tcmprod.html
```

To keep in close touch with OS/390, we suggest you look at the OS/390 home page:

```
http://www.s390.ibm.com/os390
```

To keep abreast of new products and technologies from IBM Networking, take a look at the IBM Networking home page:

```
http://www.networking.ibm.com/
```

The IBM Networking Software Glossary is now available in HTML format as well as PDF. You can access it directly at the following URL:

```
http://www.networking.ibm.com/nsg/nsggls.htm
```

How to Contact IBM Service

For telephone assistance in problem diagnosis and resolution (in the United States or Puerto Rico), call the IBM Software Support Center anytime (1-800-237-5511). You will receive a return call within 8 business hours (Monday – Friday, 8:00 a.m. – 5:00 p.m., local customer time).

Outside of the United States or Puerto Rico, contact your local IBM representative or your authorized IBM supplier.

Chapter 1. Evaluating the TCP/IP Enhancements

This chapter describes the OS/390 OpenEdition environment, OS/390 TCP/IP OpenEdition, the basic contents of OS/390 TCP/IP OpenEdition, and the differences between TCP/IP V3R2 for MVS and OS/390 TCP/IP OpenEdition.

The OpenEdition Environment

IBM OS/390 provides OpenEdition services that support an environment within which operating systems, servers, distributed systems, and workstations share common interfaces. OpenEdition supports standard application development across multivendor systems.

OpenEdition is required if you want to create and use applications that conform to the POSIX** or XPG4 standard (a UNIX** specification). OpenEdition combines the personal power of the workstation, the flexibility of open systems, and the strength of OS/390. It supports and fosters a super environment of larger operating systems or servers and of distributed systems and workstations that share common interfaces.

Users can switch between the traditional TSO/E interface and the OpenEdition shell interface. UNIX-skilled users can interact with the system, using a familiar set of standard commands and utilities. OS/390-skilled users can interact with the system, using familiar TSO/E commands and interactive menus to create and manage hierarchical file system files and to copy data between OS/390 data sets and files. Application programmers and users have both sets of interfaces to choose from and can choose to mix these interfaces.

OS/390 provides the OpenEdition shell, which is modeled after the UNIX shell.

The shell is a command processor that you use to:

- Invoke shell commands or utilities that request services from the system
- Write shell commands using the shell programming language
- Run shell scripts and C-language programs interactively (in the foreground), in the background, or in batch.

There is a high degree of interoperability between MVS and the OS/390 OpenEdition shell. Table 1 shows the parallels between the MVS and the OpenEdition shell.

Table 1 (Page 1 of 2). Working Interactively in the MVS and Shell Environments

Activity	MVS Environment	Shell Environment
Programming for everyday tasks	CLIST and REXX (TSO)	Shell scripts and REXX
Editing	ISPF	ed, vi
Job Control	SDSF	ps, jobs, kill
Background job	Submit Job (TSO)	command &
Programming	Compilers, LINK	C89, make
Debugging	TSO Test, Inspect	dbx

Table 1 (Page 2 of 2). Working Interactively in the MVS and Shell Environments

Activity	MVS Environment	Shell Environment
Data Management	DFSMShsm, ISPF	tar, cpio, pax

For more details on how to use the OS/390 OpenEdition shell and utilities on an IBM OS/390 System, see the *OS/390 OpenEdition User's Guide*.

TCP/IP provides a set of communications protocols that support peer-to-peer connectivity functions for both local and wide area networks including the most popular wide area network, the Internet. IBM TCP/IP V3R2 is shipped with OS/390. For our customers who use OpenEdition applications, an additional TCP/IP, OS/390 TCP/IP OpenEdition, is available. OS/390 TCP/IP OpenEdition not only includes the OpenEdition (OE) applications previously available in the IBM TCP/IP Version 3 for OpenEdition MVS Applications Feature, it also includes a redesigned stack and more OE functions. The OS/390 TCP/IP OpenEdition stack was redesigned to provide performance enhancements such as high throughput for file transfers.

OS/390 TCP/IP OpenEdition is described in this book and the other books in the OS/390 TCP/IP OpenEdition library. IBM TCP/IP V3R2 is discussed in the TCP/IP V3R2 library. Other components of OS/390 are described in the *OS/390 OpenEdition: Planning Guide*.

Figure 1 on page 3 shows OS/390 R3 with TCP/IP V3R2 on the left side and OS/390 TCP/IP OpenEdition on the right side of the dashed line. OS/390 provides the OpenEdition MVS Sockets, the OpenEdition MVS Logical File System (LFS), an AF_INET Physical File System (integrated sockets or Common-INET), and the Hierarchical File System (HFS) physical file system (PFS). Although the IBM Internet Connection Server for MVS is not provided with OS/390 TCP/IP OpenEdition, it was included on the figure because it works like an OpenEdition (OE) server.

TCP/IP V3R2 comes with a stack and with the traditional (non-OE) applications.

OS/390 TCP/IP OpenEdition comes with a different stack and with OpenEdition applications.

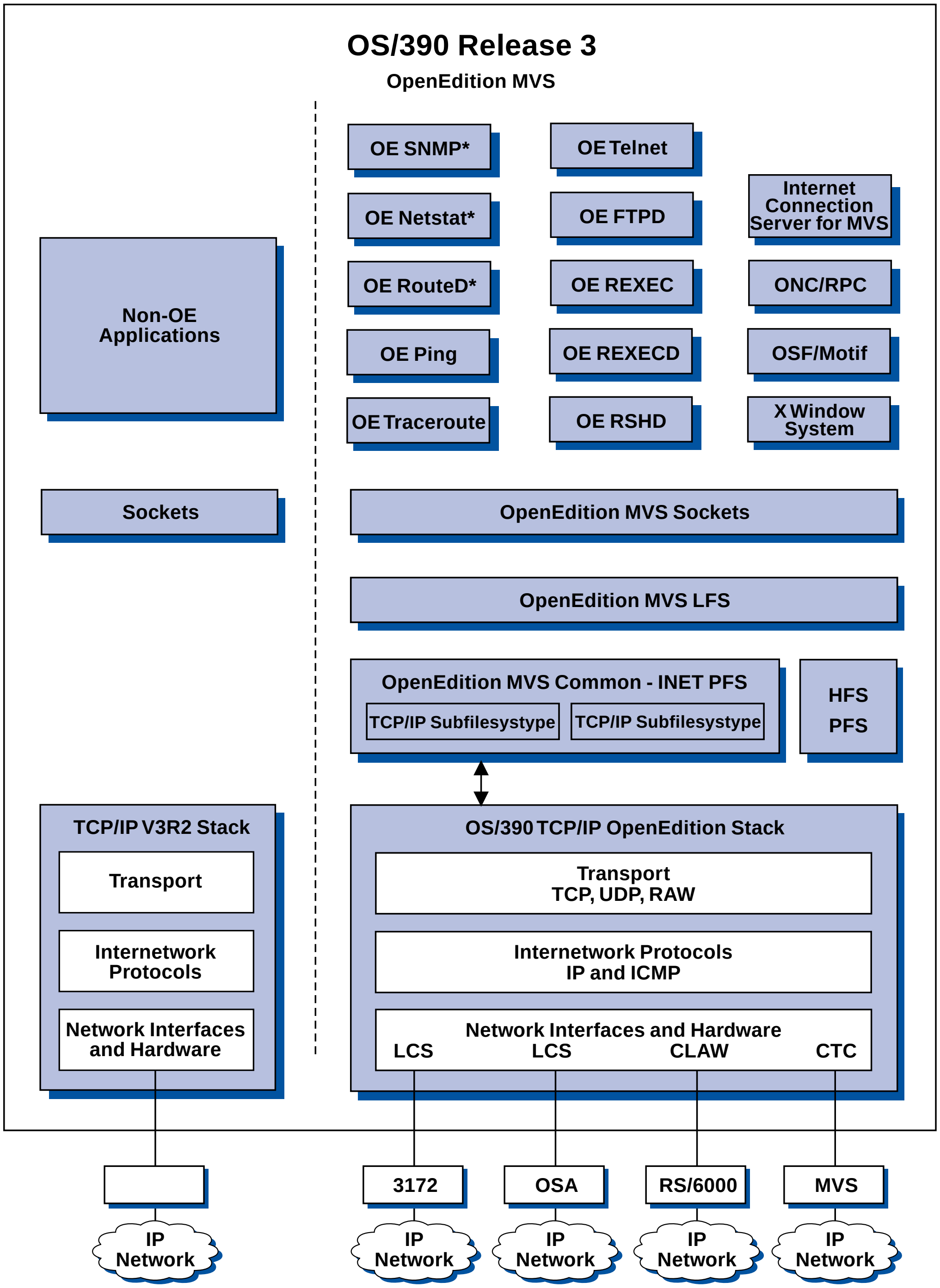


Figure 1. TCP/IP in OS/390

OS/390 TCP/IP OpenEdition

TCP/IP V3R2 and OS/390 TCP/IP OpenEdition can run on OS/390 Release 3 at the same time. You can define a total of 32 stacks and run up to eight stacks at a time on the same OS/390 System. The eight stacks that are running can be any combination of V3R2 and OS/390 TCP/IP OpenEdition stacks. If you are running only OS/390 TCP/IP OpenEdition stacks, you need to have a Domain Name Server (DNS) available to the stacks for better performance. A DNS is supplied with

TCP/IP V3R2, but not with OS/390 TCP/IP OpenEdition. If one V3R2 stack is running, the OS/390 TCP/IP OpenEdition stacks can use its DNS. To maximize the use of your stacks, run only your non-OE applications on TCP/IP V3R2 and run your OE applications on OS/390 TCP/IP OpenEdition.

For a summary of OS/390 TCP/IP OpenEdition and V3R2 functions, see the following table:

Table 2 (Page 1 of 2). OS/390 TCP/IP OpenEdition and V3R2 Functions

OS/390 TCP/IP OpenEdition	**V3R2 Non-OE**
OE SNMP including asynchronous transfer mode (ATM) network management	SNMP
OE PING	PING
OE Netstat	Netstat
OE RouteD including VIPA	RouteD including VIPA
OE Traceroute	Traceroute
Dynamic configuration	
	LPR
	TELNET
	FTP
	RPC
	X Window Systems
VIPA	VIPA
HOMETEST command[1]	HOMETEST command
TESTSITE command[1]	TESTSITE command
MAKESITE command[1]	MAKESITE command
CONVXLAT command[1]	CONVXLAT command
OE TelnetD server	OE TelnetD server[2]
OE FTPD server	OE FTPD server[2]
OE RSHD server	OE RSHD server[2]
OE REXECD server	OE REXECD server[2]
OE REXEC client	OE REXEC client[2]
OE ONC**/RPC programming libraries	OE ONC/RPC programming libraries[2]
OE OSF/Motif** and X Window System** programming libraries	OE OSF/Motif and X Window System programming libraries[2]
	C sockets
OE C sockets	OE C sockets
OE BPX sockets	OE BPX sockets
	Pascal API
	IUCV sockets
	REXX sockets
	ASM macro and COBOL

Table 2 (Page 2 of 2). OS/390 TCP/IP OpenEdition and V3R2 Functions

OS/390 TCP/IP OpenEdition	**V3R2 Non-OE**
	ASM and PL/1 callable sockets
Support for year 2000 dates	Support for year 2000 dates

Notes:

1. This function is available on OS/390 TCP/IP OpenEdition only if TCP/IP V3R2 is installed. The command uses code in the V3R2 base.
2. This function is available on TCP/IP V3R2 only if the IBM TCP/IP Version 3 for OpenEdition MVS Applications Feature is installed.

OS/390 TCP/IP OpenEdition uses OE C sockets, which are also referred to as OpenEdition MVS Sockets, and the BPX assembler Callable Services.

The Enhanced Performance of OS/390 TCP/IP OpenEdition

The improved performance of OS/390 TCP/IP OpenEdition results from the newly-designed stack, the Communications Storage Management (CSM) facility, and the Multi-Path Channel (MPC) I/O process. CSM and MPC are supplied by ACF/VTAM* V4R4 and are used by VTAM and OS/390 TCP/IP OpenEdition. The Communications Storage Management facility reduces data moves and manages subsystem data pools. The Multi-Path Channel handles multiple I/O dispatchable units of work and supports the following:

- Channel-to-channel (CTC) communication (MVS to MVS)
- Common link access to workstation (CLAW)
- 3172 LAN Channel Station (LCS) device drivers

CLAW connects an IBM RS/6000* workstation directly to the host over a channel. Some OEM boxes also use the CLAW interface. The 3172 LCS device driver links to an IBM 3172 Interconnect Controller, which connects to a token ring, an ethernet, or FDDI local area network.

RAS Enhancements

The following Reliability, Availability, and Serviceability (RAS) improvements are part of OS/390 TCP/IP OpenEdition:

- OS/390 TCP/IP OpenEdition uses the MVS CTrace
  - To trace more than 256 bytes per record
  - To provide a minimal trace option that always traces a subset of information that will always be available
  - To provide a large default trace buffer size to allow larger trace records and more trace information to remain in storage
- OS/390 TCP/IP OpenEdition enables you to dynamically change the osnmpd trace selection by using the modify command.
- OS/390 TCP/IP OpenEdition handles abends by both configuration and subagent components independently from other components.
- OS/390 TCP/IP OpenEdition continues to send IP packet traces to the Generalized Trace Facility (GTF).

- The OS/390 TCP/IP OpenEdition stack supports dynamic stack configuration to allow the deletion of definitions while TCP/IP is active. This enhances availability.

OE SNMP

Simple Network Management Protocol (SNMP) is a network management protocol that is used to monitor network elements attached to the TCP/IP internet. Specifically, SNMP defines an architecture that consists of:

1. Network management stations (network managers, also known as SNMP clients)
2. Network elements (hosts and gateways)
3. Network management agents and subagents

The network manager can ask the agent for specific information about network elements or the agent can tell the manager that something happened to one or more network elements.

The protocol used between the manager and an element is SNMP which defines the format of the packets of information exchanged. The transport protocol used for SNMP requests is UDP.

The stations that monitor the network elements are called network managers. (The network managers are also known as SNMP clients because they make requests.) For example, operator can use an osnmp command to communicate with an OS/390 TCP/IP OpenEdition SNMP agent. The purpose of the osnmp command is to query the SNMP agents for network management information.

The information about the devices managed is defined and stored in the Management Information Base (MIB) in the SNMP agent and its subagents. A MIB variable (MIB object) is a specific instance of data defined in a collection of objects relating to a common management area. (The collection is called a MIB module.) The MIB modules are maintained by the SNMP agent and the MIB modules can be changed without recompiling the agent. Other types of information are also stored in the MIB such as routing information and TCP/IP data such as counts of requests received at the IP layer, ICMP, and so forth.

OS/390 TCP/IP OpenEdition supports SNMPv2 with community-based security. The SNMPv2 agent (osnmpd) implements SNMPv1 and SNMPv2 to provide SNMP support for the IP host to which it connects. The SNMPv2 agent is an OE application and uses the HFS.

A subagent extends the set of MIB variables (objects) provided by the osnmpd agent. With the subagent, you define MIB variables in your own environment and register them with the osnmpd agent. The MIB variables are described in the *OS/390 TCP/IP OpenEdition User's Guide*.

In OS/390 TCP/IP OpenEdition, the function that provides support of the MIB objects for TCP/IP has been separated as a separate task, running within the TCP/IP address space. This subagent is configured and controlled by SACONFIG statement in the central TCP/IP PROFILE SACONFIG statement. Systems where SNMP support is not required can disable this function and save system resources. The MVS SNMP subagent uses AF_UNIX to communicate with the agent.

The SNMP agent distributed programming interface (DPI) permits you to dynamically add, delete, or replace management variables in the local Management Information Base (MIB). You can define your own subagent without recompiling the agent. OS/390 TCP/IP OpenEdition provides a complete API for the DPI 2.0 subagent, including a new DPI sample program and new AF_UNIX support. DPI 2.0 makes it easier to write subagents and simplifies your ability to develop and administer your application. The application programmers interface for the DPI subagent is described in the *OS/390 TCP/IP OpenEdition Programmer's Reference*. Whatever applications you had using DPI 1.1 can run unchanged in DPI 2.0; however, more function is now available.

DPI 2.0 does not provide support for DPIconnect_to_agent_IUCV or DPIconnect_to _agent_UDP.

OE PING

The Packet InterNet Groper, commonly known as PING, is the command that sends an Internet Control Message Protocol (ICMP) Echo Request to a gateway, router, or host with the expectation of receiving a reply. A PING is used to find out if a gateway, router, or host can be reached.

PING is an application that uses the Internetwork Protocol layer in the V3R2 stack. The command for the program is PING, which is run from TSO. OE PING (oping) is an application that uses the Internetwork Protocol layer in the OS/390 TCP/IP OpenEdition stack. The command for the program is oping, which is run from OMVS. The syntax for oping is UNIX-like. For example, the user specifies oping -c for "count."

In UNIX** or OpenEdition terms, the OE PING program that sends the request is considered a client. The owner of an IP address (a host, router, or gateway) is considered a server because it has to respond to the OE PING client request and send a message back to the OE PING program.

When a host, router, or gateway sends a message back to the PING program that requested the message, the message contains the identifier and sequence number from the original ping request.

The PING program compares the information sent with the information returned to it. It compares the number and sequence of packets sent with the number and sequence of packets received. If the information does not match, it can be used to diagnose connectivity problems. If the information is not returned at all, the IP address to which the ping was sent is not responding or the time required for the message to travel to and from the IP address exceeded the time allotted for the trip. The time allotted is specified in the time-to-live (TTL) field of the IP header. In addition to diagnostics, PING can be used to determine performance.

OE Traceroute

The OE version of tracerte, otracert, is a completely new ported version based on the UNIX implementation:

- UNIX command syntax
- Trace messages will be sent to stdout
- Error messages will be sent to stderr
- Enhanced debug, enabled with -d parameter
- OE MAN help support

- More options

The otracert command displays the route that a packet takes to reach the requested target. The trace starts at the first router and uses a series of UDP probe packets with increasing IP time-to-live (TTL) values to determine the sequence of routers that must be traversed in order to reach the target host. Traceroute provides a very useful management tool to determine the route that a packet would take to a destination.

The OE version of traceroute has more functions than the TSO version. These function are listed in “Traceroute and OE Traceroute” on page 77.

OE NETSTAT

The NETSTAT command on an MVS system displays the network status of the local host, including information about TCP/IP connections, network clients, gateways, and devices. Netstat can also be used for diagnostic purposes. The purpose of OE Netstat (onetstat) is the same as the purpose of non-OE Netstat. OE Netstat provides the network status of the local host server, and OE Netstat can be used for diagnostic purposes.

Although the same in purpose, OE Netstat differs from the non-OE Netstat in several ways. The OE NETSTAT command (onetstat) is an OE C socket application and it uses the IOCTL interface to implement its functions. The Netstat options in V3R2 were changed or deleted for OS/390 TCP/IP OpenEdition. The changed options include ALL, ALLCON, ARP, BYTEINFO, CLIENTS, CONN, DEVLINKS, GATE, HELP, HOME and SOCKETS. The POOLSIZE option was deleted for OS/390 TCP/IP OpenEdition. The syntax for onetstat is UNIX-like. For example, -c is used instead of CONN. One option, INTERVAL, is used as an output option in OS/390 TCP/IP OpenEdition. Two new OE Netstat options, CONFIG and ROUTE were added.

OE ROUTED

The OE Route Daemon (ORouteD) is a server that implements the Routing Information Protocol (RIP) described in RFC 1058. The OE route daemon provides an active alternative to TCP/IP gateway definitions that would otherwise be static. When the MVS host is configured correctly to use ORouteD, the host becomes an active RIP router in a TCP/IP network.

The OE RouteD provided with OS/390 TCP/IP OpenEdition performs the same functions as the RouteD for TCP/IP V3R2. However, there are some differences between OE RouteD and RouteD. Because OE RouteD is an OE application, OE RouteD requires the Hierarchical File System (HFS) to run.

As with most OE applications, the location and search order for configuration files for OE RouteD are different from the location and search order for configuration files for the non-OE RouteD application. The syntax for OE RouteD is orouted. ORouteD can be started from the MVS or from the OE shell by a superuser (a user who has root authority and can change the directories in an HFS.) Also, ORouteD uses a standard message catalog located in directory in the HFS. The environment variables NLSPATH and LANG set the directory location for the message catalog path. The MODIFY command can be used to dynamically change the characteristics of a task.

OE TelnetD server

The OE Telnet server allows hosts in the IP network to log on to the OpenEdition MVS shell environment directly, without going through TSO. The OE TelnetD server supports both line mode and raw mode telnet connections. It does not support the TN3270 mode.

OE FTPD Server

The OE FTPD server allows remote users to transfer files directly in and out of the Hierarchical File System (HFS) in OpenEdition MVS. In addition to support for the HFS files, the server also supports standard MVS data sets. This FTPD server includes all the functions of the non-OE C-FTP server in TCP/IP V3R2, including SQL query support and JES file type support.

OE RSHD Server

The OE RSHD server provides server functions in the OE environment for remote shell clients based on the remote shell (RSH) protocol. Commands are executed as OpenEdition MVS shell commands.

OE REXECD Server

The OE REXECD server provides server functions in the OpenEdition MVS environment for remote commands based on the remote execution (REXEC) protocol. Commands are executed as OpenEdition MVS shell commands.

OE REXEC Client

The OE REXEC client allows OpenEdition MVS shell users to execute commands on a remote system.

ONC/RPC Programming Libraries

The Open Network Computing (ONC**) / Remote Procedure Call (RPC) programming libraries supply an application programming interface used for remote procedure call server and client applications in OE and includes a library of procedures and external data representation.

OSF/Motif and X Window System Programming Libraries

These programming libraries provide OSF/Motif 1.2.4 and X Window System Version 11 Release 6 programming interfaces to be used by X Window System client programs in the OpenEdition MVS environment.

Differences between V3R2 and OS/390 TCP/IP OpenEdition

In addition to the enhanced performance of OS/390 TCP/IP OpenEdition, there are some differences in function, installation, configuration, use, and diagnosis. For example, OS/390 TCP/IP OpenEdition uses RACF authorization instead of OBEYLIST authorization.

Some statements supported in the OS/390 TCP/IP OpenEdition PROFILE are different from those in V3R2. Refer to Appendix A, “New and Changed Commands and Statements” on page 41 for a list of new, changed, obsolete, or unsupported commands and statements for OS/390 TCP/IP OpenEdition. Unlike V3R2, OS/390

TCP/IP OpenEdition uses the MVS VARY operator command instead of TSO OBEYFILE command to change the TCP/IP configuration after initialization.

Some traces are collected differently for OS/390 TCP/IP OpenEdition and some traces go to syslogd. These differences are described in detail throughout the OS/390 TCP/IP OpenEdition library:

- Refer to the *OS/390 Release 3 Program Directory* for installation.
- Refer to *OS/390 TCP/IP OpenEdition Configuration Guide*, SC31-8304 for configuration.
- Refer to *OS/390 TCP/IP OpenEdition Messages and Codes*, SC31-8307, and *OS/390 TCP/IP OpenEdition Diagnosis Guide*, SC31-8492 for diagnosis.

Commands

The following table shows the types of commands used in the OE and non-OE environments. You can compare the OE commands in OS/390 TCP/IP OpenEdition with the non-OE commands in V3R2.

OpenEdition commands are case-sensitive. The commands and parameters must be entered in the case shown in Table 3. The OS/390 TCP/IP OpenEdition commands are described in detail in the *OS/390 TCP/IP OpenEdition Configuration Guide* and in the *OS/390 TCP/IP OpenEdition User's Guide*. The TCP/IP V3R2 commands are described in detail in the *TCP/IP for MVS: Customization and Administration Guide* and in the *TCP/IP for MVS: User's Guide*.

In TCP/IP V3R2, TSO and MVS operator console commands can be entered in uppercase, lowercase, or mixed-cased. TSO is not case-sensitive.

In OS/390 TCP/IP OpenEdition and the TCP/IP Version 3 for OpenEdition MVS Applications Feature, the OMVS commands are in lowercase. OE is case-sensitive. The meaning of a parameter depends on whether it is an uppercase or lowercase character.

Table 3 (Page 1 of 2). Commands

User/Tasks	TCP/IP V3R2	Where Issued	OS/390 TCP/IP OE	Where Issued
Network Administrator				
Provides status of local host	NETSTAT	TSO	onetstat	OE shell
Determines accessibility of a foreign node (remote node)	PING	TSO	oping	OE shell
Monitors routers and attached networks	SNMP	NetView	osnmp	OE shell
Provides diagnosis information for network problems	TRACERTE	TSO	otracert	OE shell
System Administrator				
Shows information about system resources	Display	MVS operator console	Display	MVS operator console
Changes the status of system resources	Modify	MVS operator console		

Table 3 (Page 2 of 2). Commands

User/Tasks	TCP/IP V3R2	Where Issued	OS/390 TCP/IP OE	Where Issued
Changes the status of system resources	OBEYFILE	TSO	vary... obeyfile	MVS operator console
Provides device, gateway, and host status	NETSTAT	TSO	onetstat	OE shell
Verfies the host name and address configuration	HOMETEST	TSO	see note	TSO
Generates new hlq.HOSTS.SITEINFO and hlq.HOST.ADDRINFO data sets	MAKESITE	TSO	see note	TSO
Verifies that the hlq.HOSTS.SITEINFO and hlq.HOST.ADDRINFO data sets resolve the name of a host, gateway or network	TESTSITE	TSO	see note	TSO
Application Programmer				
Generates C code to implement a remote procedure call (RPC) protocol	RPCGEN	RPCGEN	orpcgen	OE shell
Displays server information	RPCINFO	RPCINFO	orpcinfo	OE shell

Note:

1. You can run HOMETEST, MAKESITE, and TESTSITE for OS/390 TCP/IP OpenEdition, if you have TCP/IP V3R2 installed on the same OS/390 system.

Servers

The following table shows the servers used in the OE and non-OE environments. You can compare the OE servers in OS/390 TCP/IP OpenEdition with the non-OE servers in V3R2 and with the OE servers in the TCP/IP OpenEdition Applications Feature for V3R2.

Table 4 (Page 1 of 2). Servers

Tasks	V3R2	Command or program to start server	TCP/IP V3R2 OE Applica-tions Feature	Command to start server	OS/390 TCP/IP OE	Command to start server
Manages the routing tables for another program	RouteD	ROUTED	n/a	n/a	OE RouteD	n/a
Monitors network resources	SNMP	SNMPD	n/a	n/a	OE SNMP	osnmpd
Transfers data sets between hosts	FTPD	EZAFTSRV	OE FTPD	ftpd	OE FTPD	ftpd
Connects a remote host	TELNET	internal	OE TELNETD	otelnetd	OE TELNETD	otelnetd
Executes command on remote host and receives results on local host	REXECD	RSHD	OE REXECD	orexecd	OE REXECD	orexecd

Table 4 (Page 2 of 2). Servers

Tasks	V3R2	Command or program to start server	TCP/IP V3R2 OE Applications Feature	Command to start server	OS/390 TCP/IP OE	Command to start server
Executes command on remote host and receives results on local host	RSHD	RSHD	OE RSHD	orshd	OE RSHD	orshd
Maps client programs to the port numbers of server programs	PORTMAP	PORTMAP	OE PORTMAP	oportmap	OE PORTMAP	oportmap

OS/390 TCP/IP OpenEdition Parts

The following chart shows the location of the OS/390 TCP/IP OpenEdition parts in OS/390. The locations are different from TCP/IP V3R2.

Table 5. OS/390 TCP/IP OpenEdition Parts

Part	Hierarchical File System Location	MVS Library Location
Syslogd server	/usr/sbin/syslogd[1]	SYS1.LINKLIB(SYSLOGD)
OE TelnetD server	/usr/sbin/otelnetd[1]	SEZALINK(OTELNETD)
OE REXECD server	/usr/sbin/rexecd[1]	SEZALINK(REXECD)
OE RSHD server	/usr/sbin/rshd[1]	SEZALINK(RSHD)
OE FTPD server		
Listener	/usr/sbin/ftpd[1]	SEZALINK(FTPD)
Server	/usr/sbin/ftpdsrv[1]	SEZALINK(FTPDSRV)
OE ONC/RPC		
Portmapper	/bin/oportmap[1]	SEZALINK(OPORTMAP)
RPCGEN	/bin/orpcgen[1]	SEZALINK(ORPCGEN)
RPCINFO	/bin/orpcinfo[1]	SEZALINK(ORPCINFO)
Header files	/usr/include/rpc	n/a
Library archive	/usr/lib/librpclib.a	n/a
Sample code	/usr/lpp/tcpip/rpc/samples	n/a
X Window Systems		n/a
Header files	/usr/include	
Library archives	/usr/lib	
Uil compiler	/bin/X11/uil	
Sample code	/usr/lpp/tcpip/X11R6/Xamples	
OE SNMP server	/usr/sbin/osnmpd[1]	SEZALINK(OSNMPD)
OE SNMP client	/usr/bin/osnmp[1]	SEZALINK(OSNMP)
OE PING client	/usr/bin/oping[1]	SEZALINK(OPING)
OE NETSTAT client	/usr/bin/onetstat[1]	SEZALINK(ONETSTAT)
OE ROUTED server	/usr/sbin/orouted[1]	SEZALINK(OROUTED)
OE TRACERT server	/usr/bin/otracert[1]	SEZALINK(OTRACERT)

Note:

1. These executable files exist in two places in OMVS. One copy is in the Hierarchical File System and the other copy is in an MVS data set. When these files have the sticky bit set in the HFS, their MVS library counterparts are being executed. OS/390 TCP/IP OpenEdition is shipped with the sticky bit set on.

Chapter 2. Reviewing the TCP/IP Requirements

This chapter describes the technical requirements that must be considered before the TCP/IP elements are installed. The installation of OS/390 Release 3 or later is the primary prerequisite for the installation of OS/390 TCP/IP OpenEdition. OS/390 Release 3 supports all of the technical requirements of OS/390 TCP/IP OpenEdition.

This section describes the requirements for the operating system and the hardware and software environments for the OS/390 TCP/IP OpenEdition stack. For the requirements for the TCP/IP V3R2 stack and for the information on how to migrate from TCP/IP V3R1 to TCP/IP V3R2, please refer to the *TCP/IP for MVS: Planning and Migration Guide*.

Operating System Requirements

OS/390 TCP/IP OpenEdition operates under the following operating system:

Table 6. Operating System Requirements

TCP/IP V3R2	**OS/390 TCP/IP OpenEdition**
MVS/SP* (MVS/ESA) Version 4.3, or later. **Note:** The TCP/IP Version 3 for MVS OpenEdition Applications Feature requires MVS/ESA SP Version 5.2.2, or later	MVS for OS/390 R3 or later

Hardware Requirements

OS/390 TCP/IP OpenEdition is designed to operate on any IBM System/390* processor that supports MVS for OS/390 Release 3 or later. The machine requirements are:

- 96 MB of direct access storage to hold the libraries and data for the base OS/390 TCP/IP OpenEdition product and features

  Note:

  In addition, you will need 370 MB of direct access storage to hold the libraries and data for the base TCP/IP V3R2 product and features if your configuration includes TCP/IP V3R2.

- A tape or cartridge drive for installing TCP/IP
- A System/390 channel for attachment of one or more of the following:
  - IBM RS/6000
  - IBM 8232 LAN Channel Station
  - IBM 3172 Interconnect Controller Model 001, 002, or 003
- One or more terminals using IBM 3270 protocols for maintenance functions

Network Attachments:

To attach TCP/IP to the network you need one of the following network processors and associated components:

IBM 3172 Interconnect Controller with the Interconnect Controller Program (ICP)

- IBM 3172 Interconnect Controller Model 001, 002, or 003
- IBM Interconnect Controller Program (5601-433 or 5621-425)
- One of the following types of adapters:
  - 3172 Interconnect Controller Token-Ring Adapter
  - 3172 Interconnect Controller Ethernet Adapter
  - Auto LANStreamer* MC 32 Adapter
  - EtherStreamer* MC 32 Adapter
  - 3172 Interconnect Controller PC Network Adapter II/A (#2271)
  - TURBOWAYS* ATM Adapter for the 3172 Model 3 (LAN emulation for Token-Ring and Ethernet)
- One of the following channel adapters:
  - PCA adapter
  - ESCON* adapter
- Standard System/370* I/O Channel Interface Cable

Note: The Pentium** P90 Processor is supported.

IBM 8232 LAN Channel Station

- IBM 8232 LAN Channel Station Model 001 or 002
- One or more adapters for Token Ring, PC Network, or Ethernet
- Standard System/370 I/O Interface cable

IBM RISC System/6000* Parallel Channel Attachment

- IBM RISC System/6000 with the Block Multiplexer Channel Adapter
- AIX* Version 3 Release 2.3 (5756-030), or later, with feature (#5056)
- Standard ESCON Adapter

IBM S/390 Open Systems Adapter Feature

- S/390 Open Systems Adapter permits direct attachment of S/390 processors/servers to the following networking standards:
  - Ethernet
    - IEEE 802.2 LAN MAC protocols
    - IEEE 802.3 CSMA/CD protocols
  - Token Ring
    - IEEE 802.2 LAN MAC protocols
    - IEEE 802.5 MAC protocols
  - FDDI
    - IEEE 802.2 LAN MAC protocols
    - ANSI X3T9.5

  - S/390 Open Systems Adapter supports the TCP/IP access standard (TCP/IP passthru)

- S/390 Open Systems Adapter 2 feature (OSA2)
  - Continues to provide direct access to Ethernet, Token Ring, and FDDI LANs.
  - TCP/IP passthru
  - Asynchronous Transfer Mode (ATM)
    - ATM LAN emulation support (only)

Software Requirements

Table 7 provides a list of software products required for TCP/IP V3R2 and OS/390 TCP/IP OpenEdition.

Table 7 (Page 1 of 3). Software Requirements

Function	**TCP/IP V3R2**	**OS/390 TCP/IP OpenEdition**
SNALINK LU0 Support SNALINK LU6.2 Support	ACF/VTAM V3.4.2 for MVS/ESA (5685-085), or later ACF/NCP V6 (5688-231), or later (Ethernet only), or ACF/NCP V7R1 (5648-063), or later	SNALINK is not supported
IBM GDDM through X-Windows System	GDDM/MVS V2R3 (5665-356) with PTF UN21085 and PTF UN85120 GDDM/MVS V3R1 (5695-167) with APAR PN84007 GDDM/MVS V3R1.1 (5695-167) with APAR PN84008	GDDM is not supported
File access protection for FTP	RACF V1.9 (5740-XXH), or later, or equivalent	OS/390 R3 RACF (optional feature) or later, or equivalent Available for OE Applications Feature FTP only
FTP DB2* Query	DB2 V2.3 (5665-DB2), or later	DB2 V2.3 (5665-DB2), or later Available for OE Applications Feature FTP only
Offload	OS/2 V2R1 (5871-P01) TCP/IP 2.0 for OS/2 (5875-P01) (base kit only) **Note:** CSD UN64092, or later, is required for packet trace support.	Offload is not supported

Table 7 (Page 2 of 3). Software Requirements

Function	TCP/IP V3R2	OS/390 TCP/IP OpenEdition
MVS TCP/IP functions written in C: Domain Name Server FTP C Server Kerberos Services MISCSERV NCPROUTE Server Network Database System (NDB) client and server Network Computing System** (NCS**) Network Print Facility ISPF interface Portmapper server Remote Procedure Call (RPC) RouteD server RPCinfo SNMP client and server Non-OpenEdition X-Window system C sample programs	IBM C/370* Runtime Library V2R2 (5688-188) or IBM AD/CYCLE LE/370 Runtime Library V1R3, V1R4, or V1R5 (5688-198) or Language Support Feature of OpenEdition/MVS (5655-068, 5655-069)	OS/390 base Language Environment Runtime Library (LE V1R7) or later Runtime Library V1R7 (5688-198) or later) **Note:** ONLY functions included as part of the OS/390 TCP/IP OpenEdition are supported. The following are **NOT** supported by OS/390 TCP/IP OpenEdition: Domain Name Server FTP C Server (non-OE) Kerberos Services Miscserv NCPROUTE Network Database System (NDB) Network Computing System (NCS) Network Print Facility Non-OE X-Windows System
Sockets API	IBM C/370 Library V2.2 (5688-188) or LE/370 V1.3, 1.4, 1.5, or 1.7 (5688-198)	OS/390 R3 Language Environment or later. **Note: Only the OE Sockets API is supported for OS/390 TCP/IP OpenEdition.**
C sample programs	IBM AD/Cycle* C/370 Compiler V1R2 and one of the following IBM libraries: • IBM C/370 Library V2.2 (5688-188) or • LE/370 V1.3, 1.4, 1.5, or 1.7 (5688-198) **Note:** Non-IBM C compilers that support IBM's TCP/IP V3R2 and that execute with the above libraries may be substituted.	IBM AD/Cycle C/370 Compiler V1R2 and LE/370 V1R7. **Note:** Non-IBM C compilers that support IBM's OS/390 TCP/IP OpenEdition and that execute with the above libraries may be substituted. **Note: Only the OE Sockets API is supported for OS/390 TCP/IP OpenEdition.**

Table 7 (Page 3 of 3). Software Requirements

Function	TCP/IP V3R2	OS/390 TCP/IP OpenEdition
OpenEdition Features • FTP C Server • Rcommands • RPC • Portmapper • Telnet Server • X Window System	• MVS/ESA SP Version 5.2.2 or later OpenEdition Shell and Utilities feature • IBM C/C++ for MVS/ESA Language Support feature or Language environment for MVS and VM Version 1.7 (5688-198), or higher. • FTP requires DFSMSdfp Version 1.2 (5695-DF1) with PTF UW02765 or later release. • File access protection for FTP requires RACF (TM) V2R1 (5695-039) or later, or equivalent. • OE FTP DB2 Query requires DB2 Version 2.3 (5665-DB2), or higher. • IBM C/C++ for MVS/ESA V3 R1 (5655-121) or IBM SAA* AD/Cycle C/370 Compiler V1 R2 (5688-216) for customer-written programs in C that interface to an OE feature: – REXEC, REXECD, RSHD – RPC – X Window System	MVS for OS/390 R3 LE/370 V1R7 (5688-198) and OS/390 R3 versions of products listed for options under TCP/IP V3R2. **Note:** These functions were compiled using C/370 V3R2 compiler
Customer-written programs in C that interface to an X-Window System client, Remote Procedure Call, TCP or UDP protocol boundary, DPI or IP	IBM C/370 Runtime Library V2R2 (5688-188) or IBM AD/CYCLE LE/370 Runtime Library V1R7 **Note:** Non-IBM C compilers that support IBM's TCP/IP V3R2 and that execute with the above libraries may be substituted.	OS/390 R3 base Language Environment or later **Note: Only the OE Sockets API is supported for OS/390 TCP/IP OpenEdition.** **Note:** Non-IBM C compilers that support IBM's OS/390 TCP/IP OpenEdition and that run with the preceding libraries may be substituted.
Customer-written programs in C that interface to an OpenEdition feature: • Rcommands • RPC • X Window System	IBM AD/Cycle C/370 Compiler V1R2 LE/370 1.7	LE/370 1.7
ATM Network Management	Not Applicable	OS/390 base Open Systems Adapter (OSA) Support Facility V1R2 with APAR OW24712

Storage Requirements

:.*i2.OS/390

When installing OS/390 TCP/IP OpenEdition, you need to consider the size of your system, storage capacity, number of applications, and the number of products.

Important Note

The amount of storage required by TCP/IP depends on the TCP/IP configuration options.

For more information on estimating storage requirements and region sizes for your configuration, see the section on OpenEdition tuning in the *TCP/IP: Performance Tuning Guide*.

TCP/IP Publication Libraries

Table 8 identifies the books in both the TCP/IP V3R2 for MVS and OS/390 TCP/IP OpenEdition libraries. Order numbers that begin with G or S indicate unlicensed documentation. You can order these books by subscribing to the IBM System Library Subscription Service (SLSS) or by contacting your IBM representative. Order numbers that begin with L contain licensed documentation. You can order these books only through your IBM representative.

Table 8 (Page 1 of 4). Library Changes for TCP/IP

OS/390 TCP/IP OpenEdition Title and Order Number	TCP/IP V3R2 Title and Order Number	Use this book to:
Installing		
Program Directory for JTCP329	*Program Directory for HTCP320* (shipped with the distribution media)	Install and maintain TCP/IP.
Migrating and Converting		
OS/390 TCP/IP OpenEdition Planning and Release Guide SC31-8303	*TCP/IP for MVS: Planning and Migration Guide* SC31-7189	Plan for TCP/IP V3R2 or OS/390 TCP/IP OpenEdition. For TCP/IP V3R2, migrate from an earlier version of IBM TCP/IP for MVS. For OS/390 TCP/IP OpenEdition, coexist with TCP/IP V3R2.
Initializing and Customizing		
OS/390 TCP/IP OpenEdition Configuration Guide SC31-8304	*TCP/IP for MVS: Customization and Administration Guide* SC31-7134	Configure your TCP/IP network, servers, and applications; customize them for your installation; and administer their operation.

Table 8 (Page 2 of 4). Library Changes for TCP/IP

OS/390 TCP/IP OpenEdition Title and Order Number	**TCP/IP V3R2 Title and Order Number**	**Use this book to:**
Operating		
OS/390 TCP/IP OpenEdition User's Guide GC31-8305	*TCP/IP for MVS: User's Guide* SC31-7136	Find out how to use TCP/IP commands to transfer data sets, log on to a remote host, execute a command on a remote host, monitor the network, manage network resources, and authenticate the use of network resources. For TCP/IP V3R2, also find out how to send electronic mail, print on a remote host, and query name servers.
Offloading		
n/a	*TCP/IP for MVS: Offloading TCP/IP Processing* SC31-7133	Install, configure, customize, and use the Offload feature.

Table 8 (Page 3 of 4). Library Changes for TCP/IP

OS/390 TCP/IP OpenEdition Title and Order Number	**TCP/IP V3R2 Title and Order Number**	**Use this book to:**
Application Development		
OS/390 TCP/IP OpenEdition Programmer's Reference, SC31-8308	*TCP/IP for MVS: Programmer's Reference* SC31-7135	Find out the syntax and usage of high-level application functions that you can use to write your own applications in a TCP/IP environment. For TCP/IP V3R2, the functions include authenticating access, use of distributed databases, distributed processing, network management, and device sharing. For OS/390 TCP/IP OpenEdition, the functions include distributed processing.
OS/390 C/C++ Runtime Library Reference SC28-1663	*TCP/IP for MVS: Application Programming Interface Reference* SC31-7187	For TCP/IP V3R2, write your own client or server application or adapt existing applications to communicate with each other using IBM TCP/IP for MVS sockets. This book describes the syntax and semantics of the IBM TCP/IP for MVS source code interface to write your own application programming interface using the C language, System/370* Assembler language, PL/I, COBOL, REXX, Pascal, IUCV, or VMCF sockets. For OS/390 TCP/IP OpenEdition, write your own client or server application or adapt existing applications to communicate with each other using IBM OS/390 OpenEdition sockets.
n/a	*TCP/IP for MVS: CICS TCP/IP Socket Interface Guide and Reference* SC31-7131	Set up the CICS TCP/IP Socket Interface function and write application programs that use it.
n/a	*TCP/IP for MVS: IMS TCP/IP Application Development Guide and Reference* SC31-7186	Set up the IMS TCP/IP system services, and write application programs that use them.
Diagnosing and Determining Problems		
OS/390 TCP/IP OpenEdition Diagnosis Guide SC31-8492	*TCP/IP for MVS: Diagnosis Guide* LY43-0105	Learn how to diagnose common TCP/IP problems and determine whether a specific problem is in the product code. This book explains how to gather information for and describe problems to the IBM Software Support Center.

Table 8 (Page 4 of 4). Library Changes for TCP/IP

OS/390 TCP/IP OpenEdition Title and Order Number	**TCP/IP V3R2 Title and Order Number**	**Use this book to:**
Messages		
OS/390 TCP/IP OpenEdition Messages and Codes SC31-8307	*TCP/IP for MVS: Messages and Codes* SC31-7132	Understand the meaning of messages and codes issued during TCP/IP processing and the appropriate user, operator, or system programmer response.
Performance and Tuning		
See the OpenEdition Tuning section of *TCP/IP: Performance Tuning Guide* for the general description of OE tuning.	*TCP/IP: Performance Tuning Guide*	Improve the performance of your network operations.
Network Print Facility		
n/a	*TCP/IP for MVS: Network Print Facility* SC31-8074	Prepare your network to route VTAM, JES2, or JES3 printer output to remote printers using TCP/IP.
OpenEdition Applications Feature		
Note: Information on the OE applications that was part of the TCP/IP Version 3 for OpenEdition MVS Applications Feature and is now part of OS/390 TCP/IP OpenEdition has been merged into the OS/390 TCP/IP OpenEdition library.	*TCP/IP Version 3 for OpenEdition MVS: Applications Feature Guide* SC31-8069	For TCP/IP V3R2, plan and migrate your TCP/IP applications for the OpenEdition MVS environment.

TCP/IP Message IDs

See Appendix B, “TCP/IP Message IDs” on page 47 for a description of the TCP/IP components and their corresponding message IDs.

Chapter 3. Running Coexisting Stacks

This chapter describes how to define and run the TCP/IP V3R2 stack and the OS/390 TCP/IP OpenEdition stack on OS/390 Release 3. Before you define your stacks, decide if you need to do the following:

- Support both OpenEdition (OE) applications and non-OE applications
- Use single or multiple transport providers (protocols/stacks)
- Configure the OE applications to communicate over a single stack or over multiple stacks

Defining Two TCP/IP Stacks at the Same Time

How you define TCP/IP V3R2 and OS/390 TCP/IP OpenEdition to run at the same time on OS/390 R3 depends on your decisions and on your system configuration. The key differences in configurations depend on the type of physical file systems (PFS) used, the transport provider (the TCP/IP stack), and the techniques used to pass data between the socket application program address space and the TCP/IP system address space.

In OpenEdition MVS, physical file systems (PFS) include the Integrated Sockets AF_INET PFS, the converged sockets AF_INET usually known as the Common Internet Physical File System (C-INET PFS), and the Hierarchical File System (HFS). The Integrated Sockets AF_INET PFS and C-INET enable applications to access the network. The HFS PFS allows applications to access files.

The HFS PFS passes file requests from an OE application (OE C program), through DFSMS/MVS, to the HFS where the traditional files or special character files are located.

The Integrated Sockets AF_INET PFS and the C-INET PFS handle socket requests from the C programs for the V3R2 stack and socket requests from the OE applications for the OS/390 TCP/IP OpenEdition stack. The Integrated Sockets AF_INET PFS and the C-INET PFS enable an OE application program to communicate with a socket program on a TCP/IP host on an attached network using transport provider (TCP/IP stack) functions.

Depending on the number of stacks you want to run on the OE sockets, you can use the Integrated Sockets AF_INET PFS or the converged sockets AF_INET (Common-INET) PFS. The Integrated Sockets AF_INET PFS supports one TCP/IP stack at a time. The Integrated Sockets AF_INET is used when OE applications communicate through a single transport provider. The Common-INET (C-INET) supports concurrent access to multiple stacks. C-INET is used when OE applications communicate through multiple transport providers. The C-INET PFS contains a pre-router that determines how individual socket calls are passed to the connected transport providers (TCP/IP stacks).

Specify your choice of the Integrated Sockets AF_INET PFS or the C-INET PFS in the NETWORK DOMAINNAME parameter of SYS1.PARMLIB(BPXPRMxx).

OE applications may be run on either TCP/IP V3R2 or OS/390 TCP/IP OpenEdition. The advantages of using OS/390 TCP/IP OpenEdition include better performance and the availablity of several new applications. If you want to run non-OE applications you must configure a TCP/IP V3R2 stack. C-INET PFS supports concurrent access to multiple stacks. You can run up to a total of eight stacks concurrently; however up to 32 stacks can be defined in the SYS1.PARMLIB(BPXPRMxx).

Note: If you plan to use C-INET to run the OE applications on the V3R2 stack and on the OS/390 TCP/IP OpenEdition stack as shown in Figure 8 on page 40, you will need to be careful about port number assignments. The port number assignments for the OE servers on the OS/390 TCP/IP OpenEdition stacks must be different from the port number assignments for the non-OE servers on the V3R2 stacks.

The simplest configuration is to run OE applications on OS/390 TCP/IP OpenEdition and to run non-OE sockets on V3R2 as shown in Figure 6 on page 33 or as shown in Figure 7 on page 36.

Port Management Overview

When there is a single transport provider, and the relationship of server to transport provider is 1:1, port management is relatively simple. The port number can be reserved for the server in the PROFILE.TCPIP for that single transport provider using the PORT statement.

Port management becomes more complex when we enter an environment where there are multiple transport providers (either multiple instances of OS/390 TCP/IP OpenEdition or a mixture of OS/390 TCP/IP OpenEdition and TCP/IP V3R2) and potentially multiple combinations of the same server (for example, OE and classic TELNET).

In a multiple transport provider environment, the following questions need to be resolved:

- Is the server generic or does the server have an affinity for one instance of the transport providers?
- How can ports be reserved across multiple transport providers? When is the port reservation determined by OMVS rather than by the jobname, procedure name, or userid?
- Synchronization between BPXPARMS and PORTRANGE for ephemeral port reservation
- How can TCP/IP V3R2 distinguish between two different instances of TELNET (OE TELNET and non-OE TELNET)?

Generic Server Versus Server with Affinity for a Specific Transport Provider

Generic Server

A generic server, a server without affinity for a specific transport provider, provides service to any client on the network. FTP and TELNET are examples of generic servers. The transport provider is merely a means of connecting client and server. The service File Transfer, for example, is *not* related to the internal functioning of the transport provider and the server can communicate over any number of transport providers concurrently.

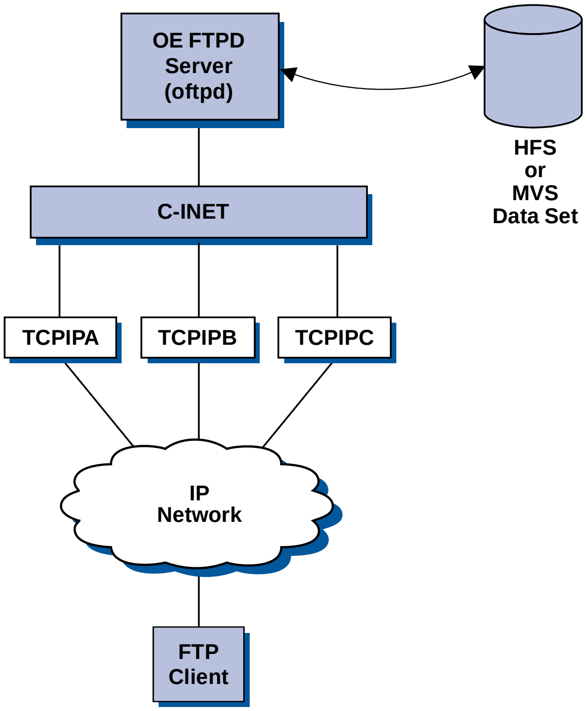


Figure 2. Generic Server

Server with Affinity for a Specific Transport Provider

When the service *is* related to the internal functioning of the transport provider (for example OROUTED, OSNMPD, and ONETSTAT, a command) there must be an explicit binding of the server application to the chosen transport provider. There must also be a method for specifying the single transport to be chosen.

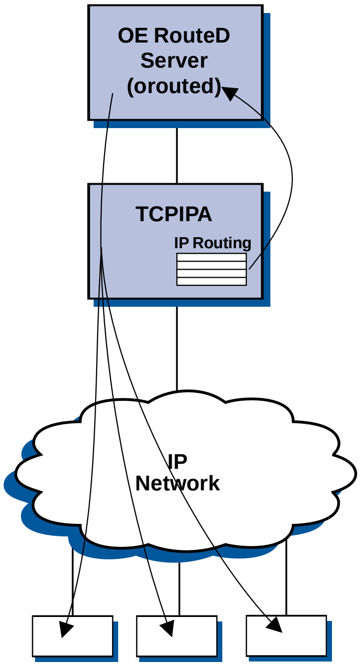


Figure 3. Server with Affinity for a Specific Transport Provider

The IBM-supplied applications which must bind to a specific transport provider use the new OE socket call setibmopt() (see OE C socket manual) to specify which TCP they have chosen. A new C function _iptcpn() described in *OS/390 C/C++ Programming Guide* enables the application to search the TCPIP.DATA file to find the name of the specific TCP/IP.

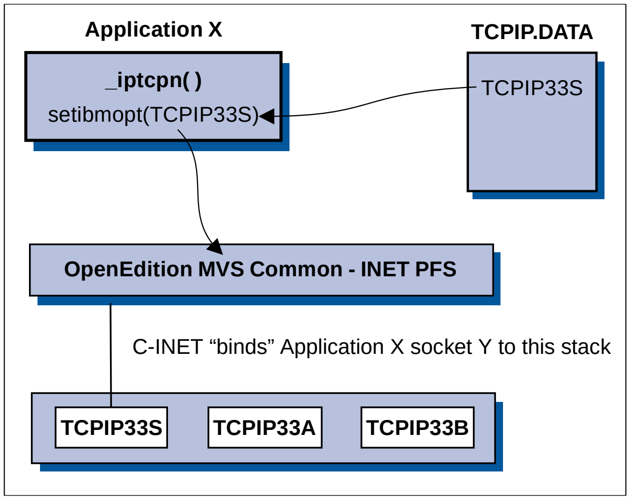

Application X
_iptcpn( )
setibmopt(TCPIP33S)
TCPIP.DATA
TCPIP33S
OpenEdition MVS Common - INET PFS
C-INET "binds" Application X socket Y to this stack
TCPIP33S
TCPIP33A
TCPIP33B

Figure 4. Example of Binding an Application to a Specific Transport Provider

Port Reservation Across Multiple Transport Providers

When there are multiple transport providers, take care to synchronize the PORT statements in each of the PROFILE.TCP files to ensure that the port reservations for each stack match the port definitions for the servers which will be using that stack.

In the illustrations for Generic Server, we have included the PORT statement for the OE FTPD to emphasize the need for synchronization.

Use of OMVS Versus Jobname

In the example PORT statements, we have used OMVS in one case, and jobname in another case. For TCPIP V3R2 when an application uses OE, we must reserve that port to OMVS. For OS/390 TCP/IP OpenEdition, it is possible to reserve the port for the originating (or HOME) Jobname, procedure name or userid. OMVS may still be used for ease of migration but more specific reservation is now possible.

Ephemeral Ports

When running with multiple transport providers, just as it is necessary to synchronize PORT reservations for specific applications across all stacks, it is required to synchronize reservations for port numbers that will be dynamically assigned across all stacks. These are the ephemeral ports above 1024, which are assigned by the stack when none is specified on the application's bind(). To reserve a group of ports in the PROFILE.TCPIP, use PORTRANGE. Specify the same PORTRANGE for each stack. In addition, we need to let the OpenEdition C-INET know which ports are guaranteed to be available on every stack. Use the BPXPARMS to do this.

- PROFILE.TCPIP
  - PORTRANGE 4000 1000 TCP OMVS; Reserved for OMVS
  - PORTRANGE 4000 1000 UDP OMVS; Reserved for OMVS
- BPXPRMxx parmlib member
  - NETWORK DOMAINNAME(AF_INET)
  - ...
  - INADDRANYPORT(4000)
  - INADDRANYCOUNT(1000)

How Can TCP/IP V3R2 Distinguish between OE Telnet and Non-OE Telnet?

When traffic comes in from the network, the UDP and TCP protocols use the port number in the packet to determine to which application the packet should be given. This means that if TCP/IP V3R2 is installed and uses OE applications in addition to non-OE applications, that OE Telnet and non-OE Telnet *cannot both use the same port number*. The PROFILE.TCPIP must contain port reservations with different port numbers for each of the servers. In Figure 5 on page 30, the non-OE servers use the default port numbers (for example, TELNET 23), and OE servers use alternate port numbers (for example, OE TELNET 1223).

When the OE servers use alternate port numbers, you should also update the TCPIP.ETC.SERVICES or /etc/services definitions for these ports to reflect the alternate port number.

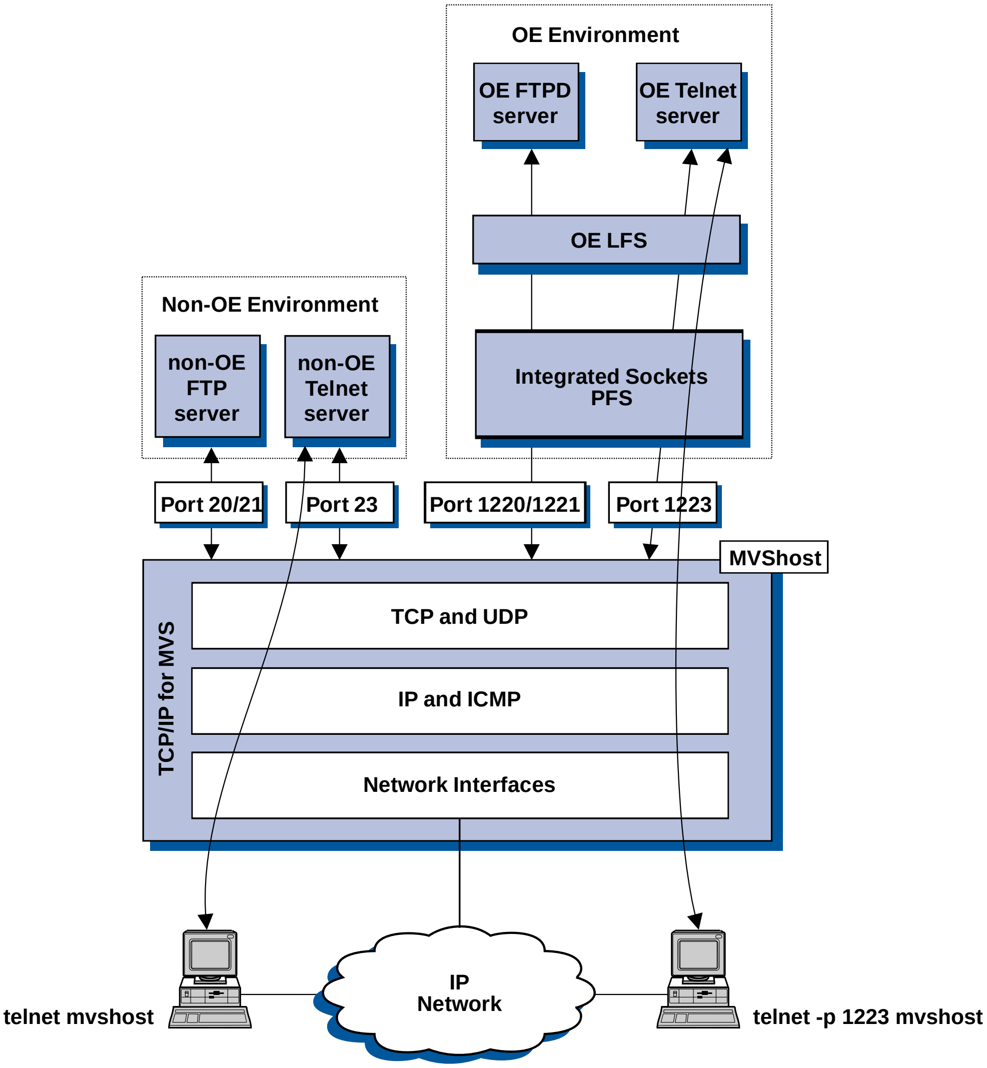


Figure 5. Example of How V3R2 Distinguishes Between OE Telnet and Non-OE Telnet

Scenario A

In this configuration, two TCP/IP stacks (one V3R2 stack and one OS/390 TCP/IP OpenEdition stack) are running on the same MVS/ESA system, but only one of them, OS/390 TCP/IP OpenEdition, is the transport provider for OpenEdition applications. The non-OE client and server applications use non-OE sockets to communicate over a network through a V3R2 stack. The OE client and server applications use OE sockets to communicate over a network (AF_INET) through a single OS/390 TCP/IP OpenEdition stack (transport provider). This configuration is shown in Figure 6 on page 33.

Port Number Assignment

Because each stack is a separate TCP/IP host and the stacks are not shared between OE and non-OE applications, the port numbers for OE server applications can be the same well-known port numbers used by the non-OE traditional MVS servers. The port number reservations are made in the PROFILE.TCPIP for each stack.

V3R2 Stack Ports

```
PORT
    20 TCP FTP32     NOAUTOLOG ; FTP Server
    21 TCP FTP32               ; FTP Server
    23 TCP INTCLIEN            ; TELNET Server
    25 TCP SMTP                ; SMTP Server
    53 TCP NAMESRV             ; Domain Name Server
    53 UDP NAMESRV             ; Domain Name Server
   111 TCP PMAP32              ; Portmap Server
   111 UDP PMAP32              ; Portmap Server
   161 UDP SNMPD               ; SNMP Agent
   162 UDP SNMPQE              ; SNMPQE Agent
   520 UDP ROUTED01            ; RouteD Server
```

OS/390 TCP/IP OpenEdition Stack Ports

```
PORT
    20 TCP OMVS                ; OE FTP Server
           DELAYACKS           ; Delay transmission acknowlegements
    21 TCP OMVS                ; OE FTP server control port
    23 TCP OMVS                ; OE TELNET Server
   111 TCP OMVS                ; OE Portmap Server
   111 UDP OMVS                ; OE Portmap Server
   161 UDP SNMPD               ; SNMP Agent
   162 UDP                     ;
   520 UDP OROUTED             ; OE RouteD Server
```

BPXPRMxx Definition for OS/390 TCP/IP OpenEdition Stack

```
/*                                                          */
/* AF_INET file system for sockets                          */
/* Integrated socket support                                */

  FILESYSTYPE TYPE(INET)  ENTRYPOINT(EZBPFINI)
  SUBFILESYSTYPE NAME(TCPV33)  TYPE(INET)  ENTRYPOINT(EZBPFINI)
  NETWORK DOMAINNAME(AF_INET)
          DOMAINNUMBER(2)
          MAXSOCKETS(10000)
          TYPE(INET)
```

Even if you are using Integrated Sockets AF_INET to run 1 OS/390 TCP/IP OpenEdition stack, you should define a subfilesystype so that the name of the stack does not have to be entered when you run OE Netstat.

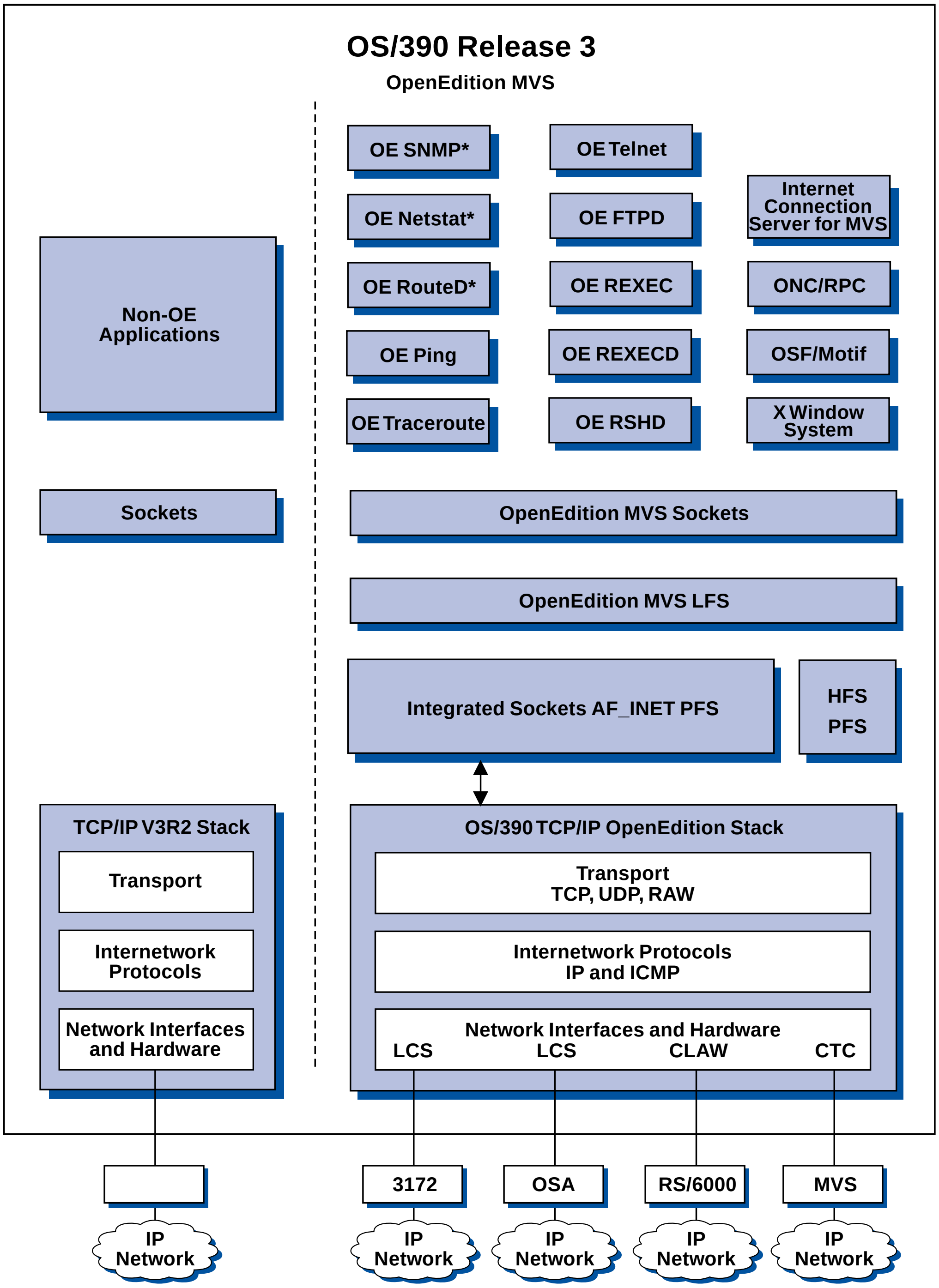


Figure 6. Scenario A. OS/390 TCP/IP OpenEdition Stack Selection by Integrated Sockets AF_INET PFS and V3R2 Stack Selection by a Different PFS

Scenario B

In this configuration, three TCP/IP stacks are running on the same OS/390 system, but only the two OS/390 TCP/IP OpenEdition stacks, TCPIPA and TCPIPB, communicate over the C-INET. The third TCP/IP stack (TCPIP32) communicates through the traditional (non-OE) sockets. The traditional non-OE client and server applications use non-OE sockets to communicate over a network through the third stack, a V3R2 stack (TCPIP32). This configuration is shown in Figure 7 on page 36.

Port Number Assignment

Because the stacks are selected differently, the port numbers for OE server applications on OS/390 TCP/IP OpenEdition can be the same well-known port numbers used by the non-OE traditional MVS servers on V3R2. The port number reservations are made in the PROFILE.TCPIP for each stack.

V3R2 Stack Ports

```
PORT
    20 TCP FTP32     NOAUTOLOG ; FTP Server
    21 TCP FTP32               ; FTP Server
    23 TCP INTCLIEN            ; TELNET Server
    25 TCP SMTP                ; SMTP Server
    53 TCP NAMESRV             ; Domain Name Server
    53 UDP NAMESRV             ; Domain Name Server
   111 TCP PMAP32              ; Portmap Server
   111 UDP PMAP32              ; Portmap Server
   161 UDP SNMPD               ; SNMP Agent
   162 UDP SNMPQE              ; SNMPQE Agent
   520 UDP ROUTED01            ; RouteD Server
```

OS/390 TCP/IP OpenEdition Stack Ports

```
PORT
    20 TCP OMVS                ; OE FTP Server
           DELAYACKS           ; Delay transmission acknowledgments
    21 TCP OMVS                ; OE FTP server control port
    23 TCP OMVS                ; OE TELNET Server
   111 TCP OMVS                ; OE Portmap Server
   111 UDP OMVS                ; OE Portmap Server
   161 UDP SNMPD               ; SNMP Agent
   162 UDP                     ; OE SNMPQE Agent
   520 UDP OROUTED             ; OE RouteD Server
```

BPXPRMxx Definition for OS/390 TCP/IP OpenEdition

Here are the definitions for SYS1.PARMLIB(BPXPRMxx) to run the OE applications on two OS/390 TCP/IP OpenEdition stacks, TCPIPA and TCPIPB. The third stack that is running is a V3R2 stack that is defined traditionally to use non-OE applications on a sockets program other than OE MVS Sockets.

```
/*                                                          */
/*  MULTIPLE TCPIP DEFINITIONS                              */
/*                                                          */
  FILESYSTYPE TYPE(CINET)  ENTRYPOINT(BPXTCINT)
  SUBFILESYSTYPE NAME(TCPIPA) TYPE(CINET)  ENTRYPOINT(EZBPFINI)
     DEFAULT
  SUBFILESYSTYPE NAME(TCPIPB)  TYPE(CINET)  ENTRYPOINT(EZBPFINI)
  NETWORK DOMAINNAME(AF_INET)
          DOMAINNUMBER(2)
          MAXSOCKETS(10000)
          TYPE(CINET)
          INADDRANYPORT(2000)
          INADDRANYCOUNT(100)
```

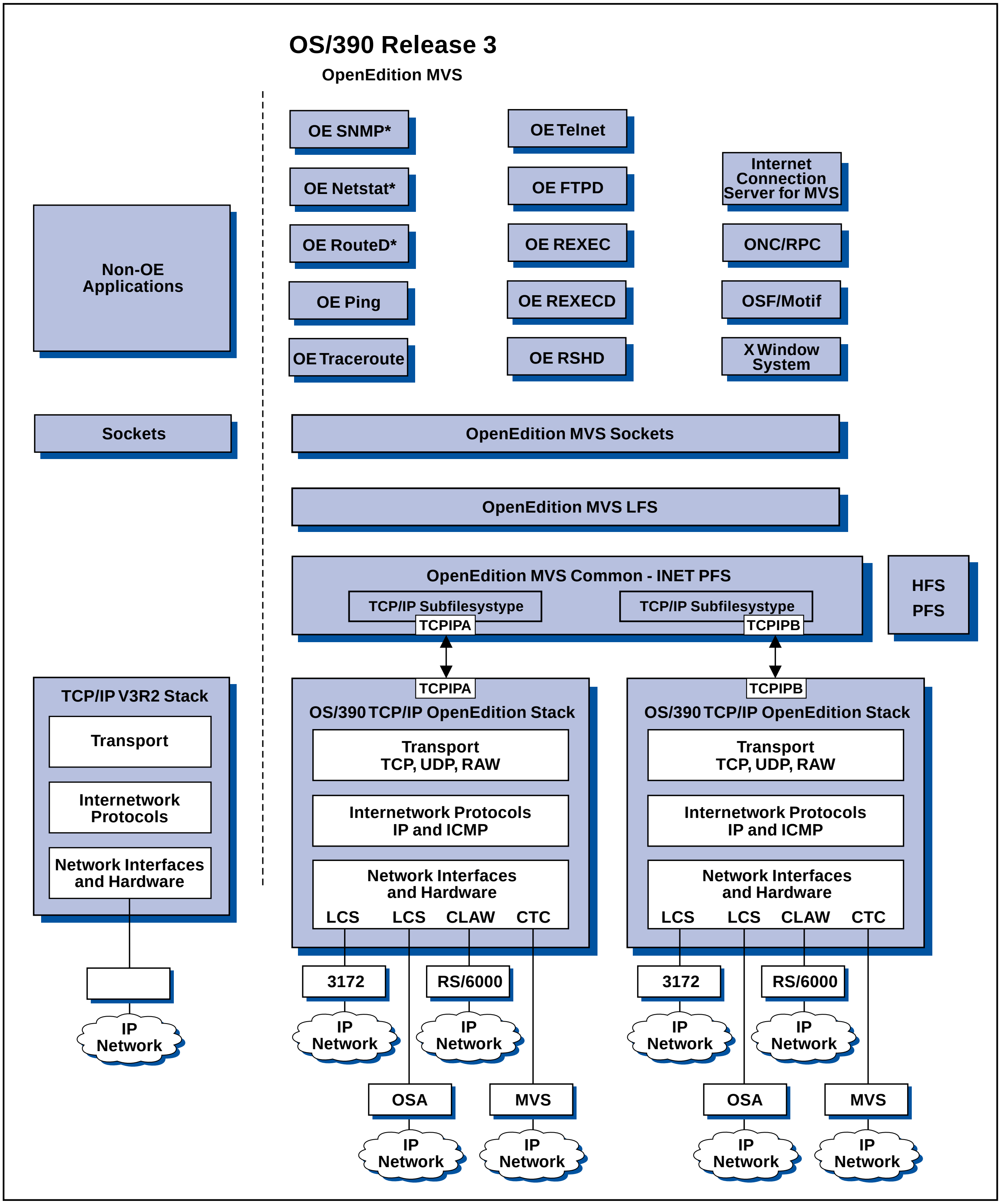


Figure 7. Scenario B. OS/390 TCP/IP OpenEdition Stack Selection Using C-INET

Scenario C

In this configuration, two TCP/IP stacks, TCPIPA and TCPIP32, are running on the same OS/390 System and the stacks communicate over the C-INET to run the OE client and server applications. The second TCP/IP stack (TCPIP32) also communicates through the traditional (non-OE) sockets to run traditional non-OE client and server applications. This configuration is shown in Figure 8 on page 40. Scenario C is a complex configuration.

You may find running C-INET as in Scenario B much simpler and more predictable than running Scenario C. The three OE applications that bind to a specific transport provider can bind to an OS/390 TCP/IP OpenEdition stack but cannot bind to a V3R2 stack. OE SNMP, OE RouteD, and OE Netstat are the three applications that cannot bind to a V3R2 stack.

Port Number Assignment

Because the stacks are selected by C-INET, the port numbers for OE server applications on OS/390 TCP/IP OpenEdition and for the OE server applications on V3R2 must be the same. However, the port numbers for OE server applications on V3R2 must be different from the non-OE server applications on V3R2 that use the traditional non-OE sockets. In the example shown here, the well-known port numbers are reserved for the non-OE traditional servers on V3R2 and different port numbers are reserved for the OE servers on V3R2. The port number reservations are made in the PROFILE.TCPIP for each stack.

V3R2 Stack Ports

```
PORT
    20 TCP T18AFTPC  NOAUTOLOG ; FTP Server data port
    21 TCP T18AFTPC            ; FTP Server control port
    23 TCP INTCLIEN            ; TELNET Server
    25 TCP T18ASNMP            ; SMTP Server
    53 TCP T18DNS              ; Domain Name Server
    53 UDP T18DNS              ; Domain Name Server
   111 TCP PMAP32              ; Portmap Server
   111 UDP PMAP32              ; Portmap Server
   161 UDP T18ASNMD            ; SNMP Agent
   162 UDP T18ASNMQ            ; SNMPQE Agent
   520 UDP T18ROUT             ; RouteD Server
  1211 TCP OMVS                ; OE Portmap Server
  1211 UDP OMVS                ; OE Portmap Server
  1212 TCP OMVS                ; OE Remote Execution
  1213 TCP OMVS                ; OE Rlogin
  1214 TCP OMVS                ; OE Remote Execution
  1220 TCP OMVS                ; OE FTP Server
           DELAYACKS           ; Delay transmission acknowlegments
  1221 TCP OMVS                ; OE FTP server control port
  1223 TCP OMVS                ; OE TELNET Server
  1261 UDP OSNMPD              ; OE SNMP Agent
  1262 UDP                     ; OE SNMPQE Agent
  1520 UDP OROUTE              ; OE RouteD Server
```

OS/390 TCP/IP OpenEdition Stack Ports

```
PORT
  1211 TCP OMVS               ; OE Portmap Server
  1211 UDP OMVS               ; OE Portmap Server
  1212 TCP OMVS               ; OE Remote Execution
  1213 TCP OMVS               ; OE Rlogin
  1214 TCP OMVS               ; OE Remote Execution
  1220 TCP OMVS               ; OE FTP Server
           DELAYACKS          ; Delay transmission acknowledgments
  1221 TCP OMVS               ; OE FTP server control port
  1223 TCP OMVS               ; OE TELNET Server
  1261 UDP OSNMPD             ; OE SNMP Agent
  1262 UDP                    ;
  1520 UDP OROUTED            ; OE RouteD Server
```

BPXPRMxx Definition

Here are the definitions for SYS1.PARMLIB(BPXPRMxx) to run all of OE applications on one OS/390 TCP/IP OpenEdition stack, TCPIPA, and some of the OE applications on one V3R2 stack, TCPIP32. The second stack that is running is a V3R2 stack that is defined in BPXPRMxx to run the OE applications under C-INET as shown below. The V3R2 stack must also be defined traditionally to run the non-OE applications on a sockets program other than OE MVS Sockets. Refer to the *TCP/IP for MVS: Customization and Administration Guide* for the information on how to define the stack to run on traditional non-OE sockets.

```
/*                                                         */
/*  MULTIPLE TCPIP DEFINITIONS                             */
/*                                                         */
  FILESYSTYPE TYPE(CINET)  ENTRYPOINT(BPXTCINT)
  SUBFILESYSTYPE NAME(TCPIPA)  TYPE(CINET)  ENTRYPOINT(EZBPFINI)
     DEFAULT
  SUBFILESYSTYPE NAME(TCPIP32) TYPE(CINET)  ENTRYPOINT(BPXTIINT)
  NETWORK DOMAINNAME(AF_INET)
          DOMAINNUMBER(2)
          MAXSOCKETS(10000)
          TYPE(CINET)
          INADDRANYPORT(2000)
          INADDRANYCOUNT(100)
```

In TCP/IP for MVS, the NAME keyword of the SUBFILESYSTYPE statement of the BPXPRMS contains the RACF started task USERID. In this example, the USERID is the same as the jobname.

In OS/390 TCP/IP OpenEdition, the NAME keyword of the SUBFILESYSTYPE statement of the BPXPRMS contains either the PROCNAME or the dot qualifier for the PROCNAME.

For example:

```
  S TCPIPA                   When starting the TCPIPA  procedure,

  SUBFILESYSTYPE NAME(TCPIPA)   make sure there is a corresponding
                                 BPXPRM SUBFILESYSTYPE NAME.


  S TCPIPA.TCPIP33A          When starting this dot qualified procedure,

  SUBFILESYSTYPE NAME(TCPIP33A)   make sure there is a BPXPRM
                                  SUBFILESYSTYPE NAME that corresponds
                                  to the dot qualifier.
```

If you supply the wrong NAME field for OS/390 TCP/IP OpenEdition, you will see the following message:

```
EZZ4203I OPENEDITION-TCP/IP CONNECTION ERROR FOR TCPIP33-BPX1IOC,
8008C981,FFFFFFFF,0000009E,12B2005A
```

If you supply the wrong NAME field for the TCP/IP for MVS, you will see the following message:

```
EZY2141E OPENEDITION-TCP/IP CONNECTION ERROR FOR IBMUSER -BPX1IOC ,
FFFFFFFF,0000009E,12B2005A
```

The reason code 5A in both messages translates to: “The file_system_type specified must match one of the FILESYSTYPE statements in BPXPRMxx parmlib member used to start Open MVS.”

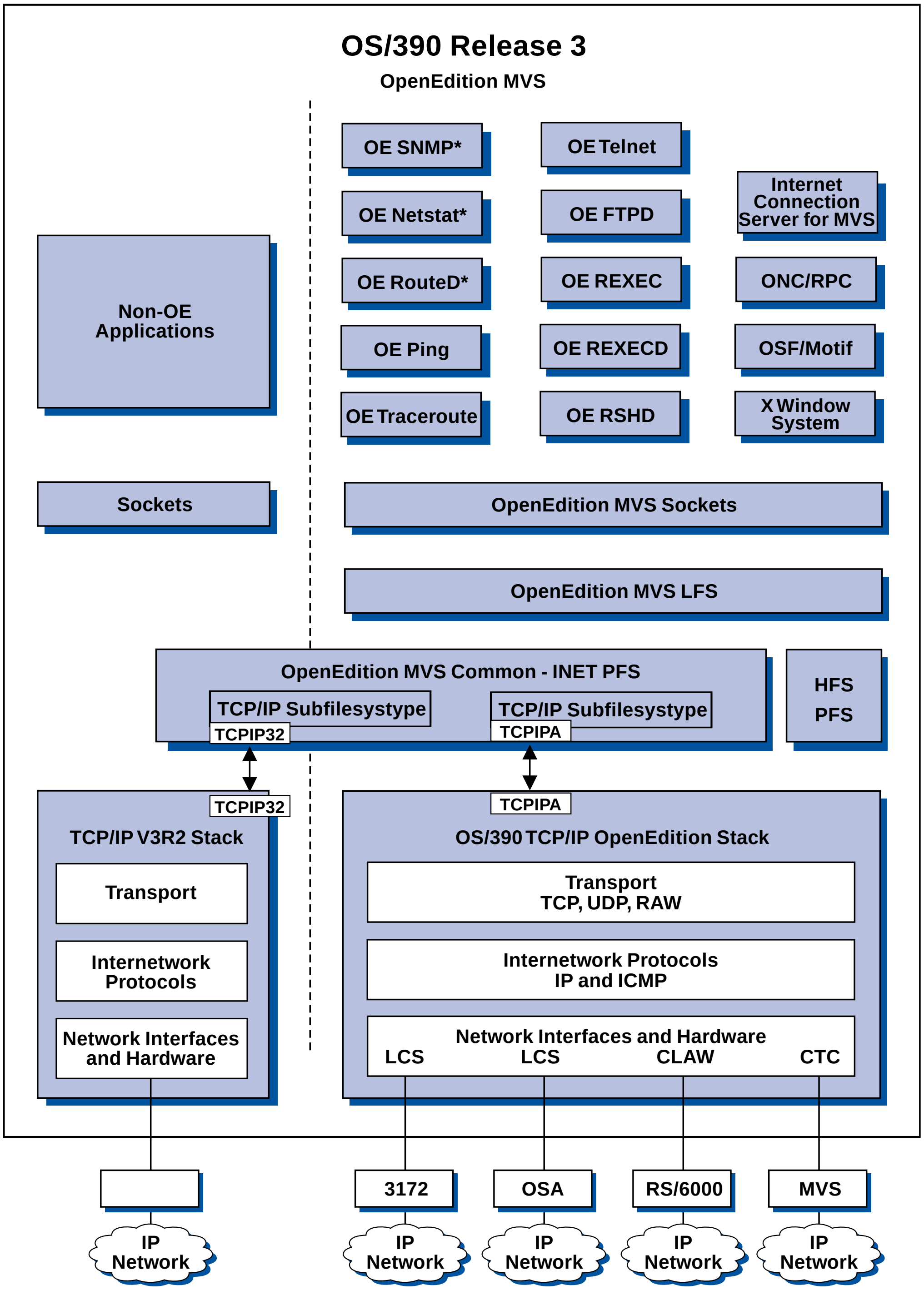


Figure 8. Scenario C. The OS/390 TCP/IP OpenEdition Stack Running OE Applications and V3R2 Stack Running OE or Classic Applications.

The OE applications marked with an asterisk run exclusively on an OS/390 TCP/IP OpenEdition stack.

Appendix A. New and Changed Commands and Statements

This appendix lists:

- New and changed MVS operator commands
- New and changed stack characteristics
- New and changed device, link, and interface statements
- New and changed RAS statements
- New SNMP subagent statement
- Miscellaneous changed and unchanged statements
- Obsolete storage and pool statements
- Unsupported statements

New and Changed MVS Operator Commands

Processing Obeyfiles

VARY OBEYFILE replaces OBEYFILE. The advantage of VARY OBEYFILE is two-fold. It uses the standard MVS interface and it grants RACF authorization through MVS.VARY.TCPIP.OBEY. The OBEYFILE command is eliminated for OS/390 TCP/IP OpenEdition.

In TCP/IP 3.2, TSO users used the OBEYFILE TSO command to administer TCP/IP and change its characteristics. This required authorization by an internal OBEYLIST parameter.

In OS/390 TCP/IP OpenEdition, you must obtain TCP/IP MVS operator authority first. Then, to authorize a user to administer TCP, you must grant RACF authority to MVS.VARY.TCPIP.OBEYFILE.

You can use one of the following commands:

- VARY TCPIP,,CMD=Obeyfile,DSN=data.set.name
- V TCPIP,procname,CMD=Obeyfile,DSN=data.set.name

Canceling Connections

VARY DROP allows MVS operator to cancel TCP/IP connections. The advantage of VARY DROP is two-fold. It uses the standard MVS interface and it grants RACF authorization through MVS.VARY.TCPIP.DROP. Canceling connections is performed in a manner similar to processing obeyfiles.

In OS/390 TCP/IP OpenEdition, you must obtain TCP/IP MVS operator authority first. Then, to authorize a user to administer TCP, you must grant RACF authority to MVS.VARY.TCPIP.DROP.

You can use one of the following commands:

- VARY TCPIP,,CMD=DRop,CONN=connid
- V TCPIP,proc,CMD=DRop,CONN=connid

New and Changed Stack Characteristics

If you use the new statements, IPCONFIG, TCPCONFIG, and UDPCONFIG, in place of the ASSORTEDPARMS statement, you can eliminate the side effects of ASSORTEDPARMS.

When you use the ASSORTEDPARMS statement and you specify only one parameter on the ASSORTEDPARMS statement, all the parameters that are not specified are turned off. There is *no cumulative effect* for ASSORTEDPARMS. Therefore, when you set ASSORTEDPARMS, all parameters must be known whether or not you are explicitly setting them. Also, as you add more parameters, ASSORTEDPARMS becomes more unwieldy.

To eliminate these side effects, use IPCONFIG, TCPCONFIG, and UDPCONFIG instead of ASSORTEDPARMS.

Note: No new parameters will be added to ASSORTEDPARMS. All parameters will go into IPCONFIG, TCPCONFIG or UDPCONFIG.

The new configuration statements and parameters are as follows:

- IPCONFIG
  - ARPTO
  - CLAWUSEDOUBLENOP*
  - DATAGRAMFWD NODATAGRAMFWD*
  - IGNOREREDIRECT*
  - REASSEMBLYTIMEOUT
  - STOPONCLAWERROR*
  - TTL
- TCPCONFIG
  - INTERVAL (Keepalive)
  - RESTRICTLOWPORTS* UNRESTRICTLOWPORTS
  - TCPSENDBUFRSIZE
  - TCPRCVBUFRSIZE
- UDPCONFIG
  - RESTRICTLOWPORTS* UNRESTRICTLOWPORTS
  - UDPCHKSUM NOUDPCHKSUM
  - UDPQUEUELIMIT NOUDPQUEUELIMIT*
  - UDPSENDBFRSIZE
  - UDPRCVBFRSIZE

Parameters marked with an asterisk also exist for ASSORTEDPARMS. The ASSORTEDPARMS equivalent for NODATAGRAMFWD is NOFWD.

IPCONFIG

ARPTO Specifies the number of seconds between creation or revalidation and deletion of ARP table entries.

CLAWUSEDOUBLENOP Forces channel programs for CLAW devices to have two NOP CCWs to end the channel program. This parameter is required for some vendor devices used on first level MVS systems.

DATAGRAMFWD Enables the transfer of data between networks. NODATAGRAMFWD is identical to ASSORTEDPARMS NOFWD and is used for security. DATAGRAMFWD is the default at IPL time.

IGNOREREDIRECT Ignore ICMP Redirect packets. This parameter is required when using ORouteD.

REASSEMBLYTIMEOUT IP reassembly time-out value in seconds

STOPONCLAWERROR Stops channel programs when a device error is detected

TTL value IP time to live or hop count

TCPCONFIG

INTERVAL Number of minutes TCP waits after receiving a packet before it sends a keep-alive packet for that connection

RESTRICTLOWPORTS UNRESTRICTLOWPORTS
Determines whether TCP ports 1 through 1023 are reserved for users by the PORT statement

TCPSENDBFRSIZE Defaults to approximately 16K, which is the TCP send buffer size

TCPRCVBUFRSIZE Defaults to approximately 16K, which is the TCP receive buffer size

UDPCONFIG

RESTRICTLOWPORTS UNRESTRICTLOWPORTS
Determines whether UDP ports 1 through 1023 are reserved for users by the PORT statement.

UDPCHKSUM NOUDPCHKSUM
Determines whether or not UDP does check summing

UDPQUEUELIMIT NOUDPQUEUELIMIT
Used to set a limit UDP requests that can be queued up to a port

UDPSENDBFRSIZE Defaults to approximately 16K, which is the UDP send buffer size

UDPRCVBUFRSIZE Defaults to approximately 16K, which is the UDP receive buffer size

New and Changed Device, Link, Interface Statements

The following device statements are used to define devices that are supported in OS/390 TCP/IP OpenEdition:

- DEVICE ATM (added for SNMP management)
- DEVICE CLAW
- DEVICE CTC
- DEVICE LCS
- DEVICE VIRTUAL

The following device statements are *not* supported in OS/390 TCP/IP OpenEdition, because the devices associated with these statements are *not* supported by OS/390 TCP/IP OpenEdition:

- DEVICE CDLC
- DEVICE HCH
- DEVICE X25NPSI
- DEVICE DDN1822
- DEVICE IUCV/PVMIUCV
- DEVICE SNALU62/SNAIUCV
- DEVICE ELANS, ILANS
- DEVICE X25ICA

New Delete Statements

The following statements have been added so that devices can be deleted from a configuration without restarting the OS/390 TCP/IP OpenEdition stack:

- DELETE LINK
- DELETE DEVICE
- DELETE PORT
- DELETE PORTRANGE

DELETE DEVICE is used to delete any device without a link defined. DELETE LINK is used to delete any Link without an IP address (HOME Address).

Ports can be *unreserved*. DELETE PORT is used to delete ports, if the ports are defined by a PORT statement. If the ports are defined by the PORT statement, they can be deleted by using DELETE PORT. If the ports are reserved by PORTRANGE, they can be deleted by using DELETE PORTRANGE. All PORT parameters (protocol, user, and so forth) must be identical to reservation.

New and Changed RAS Statements

The following new RAS statements were added for OS/390 TCP/IP OpenEdition:

- ITRACE Internal trace cmd
- PKTTRACE

The following RAS statements are *not* supported in OS/390 TCP/IP OpenEdition:

- TRACE
- LESSTRACE
- MORETRACE
- NOTRACE

ITRACE and PKTTRACE replace the TRACE, LESSTRACE, MORETRACE, and NOTRACE commands.

TRUNC was added to remove sequence numbers. TRUNC allows datasets with MVS sequence numbers to be used for VARY OBEYFILE processing.

New SNMP Subagent Statement

The new SACONFIG statement is used to configure the SNMP Subagent. Its parameter ATMENABLED allows SNMP management of ATM networks. The parameters of SACONFIG are as follows:

- COMMUNITY string
- AGENT port address
- OSASF port address
- DISABLED no SNMP Subagent
- ATMENABLED to retrieve ATM network management

The SNMP subagent can be configured to listen on different ports, to monitor ATM traffic, and so forth.

Miscellaneous Changed and Unchanged Statements

The default for ASSORTEDPARMS ARPAGE was changed to 20 minutes.

The following statements are unchanged for OS/390 TCP/IP OpenEdition: BSDROUTINGPARMS, PORT, and PORTRANGE.

Obsolete Storage and Pool Statements

No storage subpools are supported in OS/390 TCP/IP OpenEdition. The following statements are unsupported:

- ACBPOOLSIZE
- ADDRESSTRANSLATIONPOOLSIZE
- CCBPOOLSIZE
- ENVELOPEPOOLSIZE
- IPROUTEPOOLSIZE
- HPNSSTAGINGBUFFERS
- LARGEENVELOPEPOOLSIZE
- RCBPOOLSIZE
- SCBPOOLSIZE
- SKCBPOOLSIZE
- SMALLDATABUFFERPOOLSIZE
- TCBPOOLSIZE
- TINYDATABUFFERPOOLSIZE
- UCBPOOLSIZE

Unsupported Statements

The following statements are *not* supported in OS/390 TCP/IP OpenEdition:

- KEEPALIVE SENDGARBAGE FALSE
- INFORM
- OFFLOADAPIOTHER
- RESTRICT
- SCREEN, NOSCREEN
- SYSCONTACT, SYSLOCATION
- TIMESTAMP
- OBEYLIST

The following parameters of ASSORTEDPARMS are *not* supported in OS/390 TCP/IP OpenEdition:

- ALWAYSWTO
- MSGRYN
- NOACBCUSHION
- SMALLWIND

Appendix B. TCP/IP Message IDs

The following tables show the relationship between the OS/390 TCP/IP OpenEdition components and their corresponding message IDs.

EZY Messages

OS/390 TCP/IP OpenEdition includes messages for previously shipped OE applications. These EZY messages use the component identifiers listed in the following table.

Table 9. EZY Message Component Identifiers

EZY Component ID	**Component**
FS	FTP Server
FT	FTP Server
RC	orexec Client
RD	rexecd Daemon
RG	RPCGEN
RS	RSHD Client
TE	Telnet Server - Executive
TO	Telnet Server - Other
TS	Telnet Server - State
TU	Telnet Server - Utility
TY	Telnet - System
XM	OSF/Motif
XN	OSF/Motif
XO	OSF/Motif
XP	OSF/Motif
XQ	OSF/Motif
XR	OSF/Motif
XU	OSF/Motif
XW	X Window System

EZZ Messages

The following table describes the message ranges for the EZZ messages used by OS/390 TCP/IP OpenEdition.

Table 10 (Page 1 of 2). EZZ Message Ranges

EZZ Message ID Range	**Component**
EZZ0050I–EZZ0345I	Configuration and Initialization
EZZ2351I–EZZ2382I	onetstat

Table 10 (Page 2 of 2). EZZ Message Ranges

EZZ Message ID Range	Component
EZZ3100I–EZZ3119I	oping
EZZ3200I–EZZ3220I	SNMP SubAgent
EZZ3250I–EZZ3255I	VARY Command
EZZ3300I–EZZ3327I	SNMPCLIM
EZZ4200I–EZZ4210I	Initialization
EZZ4300I–EZZ4315I	If Layer
EZZ4820I–EZZ4993I	orouted
EZZ6201I–EZZ6272I	osnmpd
EZZ6354I–EZZ6376I	otracert

Appendix C. The Hierarchical File System

Data sets and files are comparable terms. If you are familiar with MVS, you probably use the term data set to describe a unit of data storage. If you are familiar with AIX or UNIX, you probably use the term file to describe a named set of records stored or processed as a unit. In the OS/390 TCP/IP OpenEdition environment, the files are arranged in a hierarchical file system (HFS). If you are unfamiliar with hierarchical file systems, please read on.

OpenEdition Hierarchical File System Concepts

The *hierarchical file system* allows you to set up a file hierarchy that consists of:

- HFS **files**, which contain data or programs. A file containing a load module or shell script or REXX program is called an *executable file*. Files are kept in directories.
- **Directories**, which contain files, other directories, or both. Directories are arranged hierarchically, in a structure that resembles an upside-down tree, with root directory at the top and the branches at the bottom. The **root** is the first directory for the file system at the peak of the tree and is designated by a slash (/).
- Additional local or remote **file systems**, which are mounted on directories of the root file system or of additional file systems.

To the MVS system, the file hierarchy is a collection of hierarchical file system (HFS) data sets. Each HFS data set is a mountable file system. The root file system is the first file system mounted. Subsequent file systems can be logically mounted on a directory within the root file system or on a directory within any mounted file system.

Except for the direction of the slashes, the hierarchical file system is similar to a Disk Operating System (DOS) or an OS/2* file system.

Each *mountable* file system resides in a hierarchical file system (HFS) data set on direct access storage. DFSMS/MVS manages the HFS data sets and the physical files. Figure 9 on page 50 shows an example of a root file system.

Utilities for You: Most utilities specially provided for a system programmer are located in the **/usr/sbin** directory.

Defining the Root File System

When you install OpenEdition, you define the root file system using the ROOT statement in the BPXPRMxx member of SYS1.PARMLIB. The sample BPXPRMxx parmlib member is shown in Figure 9 on page 50. The ROOT statement is as follows:

```
ROOT   FILESYSTEM('OMVS.ROOT')
       TYPE(HFS)
       MODE(RDWR)
```

The root system is the starting point for the overall HFS file structure. It contains the root directory and any related HFS files or subdirectories.

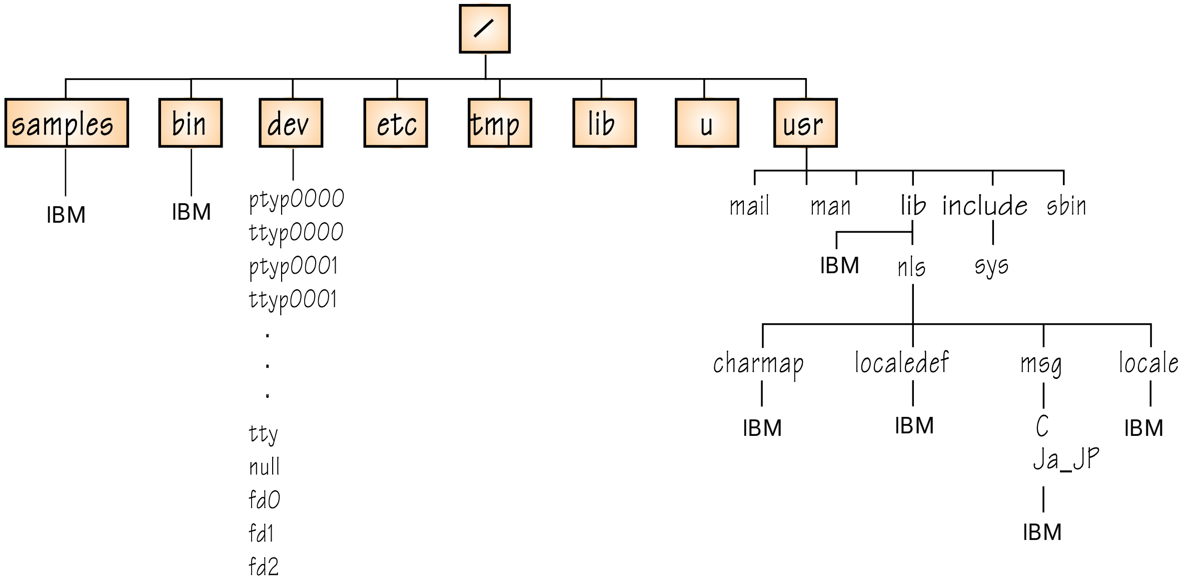


Figure 9. The Root Directory of the HFS

Directories or Files

The following files, with definitions, are OS/390 TCP/IP OpenEdition files that are installed in the HFS during the OS/390 TCP/IP OpenEdition installation. The files without definitions are OS/390 OpenEdition files that are part of the Root File System shown in Figure 9. The files without definitions are explained in the *OS/390 OpenEdition Planning Guide*.

Directory or File Definition

/

/bin

/bin/IBM

/bin/ls
This file is provided by OpenEdition MVS and is required by the FTP server.

/bin/orexec
This is the client program

/dev

/dev/console

/dev/fdx

/dev/null
This file is provided by OpenEdition MVS and is required by the FTP server.

/dev/ptypnnnn

/dev/ptypXXXX and dev/ttypXXXX

These files should be used by OE Telnet. OpenEdition provides a UNIX-like environment in MVS. UNIX does not know about devices. It understands only streams of data. Files found in /dev/ptypXXXX correlate with information being passed between TELNETD software and the master pty. Files found in /dev/ttypXXXX correlate with information being passed between the master pty and the slave pty.

Note: These device drivers are not only used by OE Telnet. For example, for every OMVS that is started up, a /dev/ptypXXXX and /dev/ttypXXXX are allocated for usage.

/dev/tty

/dev/ttypnnnn

/etc

/etc/banner

A file used by the OE Telnet server. If -h is not specified in the /etc/inetd.conf, then an additional banner is expected to be printed to the client's screen. This banner should be stored here.

/etc/bin/fomtlinp

The executable file for utmp entry is stored here. This code will update the utmp entry as well as generate the child.

/etc/ftp.data

This is the OE FTP server configuration file.

/etc/gateways

The gateways file is used to configure the routing table on the host system running OE RouteD (ORouteD).

/etc/inetd.conf

The configuration parameters for all applications started by inetd are defined in this file. Some of the applications started by inetd are OE TELNET, OE REXECD, and OE RSHD.

/etc/mibs.data

This file defines textual names for MIB variables. It is used by the osnmp command.

/etc/osnmpd.data

This file contains configuration information pertaining to osnmpd.

/etc/pw.src

This file defines a list of community names and IP addresses that can use these community names. It is used by the SNMP agent.

/etc/pw.src.cli

This file defines a list of community names and IP addresses that can use these community names. It is used by the osnmp command.

/etc/resolv.conf

The OE applications reference this file (or equivalent file specified by the environment variable RESOLVER_CONFIG) to determine the data set prefix, message case setting, and the DBCS translate tables that are to be loaded.

/etc/services
The ports for each OE application are defined here.

/etc/snmptrap.dest
This file provides a list of entities that should receive SNMP traps. It is used by osnmpd.

/etc/snmpv2.conf
This file provides configuration information for use by the osnmp command.

/etc/syslog.conf
The configuration parameters for usage of syslogd are defined in this file. For example, the daemon.* entry in this file determines where FTP error messages and trace entries are recorded.

/etc/utmpx
This file is updated by the call to fsumoclp. It contains a list of all the users who are logged in with their associated tty.

/lib

/samples

/samples/IBM

/tmp

/u

/usr/bin

/usr/bin/orpcgen
Orpcgen is the ONC RPC protocol compiler which produces client and server stubs that use low-level RPC calls. It accepts a definition of the protocol from an ".x" file written by the user.

/usr/bin/orpcinfo
Orpcinfo is a utility program that allows one to look at the portmaps of networked machines and determine what procedure numbers and versions are registered. It makes RPC calls to servers and reports the results back to the user.

/usr/bin/oportmap
Oportmap is a network service program that maps ONC RPC program and version numbers to transport-specific port numbers. The portmapper program makes dynamic binding of remote programs possible. By running the portmapper on a reserved port (111), the port numbers of other remote programs can be determined by querying the portmapper, provided these programs have registered with the portmapper.

/usr/include

/usr/include/arpa

/usr/include/arpa/IBM

/usr/include/IBM

/usr/include/net

/usr/include/net/IBM

/usr/include/netinet

/usr/include/netinet/IBM

/usr/include/rpc
All OE RPC header files are contained here.

/usr/include/sys

/usr/include/sys/IBM

/usr/lib

/usr/lib/cron

/usr/lib/cron/IBM

/usr/lib/nls

/usr/lib/nls/charmap

/usr/lib/nls/charmap/IBM

/usr/lib/nls/locale

/usr/lib/nls/locale/IBM

/usr/lib/nls/localedef

/usr/lib/nls/localedef/IBM

/usr/lib/nls/msg

/usr/lib/nls/msg/C

/usr/lib/nls/msg/C/ftpdmsg.cat
This is the default OE FTP message catalog used by the FTP server. It must be in the appropriate directory within the nlspath.

Where the server looks for the message catalog (msg.cat) depends on the value of NLSPATH and LANG environment variables. If you want to store the msg.cat elsewhere, you need to change the NLSPATH or the LANG environment variables. If the msg.cat does not exist, the software will default to the messages hard-coded within the software. These messages duplicate the English message catalog that is shipped with the product.

/usr/lib/nls/msg/C/ftpdrply.cat
This is the default reply catalog used by the OE FTP server.

There can be up to 10 reply catalogs. See “OE FTP Translation Support” in the OS/390 TCP/IP OpenEdition User's Guide.

/usr/lib/nls/msg/C/IBM

/usr/lib/nls/msg/C/rexcmsg.cat

The message catalog associated with the OE REXEC client is stored here. If this file does not exist, the software will default to the messages hard-coded within the software. These messages duplicate the English message catalog that is shipped with the product.

/usr/lib/nls/msg/C/rexdmsg.cat

This is the message catalog used by the OE REXECD server.

Where the server looks for the message catalog (rexdmsg.cat) depends on the value of NLSPATH and LANG environment variables. If you want to store the msg.cats elsewhere, you need to change the NLSPATH or the LANG environment variables. If rexdmsg.cat does not exist, the software will default to the messages hard-coded within the software. These messages duplicate the English message catalog that is shipped with the product.

/usr/lib/nls/msg/C/rshdmsg.cat

The message catalog associated with the OE RSHD client is stored here. If this file does not exist, the software will default to the messages hard-coded within the software. These messages duplicate the English message catalog that is shipped with the product.

/usr/lib/nls/msg/C/tnmsgs.cat

The message catalog used by the OE Telnet server is stored here.

Where the server looks for the message catalog (msg.cat) depends on the value of NLSPATH and LANG environment variables. If you want to store the msg.cat elsewhere, you need to change the NLSPATH or the LANG environment variables. If the msg.cat does not exist, the software will default to the messages hard-coded within the software. These messages duplicate the English message catalog that is shipped with the product.

/usr/lib/nls/msg/Ja_JP

/usr/lib/nls/Ja_JP/IBM

/usr/lib/librpclib.a

This is the location of the objects that are linked with an OE RPC application.

/usr/lib/terminfo

This file is used by the OE Telnet server. The terminal information that is verified during the tgetent call is stored here. This file requires permissions of 644. To expand it, issue the following command:

```
tic /usr/lib/terminfo/ibm.ti
```

Note: This should have been done during system setup. If this is not the first time that the tic command has been issued, you may have difficulty re-issuing this command. Make sure that you delete the symbolic links to /usr/lib/terminfo prior to re-issuing the command; then, reset the symbolic links.

/usr/lpp

/usr/lpp/tcpip/X11R6/Xamples/clients

This file contains standard clients supplied with MVS OE X Window System Support. You can use the italicized man command to display information about this client as shown below. Substitute the name of the client for the word “client” in this command.

```
man -M /usr/lpp/tcpip/X11R6/Xamples/man client
```

/usr/mail

/usr/man

/usr/man/C

/usr/man/C/cat1

/usr/man/C/cat1/IBM

/usr/man/C/cat1/rexec.1

This file contains the manual (man) page for the OE REXEC client. It provides online help for the user.

/usr/man/C/cat1/telnetd.1

This file contains the associated manual (man) pages for the OE Telnet server. It provides online help for the user.

/usr/man/C/IBM

/usr/man/Ja_JP

/usr/man/Ja_JP/cat1

/usr/man/Ja_JP/cat1/IBM

/usr/sbin

/usr/sbin/ftpd

This is the OE FTP server.

If the BPX.DAEMON is specified, the sticky bit must be set on for /usr/sbin/ftpd and ftpd must reside in an authorized MVS data set. OS/390 TCP/IP OpenEdition is shipped with the sticky bit on.

/usr/sbin/ftpdsrv

This is the client session module for OE FTP.

Whether or not BPX.DAEMON is specified, the sticky bit must be set on for /usr/sbin/ftpdsrv and ftpdsrv must reside in an authorized MVS data set.

/usr/sbin/IBM

/usr/sbin/login

The file is used by OE Telnet. The executable file for OE login is stored here.

/usr/sbin/otelnetd

The executable file for the OE Telnet server is stored here.

If BPX.DAEMON is specified, then the sticky bit must be set on, and/usr/sbin/otelnetd and otelnetd must reside in an authorized MVS data set.

/usr/sbin/rexecd

This is the OE REXECD server.

If BPX.DAEMON is specified, then the sticky bit must be set on, and /usr/sbin/rexecd and rexecd must reside in an authorized MVS dataset.

/usr/sbin/rshd

This is the OE RSHD server.

If BPX.DAEMON is specified, the sticky bit must be set on and /usr/sbin/rshd and rshd must reside in an authorized MVS data set.

/usr/share

/usr/share/lib

/usr/share/lib/terminfo

/usr/share/lib/terminfo/IBM

/usr/spool

/usr/spool/cron

/usr/spool/cron/atjobs

/usr/spool/cron/crontabs

Appendix D. Resolver Configuration Files

Files called “resolver configuration files” are used by the resolver component, which is part of the socket library. TCP/IP for MVS delivers one resolver that is used with all the socket libraries that are shipped with TCP/IP for MVS. Language Environment 1.7 delivers another resolver that is used by all OpenEdition socket programs.

Not only is there a search order for each type of data set, for each data set or file there is a hierarchy of possible locations.

For the TCP/IP for MVS resolver, the TCPIP.DATA data set is searched for using the following search order:

1. Any data set that is allocated to DD-name SYSTCPD
2. jobname.TCPIP.DATA for batch jobs, or userid.TCPIP.DATA for TSO users
3. SYS1.TCPPARMS(TCPDATA)
4. zapped-*hlq*.TCPIP.DATA

For the LE resolver, hereafter called the OE resolver, the base resolver configuration data set or file (the one that corresponds to TCPIP.DATA) is searched for using the following search order:

1. The MVS data set or HFS file that is identified in an environment variable called RESOLVER_CONFIG
2. /etc/resolv.conf
3. Any data set that is allocated to DD-name SYSTCPD
4. jobname.TCPIP.DATA for batch jobs, or userid.TCPIP.DATA for TSO users or OE shell users
5. SYS1.TCPPARMS(TCPDATA)
6. TCPIP.TCPIP.DATA

Similar hierarchies exist for all the resolver configuration data sets or files as shown in the following table:

Table 11 (Page 1 of 2). Resolver Configuration Data Sets and Files

TCP/IP for MVS Resolver	**OE Resolver**
Base resolver configuration:	**Base resolver configuration**:
1. SYSTCPD DD-name 2. jobname.TCPIP.DATA or userid.TCPIP.DATA 3. SYS1.TCPPARMS(TCPDATA) 4. zapped-*hlq*.TCPIP.DATA	1. RESOLVER_CONFIG environment variable 2. /etc/resolv.conf 3. SYSTCPD DD-name 4. jobname.TCPIP.DATA or userid.TCPIP.DATA 5. SYS1.TCPPARMS(TCPDATA) 6. TCPIP.TCPIP.DATA
Local hosts tables:	**Local hosts tables**:

Table 11 (Page 2 of 2). Resolver Configuration Data Sets and Files

TCP/IP for MVS Resolver	OE Resolver
1. jobname.HOSTS.xxxxINFO or userid.HOSTS.xxxxINFO 2. *hlq*.HOSTS.xxxxINFO	1. X_SITE and X_ADDR environment variable 2. /etc/hosts 3. jobname.HOSTS.xxxxINFO or userid.HOSTS.xxxxINFO 4. *hlq*.HOSTS.xxxxINFO
Protocol information:	**Protocol information**:
1. jobname.ETC.PROTO or userid.ETC.PROTO 2. *hlq*.ETC.PROTO	1. /etc/protocol 2. jobname.ETC.PROTO or userid.ETC.PROTO 3. *hlq*.ETC.PROTO
Service information:	**Service information**:
1. SERVICES DD-name 2. jobname.ETC.SERVICES or userid.ETC.SERVICES 3. *hlq*.ETC.SERVICES	1. /etc/services 2. jobname.ETC.SERVICES or userid.ETC.SERVICES 3. *hlq*.ETC.SERVICES
Translate table:	**Translate table**:
1. jobname.STANDARD.TCPXLBIN or userid.STANDARD.TCPXLBIN 2. *hlq*.STANDARD.TCPXLBIN	1. X_XLATE environment variable 2. *hlq*.STANDARD.TCPXLBIN 3. Internal hard-coded translate table
Note: *hlq.* comes from the DATASETPREFIX parameter in the base resolver configuration data set (TCPIP.DATA)	**Note:** *hlq.* comes from the DATASETPREFIX parameter in the base resolver configuration data set or file (TCPIP.DATA or /etc/resolv.conf Where environment variables are used, they can either refer to an MVS data set name or to an HFS file name.

Understanding TCP/IP Data Set Names with OpenEdition Services

The OpenEdition socket runtime library (RTL) functions can use the following TCP/IP data sets:

- *tcpip*.TCPIP.DATA
- *tcpip*.STANDARD.TCPXLBIN
- *tcpip*.HOSTS.SITEINFO
- *tcpip*.HOSTS.ADDRINFO
- *tcpip*.ETC.PROTO
- *tcpip*.ETC.SERVICES

tcpip represents the value of the DATASETPREFIX in the TCPIP.DATA data set, if it was found; otherwise *tcpip* is by default TCPIP.

Note: With Language Environment 1.7, for OE_SOCKETS and XOPEN_SOCKETS, initialization of the Resolver environment is deferred until the first request for the Internet Protocol (IP) Address Resolution function.

The name of the file that is ultimately opened and read by the RTL code differs for each data set. This section outlines the differences between how OpenEdition services use the data sets and how they are used by TCP/IP. For more detailed information about each data set, see *OS/390 TCP/IP OpenEdition Configuration Guide*.

Note: The TCP/IP installation job, EZAPPRFX, will *not* work for the OpenEdition socket RTL and will *not* alter the default data set prefix.

In the following data set descriptions, the values of environment variables are pointers to an OS/390 data set rather than to an HFS.

For example, a user who wants to access the HOSTS.SITEINFO data set can issue the following command from the shell:

```
export X_SITE="//'TCP31.HOSTS.SITEINFO'"
```

When a program running from the shell issues a **gethostbyname()** function, the search order will be for this data set on the host system.

If a user issues the following command, without the single quotes.

```
export X_SITE="//TCP31.HOSTS.SITEINFO"
```

the search is for the exported data set with the user ID as a prefix to the data set; that is, userid.TCP31.HOSTS.SITEINFO.

TCPIP.DATA

This file is referenced to determine, among other things, the data set prefix (DATASETPREFIX keyword) to be used when trying to access the rest of the data sets in the previous list. To access this file, the search order is:

1. The value of the environment variable **RESOLVER_CONFIG**

   This name is used as is to **fopen()** the data set. All OpenEdition I/O rules apply, and you should refer to the chapter on “Performing HFS I/O Operations” in *OS/390 C/C++ Programming Guide* for a complete discussion.

2. **/etc/resolv.conf** that resides in the HFS
3. The data set specified on the //SYSTCPD DD card

   Note: DD names do *not* work when you are using the **fork()** and **exec** function calls

4. *userid*.TCPIP.DATA for TSO/E or *jobname* TCPIP.DATA for a batch request
5. SYS1.TCPPARMS(TCPDATA)

STANDARD.TCPXLBIN

This file is referenced by the name resolver code to determine the translate data sets to be used. To access this file, the search order is:

1. The value of the environment variable **X_XLATE**

   This name is used as is to **fopen()** the data set. All OpenEdition I/O rules apply, and you should refer to the chapter on “Performing HFS I/O Operations” in *OS/390 C/C++ Programming Guide*, for a complete discussion.

2. *tcpip*.STANDARD.TCPXLBIN

tcpip represents the value of the DATASETPREFIX in the TCPIP.DATA data set, if it was found; otherwise, *tcpip* is by default TCPIP.

HOSTS.SITEINFO

This file supplies the information for the following four network host database functions:

gethostbyname()
sethostent()
gethostent()
endhostent()

and for the following network database function:

getnetbyname()

To access this file, the search order is:

1. The value of the environment variable **X_SITE**

   This name is used as is to **fopen()** the data set. All OpenEdition I/O rules apply, and you should refer to the chapter on “Performing HFS I/O Operations” in *OS/390 C/C++ Programming Guide*, for a complete discussion. The only valid data set identified by this environment variable must contain the HOSTS.SITEINFO information created by the MAKESITE command.

   It is not recommended that an X_SITE refer to an HFS file, because the two types of data sets are incompatible. For SITEINFO and ADDRINFO you can use X_SITE and X_ADDR variables, which indicates that you are not using **/etc/hosts**.

2. **/etc/hosts** that resides in the HFS

   Note: If the **X_SITE** environment variable was set and it failed to open successfully, the open of **/etc/hosts** is bypassed and the HOSTS.SITEINFO is the next data set to be opened.

   userid.HOSTS.SITEINFO for TSO/E or *jobname*.HOSTS.SITEINFO for batch requests

3. *tcpip*.HOSTS.SITEINFO

   tcpip represents the value of the DATASETPREFIX in the TCPIP.DATA data set, if it was found; otherwise, *tcpip* is by default TCPIP

HOSTS.ADDRINFO

This file supplies the information for the following four network database functions:

- **getnetbyaddr()**
- **setnetent()**
- **getnetent()**
- **endnetent()**

and this one network database function,

- **gethostbyaddr()**

To access this file, the search order is:

1. The value of the environment variable **X_ADDR**

This name is used as is to **fopen()** the data set. All OpenEdition I/O rules apply, and you should refer to the chapter on “Performing HFS I/O Operations” in *OS/390 C/C++ Programming Guide*, for a complete discussion. The only valid data set identified by this environment variable must contain the HOSTS.ADDRINFO information created by the MAKESITE command.

2. **/etc/hosts**, only if the request was **gethostbyaddr()**; otherwise, this open step is skipped. **/etc/hosts** resides in the HFS.
3. *userid*.HOSTS.ADDRINFO for TSO/E and *jobname*.HOSTS.ADDRINFO for batch requests
4. *tcpip*.HOSTS.ADDRINFO

   tcpip represents the value of the DATASETPREFIX in the TCPIP.DATA data set, if it was found; otherwise, *tcpip* is by default TCPIP.

ETC.PROTO

This file supplies the information for the following five protocol database functions:

- **getprotobynumber()**
- **getprotobyname()**
- **setprotoent()**
- **getprotoent()**
- **endprotoent()**

To access this file, the search order is:

1. **/etc/protocol** that resides in the HFS
2. *userid*.ETC.PROTO
3. *tcpip*.ETC.PROTO

   tcpip represents the value of the DATASETPREFIX in the TCPIP.DATA data set, if it was found; otherwise, *tcpip* is by default TCPIP.

ETC.SERVICES

This file supplies the information for these five services database functions;

- **getservbyport()**
- **getservbyname()**
- **setservent()**
- **getservent()**
- **endservent()**

To access this file, the search order is:

1. **/etc/services** that resides in the HFS
2. *userid*.ETC.SERVICES for TSO/E or *jobname*.ETC.SERVICES for batch requests
3. *tcpip*.ETC.SERVICES

   tcpip represents the value of the DATASETPREFIX in the TCPIP.DATA data set, if it was found; otherwise, *tcpip* is by default TCPIP.

Setting Up the OE Resolver

Although it is possible to share resolver configuration data between the OE resolver and the TCP/IP for MVS resolvers (the resolvers used by TCP/IP V3R2), it is better to set up a separate set of resolver configuration data sets or files for each of these resolvers. The OE resolver is part of Language Environment and is a separate entity, different from the TCP/IP for MVS resolver. The OE resolver is used by OpenEdition socket applications, which are part of OS/390 TCP/IP OpenEdition, the TCP/IP for MVS resolver is used by non-OE TCP/IP for MVS socket applications. The non-OE socket applications are also known as native, classical, or traditional applications. By *native* we mean socket applications that run directly on top of the TCP/IP for MVS protocol stack, for example, the TCP/IP V3R2 for MVS protocol stack, without using any OpenEdition functions at all.

The following topics are partial excerpts from *International Technical Support Organization: Accessing OpenEdition MVS from the Internet*.

- Resolver Data Formats
- /etc/resolv.conf
- /etc/protocol
- /etc/services
- /etc/hosts
- OE Resolver Translation Table
- OE Resolver Configuration Summary

Resolver Data Formats

The two resolvers support more configuration data formats for some of the data sets or files. See Table 12 for an overview of the supported formats.

Table 12. Supported Resolver Configuration Data Formats

Configuration data		**OE**	**TCP/IP**
Name	**Format**		
TCPIP.DATA or RESOLV.CONF	Text (BSD format)	√	√
PROTOCOL	Text (BSD format)	√	√
SERVICES	Text (BSD format)	√	√
HOSTS	Text (BSD format)	√	
	Binary (output from MAKESITE)	√	√
TRANSLATE table	Text		
	Binary (output from CONVXLAT)	√	√

Only the OE resolver offers you the option of placing the configuration files in your Hierarchical File System. The TCP/IP for MVS resolver is able to read configuration data from the Hierarchical File System. If you do plan to share resolver configuration data among the different resolvers, you must place all configuration data in traditional MVS data sets.

TCP/IP for MVS, TCP/IP V3R2, supplies a set of sample resolver configuration data sets that you can use as a base for creating your OE resolver configuration data sets or files.

Table 13. Sample Resolver Configuration Data

File	**TCP/IP for MVS**
TCPIP.DATA or RESOLV.CONF	*datasetprefix*.SEZAINST(TCPDATA)
PROTOCOL	*datasetprefix*.SEZAINST(PROTO)
SERVICES	*datasetprefix*.SEZAINST(SERVICES)
HOSTS	*datasetprefix*.SEZAINST(HOSTS)[1]
Translate table	*datasetprefix*.SEZATCPX(OEMVS311)

Note:

1. The text format of the TCP/IP for MVS sample for the local host file does *not* match the format that is required by the OE resolver. If you base your OE resolver local host file on this sample, you will have to edit it before the OE resolver is able to use it.

The translate table sample from TCP/IP for MVS must be processed by a TCP/IP for MVS utility called CONVXLAT to produce a binary form of the translate table. The OE resolver supports this binary form, but not the text form of the translate table. If you do not have TCP/IP for MVS available, you cannot create the required binary form of a translate table, but the OE resolver will work without it; the OE resolver has a default built-in translate table available in case no translate table data set has been configured.

All the other configuration data sets or files for the OE resolver can be created based on samples from TCP/IP for MVS (V3R2).

In the following sample setup, we chose to place as many of the OE resolver configuration files in the Hierarchical File System as supported.

/etc/resolv.conf

The resolver configuration data set or file is where you specify, among other things, if the OE resolver should use a name server or not.

By default, the OE resolver will look for /etc/resolv.conf and use that as the resolver configuration file. The contents of a sample /etc/resolv.conf file is shown in Figure 10.

The DATASETPREFIX keyword is used if the OE resolver has to allocate any standard MVS data sets. In the sample setup that is documented in this book, the OE resolver needs this prefix in order to allocate the standard TCP/IP for MVS translation table data set (*datasetprefix*.STANDARD.TCPXLBIN).

You may use the exact same syntax in your /etc/resolv.conf file as you use in your TCP/IP for MVS TCPIP.DATA data set.

```
;***********************************************************************
;                                                                      *
;     /etc/resolv.conf                                                 *
;                                                                      *
;     OpenEdition MVS Resolver on mvs18                                *
;                                                                      *
;     This file is used by the OpenEdition MVS resolver code in        *
;     all OE socket applications.                                      *
;                                                                      *
;                                                                      *
;***********************************************************************
;
TCPIPjobname TCPV33
Datasetprefix TCPIP.OMVS       ; Only used for TCPXLBIN data set
Messagecase mixed              ;
HostName mvs18oe               ; MVS18 OE environment host name
DomainOrigin itso.ral.ibm.com  ;
NSinterAddr  9.24.104.126      ; Use our own caching-only DNS first
NSinterAddr  9.24.104.108      ; Then go to rsserver.itso.ral.ibm.com
NSportAddr 53                  ;
ResolveVia UDP                 ;
ResolverTimeout 10             ; Don't wait too long
ResolverUdpRetries 1           ;
```

Figure 10. Sample /etc/resolv.conf File

Please note that the OE resolver will use the DOMAINORIGIN value from the OE resolver configuration file, but it will use the HOSTNAME of the resolver configuration data set from your default AF_INET transport provider. If you have defined your TCP/IP for MVS stack as the default AF_INET transport provider and have defined a TCPIP.DATA data set for this stack with the HOSTNAME and DOMAINORIGIN statements shown in Figure 11 on page 65, an OE socket application will receive host name HOOLIGAN on a gethostname() call, and a gethostbyname() call for host name friendly will be sent to the name server at

9.24.104.126 querying the fully qualified name of friendly.itso.ral.ibm.com based on the DOMAINORIGIN and name server IP address from the /etc/resolv.conf file.

```
 ...
HostName hooligan
DomainOrigin your.neck.of.the.woods.com
NSinterAddr  192.168.1.1
 ...
```

Figure 11. Sample Extract of TCPIP.DATA of Default Stack

Apart from the host name that is returned on a gethostname() call, all information used by the OE resolver comes from the OpenEdition resolver configuration file.

If an application wants to override the system default of /etc/resolv.conf, it can initialize the RESOLVER_CONFIG environment variable to point to an alternate resolver configuration file or data set. If you are logged on to the OpenEdition shell, the following command can be used to point to an MVS data set:

```
export RESOLVER_CONFIG="//'TCPIP.ITSC.CNTL(TCPDATA)'"
```

Please note the required syntax for referring to an MVS data set name: double quotes followed by two slashes followed by the MVS data set name enclosed in single quotes. You will have to use the same syntax for other environment variables that point to MVS data sets.

The following shell command can be used to point to an HFS file:

```
export RESOLVER_CONFIG=/etc/testresolv.conf
```

You might want to use these options if you have a requirement for different servers to use, for example, different name servers. If you implement, for example, both a Web server for internal use and a Web server for external use, you can point the Web server for external use to a different name server than the one you use internally.

/etc/protocol

The /etc/protocol file holds information about supported protocols and is used by the following resolver calls:

Getprotobyname	Get protocol number based on a protocol name
Getprotobynumber	Get protocol name based on a protocol number
Setprotoent	Prepare to read the protocol file sequentially
Getprotoent	Read next protocol file entry
Endprotoent	End reading the protocol file sequentially

The protocol number is used in the 8-bit protocol field in an IP packet header.

To create the OE resolver /etc/protocol file, we used the sample protocol member from *datasetprefix*.SEZAINST(PROTO), which has the following content (some comments removed for readability):

```
# offical name, protocol number, aliases

ip              0               # dummy for IP
icmp            1               # control message protocol
tcp             6               # tcp
udp             17              # user datagram protocol
```

Figure 12. /etc/protocol Sample

/etc/services

The /etc/services file holds information about how individual applications are assigned to socket layer port numbers. Standard applications, like telnet or FTP, are assigned port numbers inside the well-known port number range (from 0 to 1023). You can assign port numbers to your own server applications by adding entries to the /etc/services file. The content of /etc/services is typically used by a server program through a getservbyname() call, where the server program passes its own name and receives the assigned port number when the call returns. The server program can then bind its socket to the assigned port number. This technique allows you to keep port number assignments external to your server program. This is of particular importance if you want to start more instances of your server program on the same TCP/IP stack. By using different /etc/services files, each instance of the server program may be assigned alternate port numbers.

The /etc/services file is used by the following resolver calls:

Getservbyname	Get server port number based on server name
Getservbynumber	Get server name based on server port number
Setservent	Prepare to read /etc/services sequentially
Getservent	Read next entry in /etc/services
Endservent	End reading /etc/services sequentially

The port number is used on various socket calls and is also included in both the header of a TCP segment and the header of a UDP datagram.

Figure 13 on page 67 is an extract of the /etc/services file.

```
# Name          Port/protocol   Aliases

echo            7/tcp
echo            7/udp
discard         9/tcp           sink null
discard         9/udp           sink null
systat          11/tcp          users
daytime         13/tcp
daytime         13/udp
netstat         15/tcp
qotd            17/tcp          quote
chargen         19/tcp          ttytst source
chargen         19/udp          ttytst source
ftp             21/tcp
telnet          23/tcp
smtp            25/tcp          mail
time            37/tcp          timserver
time            37/udp          timserver
rlp             39/udp          resource        # resource location
nameserver      42/tcp          name            # IEN 116
whois           43/tcp          nicname
domain          53/tcp          nameserver      # name-domain server
domain          53/udp          nameserver

(more entries in actual /etc/services)
```

Figure 13. /etc/services Sample (Extract)

In general, servers use a getservbyname() call to find the assigned port number. Some servers allow you to override the port number by using a server-specific configuration or start option. The OE FTPD server, for example, allows you to pass a run-time option or program parameter:

```
PORT 7021
```

If this option is specified, the FTPD server will not use the value assigned in /etc/services, but use the value specified in the run-time option. You can use this technique to start alternate FTPD server instances on alternate port numbers.

Servers that are started by InetD use the service name, which you specify in the /etc/inetd.conf file as argument on the getservbyname() call. If you, for example, want to have your OE TelnetD server to operate on port 2023 instead of the default port 23, you can edit the line in your /etc/services file that corresponds to the telnet service name in your /etc/inetd.conf file.

```
/etc/inetd.conf (extract):
--------------------------

telnet   stream tcp nowait OMVSKERN /usr/sbin/otelnetd otelnetd

/etc/services (extract):
------------------------

telnet        2023/tcp
```

Figure 14. Assigning Port Number to the TelnetD Server

If you want to start two telnet servers in your OE environment, you can use the following set of definitions:

```
/etc/inetd.conf (extract):
--------------------------

telnet     stream tcp nowait OMVSKERN /usr/sbin/otelnetd otelnetd
testtelnet stream tcp nowait OMVSKERN /usr/test/otelnetd otelnetd

/etc/services (extract):
------------------------

telnet          23/tcp
testtelnet    2023/tcp
```

Figure 15. Assigning Port Numbers to Two TelnetD Servers

With this setup, you will have your normal OE telnet server operating on the standard port 23, and you will have your test telnet server operating on port 2023. If a telnet client connects to port 2023, the telnet client will use your test telnet server.

/etc/hosts

You can set up the local hosts file to support local host name resolution. If you use the local hosts file for this purpose, your socket applications will only be able to resolve names and IP addresses that appear in your local hosts file.

If you need to resolve host names outside your local area, you can configure the OE resolver to use a domain name server (see "/etc/resolv.conf" on page 64). If you use a domain name server, you do not need to set up any host definitions in your OE resolver configuration, but you may, however, still do so.

If you have configured your OE resolver to use a name server, it will always try to do so, unless your applications were written with a RESOLVE_VIA_LOOKUP symbol in the source code. If this is the case, all name resolution calls from such a program will always use the local hosts file. This is probably not a technique you will see for standard socket applications, but it may be a technique you could find use for when you develop your own socket applications for the OpenEdition environment.

It may also be a good idea to have some basic local hosts file available for the OE resolver to use if the name server is not reachable. If the name server does not respond to name resolution requests, the OE resolver will try to use the local hosts file.

If the name server is reachable but returns a negative reply for a name resolution request, the OE resolver will try to resolve the unqualified name through the local hosts file, if such a file is present. Assume you try to resolve the host name *friendly* and your DOMAINORIGIN is *my.wood.com*, the OE resolver will send a query to the name server for *friendly.my.wood.com*. If the name server returns a negative reply (the name is not registered), the OE resolver will look into the local hosts file for an entry of friendly and, if not found, for an entry of friendly.my.wood.com.

Due to the higher flexibility of the Domain Name System, we recommend you use a domain name server. If you have TCP/IP for MVS installed, you can configure and use the name server that comes with that product. You need to configure your OE resolver configuration file to point to the IP host, where your name server is running. It may be any host that is reachable from your OpenEdition environment. There is no requirement that the name server must run on MVS/ESA. It may, from a functional point of view, run equally well on, for example, an AIX/6000 or an OS/2 TCP/IP host.

If you set up a small TCP/IP network, the simplicity of the local hosts file approach is preferable.

The following resolver calls are serviced either through the local hosts file, or by requests sent by the OE resolver to a name server:

Gethostbyname Resolve a host name into one or more IP addresses

Gethostbyaddr Resolve an IP address into a host name

The following resolver calls only apply to your local hosts file. If you have configured your system to use a name server, these calls will bypass the name server and use your local hosts file, if you have configured one. If no local hosts file exists, the sethostent() call will return with an error.

Sethostent Prepare to read your local hosts file sequentially

Gethostent Read next entry in your local hosts file

Endhostent End reading your local hosts file sequentially

The OE resolver supports two different formats for the local hosts file:

1. The standard BSD formatted text file - as it is supported in most TCP/IP implementations.
2. The format that the TCP/IP for MVS MAKESITE utility program creates. This option is only available if you have TCP/IP for MVS installed. The MAKESITE utility program comes with TCP/IP for MVS.

You can choose freely which format you want to use in your OE resolver environment, with one exception: if your applications rely on one of the following resolver calls, you must use the format that is created by the MAKESITE utility:

Getnetbyaddr Get a net entry by name

Getnetbyname	Get a net entry by network address
Setnetent	Prepare to read the net entries sequentially
Getnetent	Get next net entry
Endnetent	End reading net entries sequentially

In most UNIX systems these calls are serviced by a file called /etc/networks, but this file is currently not supported by the OE resolver. If you use the TCP/IP for MVS MAKESITE utility, this utility supports a HOSTS.LOCAL source file according to the RFC952 syntax, which allows you to specify both host and network entries. The resulting files from MAKESITE may therefore hold both host and network entries, which is the reason why these calls are supported if you use the MAKESITE format.

If you want to use the text format, we recommend that you place your local hosts file in the Hierarchical File System under the name /etc/hosts. Please observe that some of the available documentation specifies this name differently (without the last "s"), which will not work.

See Figure 16 for a sample /etc/hosts file.

```
#
# OE Resolver /etc/hosts file on mvs18oe.
#
# The format of this file is:
#
# Internet Address      Hostname   Aliases     # Comments
#
# Items are separated by any number of blanks and/or tabs.  A '#'
# indicates the beginning of a comment; characters up to the end of the
# line are not interpreted by routines which search this file.  Blank
# lines are allowed in this file.

9.24.104.126    mvs18oe mvsoe # OE host
192.168.210.1   mvs18an       # AnyNet MVS host
192.168.210.8   mypcaa        # AnyNet gw host
9.24.104.79     mypc          # A workstation
```

Figure 16. Sample /etc/hosts File

There are some syntax requirements for the /etc/hosts file. The most important are the following:

1. A host name can have a maximum of four qualifiers:
   - Host name a.b.c.d is a valid host name.
   - Host name a.b.c.d.e is *not* a valid host name.
2. You can specify a maximum of 35 aliases per IP address.
3. A qualifier can have a maximum length of eight characters:
   - Host name mypchost.mynet is a valid host name.
   - Host name myotherpchost.mynet is *not* a valid host name.

   This restriction will be removed when APAR PN79720 is solved. The solution of this APAR will increase the length of each qualifier to 24 characters.

As there is no syntax checking of the /etc/hosts file before an application program tries to use the information in it, syntax errors may show up as error messages from application programs:

```
Error in line 2: A name qualifier must be 8 characters or less
sethostent() returned error:
             EDC5049I The specified file name could not be located.
```

The above error message was returned due to the following contents of an /etc/hosts file:

```
9.24.104.126    mvs18o  mvsoe  # OE host
9.24.104.79     mypc           # A workstation
```

If you already have a TCP/IP for MVS stack implemented and want to maintain your local hosts file in one place, you can instruct the OE resolver to use the same file format as TCP/IP for MVS uses.

You can use two different approaches:

1. Maintain your local hosts file in the source format that is accepted by the TCP/IP for MVS MAKESITE utility program. This format is documented in RFC952.
2. Maintain your local hosts file in the BSD source format that is accepted by the OE resolver and most other TCP/IP platforms.

Maintaining Shared Source in HOSTS.LOCAL

You maintain your HOSTS.LOCAL file in the format that is required by the TCP/IP for MVS MAKESITE utility program. See Figure 17 for a sample HOSTS.LOCAL data set.

```
; HOSTS.LOCAL (Input to TCP/IP for MVS MAKESITE utility)
;
; Syntax requirements documented in RFC952.
;
HOST : 9.24.104.126  : mvs18a, mvsoe ::::
HOST : 9.24.104.79   : mypc ::::
HOST : 9.24.104.80   : abc.ibm.com ::::
HOST : 9.24.104.81   : abc1, abc1.ibm.com ::::
;
NET :  9.24.104.0    : itso.ral.ibm.com :
NET :  9.0.0.0       : ibm.com :
```

Figure 17. Sample HOSTS.LOCAL Source

When you run the MAKESITE utility program, it produces two output data sets:

- *datasetprefix*.HOSTS.SITEINFO
- *datasetprefix*.HOSTS.ADDRINFO

You can instruct the OE resolver to use these data sets in two ways:

1. To use these data sets as a system default, you must ensure that there is no file called /etc/hosts in your Hierarchical File System, and in addition, you must in the resolver configuration data set or file (see "/etc/resolv.conf" on page 64)

specify the DATASETPREFIX keyword with the value of the high-level qualifier of your HOSTS.SITEINFO and HOSTS.ADDRINFO data sets.

If your DATASETPREFIX in the OE resolver configuration data set or file is TCPIP.OMVS, the OE resolver will use:

```
TCPIP.OMVS.HOSTS.SITEINFO
TCPIP.OMVS.HOSTS.ADDRINFO
```

2. If you want to override your system default for a specific application, you can set the two environment variables called X_SITE and X_ADDR to point to two data sets that are created with the MAKESITE utility. If your application executes in the shell environment, you can use the following commands to assign values to the environment variables:

```
export X_SITE="//'MYOWN.HOSTS.SITEINFO'"
export X_ADDR="//'MYOWN.HOSTS.ADDRINFO'"
```

If you use the same DATASETPREFIX for your TCP/IP for MVS resolver, the same set of HOSTS.SITEINFO and HOSTS.ADDRINFO data sets can be used by both TCP/IP for MVS and OpenEdition.

Maintaining Shared Source in /etc/hosts

TCP/IP for MVS can not directly read the /etc/hosts file, so you have to create HOSTS.SITEINFO and HOSTS.ADDRINFO data sets from your /etc/hosts source file. Because the format of the /etc/hosts file is not compatible with the format that is required by the TCP/IP for MVS MAKESITE utility, you need to write your own REXX program to process the /etc/hosts file before running the MAKESITE command. The sample REXX program in Appendix E, “Sample REXX to Create HOSTS.LOCAL from /etc/hosts” on page 75 can be used for such a purpose.

OE Resolver Translation Table

The OE resolver has to translate EBCDIC host names and IP addresses into ASCII before sending a request to a name server, and it has to translate the ASCII response from the name server into EBCDIC before presenting the result to the calling application program.

The OE resolver has a built-in translation table, so you do not have to configure a translation table if the built-in table meets your requirement.

The translate table cannot be used by the OE resolver in the supplied source format; you have to use the TCP/IP for MVS utility called CONVXLAT to convert it into a binary form that is acceptable to both TCP/IP for MVS and OpenEdition:

```
convxlat 'tcpip.sezatcpx(oemvs311)' 'tcpip.omvs.standard.tcpxlbin'
```

If your DATASETPREFIX in the OE resolver configuration data set or file has been set to TCPIP.OMVS, the OE resolver will by default use the TCPIP.OMVS.STANDARD.TCPXLBIN translate table data set.

If some applications have specific requirements that are not met by your default translate table data set, you can create alternate data sets and point to an alternate data set by setting the X_XLATE environment variable before using the resolver calls.

OE Resolver Configuration Summary

The following setup was used in a Raleigh sample environment:

1. OE resolver:
    - /etc/resolv.conf for the basic OE resolver configuration data. This file includes a DATASETPREFIX keyword with a value of TCPIP.OMVS.
    - /etc/protocol for the protocol configuration data.
    - /etc/services for the OpenEdition socket server port number assignment.
    - /etc/hosts for local host definitions. Primary source for name resolution was a name server. /etc/hosts was only used for backup purposes and to support the resolver calls that bypass a name server.
    - TCPIP.OMVS.STANDARD.TCPXLBIN as the OE resolver ASCII-EBCDIC translate table data set.
2. TCP/IP for MVS resolver:
    - TCPIP.T18OE.TCPPARMS(TCPDATA) for the TCPIP.DATA configuration data. This file includes a DATASETPREFIX keyword with the value of TCPIP.T18OE.
    - TCPIP.T18OE.PROTO for the protocol configuration data.
    - TCPIP.T18OE.SERVICES for the TCP/IP for MVS socket server port number assignment.
    - TCPIP.T18OE.HOSTS.ADDRINFO and TCPIP.T18OE.HOSTS.SITEINFO. These data sets were created from /etc/hosts by a REXX program. described in Appendix E, “Sample REXX to Create HOSTS.LOCAL from /etc/hosts” on page 75.
    - TCPIP.T18OE.STANDARD.TCPXLBIN as the TCP/IP for MVS resolver ASCII-EBCDIC translate table data set.

Appendix E. Sample REXX to Create HOSTS.LOCAL from /etc/hosts

This sample REXX program converts an /etc/hosts local file into a HOSTS.LOCAL file and executes the TCP/IP for MVS MAKESITE utility to build the corresponding *datasetprefix*.HOSTS.ADDRINFO and *datasetprefix*.HOSTS.SITEINFO data sets.

```
/* REXX */
call syscalls 'ON'       /* Allow OMVS REXX services                 */

hostfile = '/etc/hosts'  /* Input hosts file                         */
tcpiphlq = 'TCPIP.OMVS'  /* xxxxINFO data sets HLQ                   */
dotrace = 0              /* Set to 1 for application trace hooks     */

/* Let's start by reading the /etc/hosts file into a REXX stem       */

address syscall "readfile" hostfile "hostline."
if retval = -1 then do
  say hostfile 'not read, errno='errno' - errnojr='errnojr
  exit(8)
end
if hostline.0 = 0 then do
  say hostifle 'is empty - no lines read'
  exit(8)
end

/* Create or allocate existing &userid.HOSTS.LOCAL data set         */

If sysdsn(hosts.local) ¬= 'OK' then do
  address TSO "alloc fi(hostloc) da(hosts.local) lrecl(255),
  blksize(0) dsorg(ps) recfm(v b) new catalog"
  alcrc = rc
  If alcrc > 0 then do
    say 'Allocation of new 'userid()'.HOSTS.LOCAL failed - rc = 'alcrc
    exit(alcrc)
  end
end
else do
  address TSO "alloc fi(hostloc) da(hosts.local) shr"
end

/* Process /etc/hosts lines and create stem var for HOSTS.LOCAL     */

oix = 0
do i=1 to hostline.0
  If dotrace then say 'Trace input  ==> 'hostline.i
  if words(hostline.i) > 0 then do
    if left(word(hostline.i,1),1) = '#' then iterate
    if words(hostline.i) = 1 |,
    left(word(hostline.i,2),1) = '#' then do
      say 'The following line from 'hostfile' is not valid syntax:'
      say '==> 'hostline.i
      iterate
    end
    oix = oix+1
    outline.oix = 'HOST:'||word(hostline.i,1)||':'
    dropline = 0
    do x=2 to words(hostline.i)
      if left(word(hostline.i,x),1) = '#' then leave
      newdata = word(hostline.i,x)||','
      newlength = length(outline.oix) + length(newdata)
      if newlength+4 > 255 then do
        say 'The following output line is too long and will be dropped:'
        say '==> 'outline.oix
        dropline = 1
        leave
      end
      else do
        outline.oix = outline.oix||word(hostline.i,x)||','
      end
    end
    if dropline then
      oix = oix-1
    else do
      outline.oix=left(outline.oix,length(outline.oix)-1)
      outline.oix = outline.oix||'::::'
      If dotrace then say 'Trace output ==> 'outline.oix
    end /*Test for dropping line */
  end /* Input-line with data     */
end /* Do-loop for input lines    */
outline.0 = oix

/* Write lines to &userid.HOSTS.LOCAL and run MAKESITE               */

"EXECIO * DISKW HOSTLOC (STEM OUTLINE. FINIS"
address TSO "MAKESITE HLQ="tcpiphlq

exit(0)
```

Appendix F. Command Examples

SNMP and OE SNMP

Following are examples of the commands:

Table 14. SNMP and OSNMP Command Formats

The V3R2 syntax for the SNMP command issued from NetView	The OS/390 TCP/IP OpenEdition format for the osnmp command issued from OE shell
SNMP Get hostname community_name variable_name	osnmp -h hostname -c community_name get variable_name
SNMP GETNext hostname community_name variable_name	osnmp -h hostname -c community_name getnext variable_name
SNMP Set hostname community_name variable_name	osnmp -h hostname -c community_name set variable_name value
(No equivalent)	osnmp -h hostname -c community_name getbulk variable_name
(No equivalent)	osnmp -h hostname -c community_name walk variable_name
(No equivalent)	osnmp -h hostname -c community_name bulkwalk variable_name
SNMP TRAPSon netmask net_desired	osnmp trap
SNMP TRAPSOff filter_id	(No equivalent; use OE shell kill command)
SNMP MIBvname asn.1_name	(No equivalent)
SNMP PING hostname	osnmp -h hostname -c community_name get ibmMvsRPingResponseTime.packet_size.time_out.host_addr

Notes:

1. The osnmp command supports additional parameters that are not supported by the V3R2 commands. The additional parameters are described in *OS/390 TCP/IP OpenEdition User's Guide*.
2. The Remote Ping (RPing) function is documented in the *OS/390 TCP/IP OpenEdition User's Guide*.
3. You can use the TCP/IP V3R2 SNMP command from NetView to query the OS/390 TCP/IP OpenEdition SNMP agent.
4. You can use the OS/390 TCP/IP OpenEdition osnmp command to query the TCP/IP V3R2 agents.

Traceroute and OE Traceroute

The OE version of traceroute has more functions than the TSO version. The syntax of the TSO version can be mapped to the new version as follows:

Table 15. Traceroute

TSO Version	OE Version
MAX	-m
TRY	-q
PORT	-p
DEBUG	-d
WAIT	-w

In addition, the OE version supports the following options:

- -t = type-of-service (tos)
- -s = source IP Address (srcAddr)
- -n = print hop addresses numerically
- -r = bypass the normal routing tables
- -v = verbose output
- -l = print value of the ttl field
- packetSize specification

Appendix G. OS/390 TCP/IP OpenEdition Samples

This appendix contains samples used in OS/390 TCP/IP OpenEdition regression testing.

Scenario A Samples

This test system configuration consists of one OS/390 TCP/IP OpenEdition executing with TCP/IP V3R2. The Integrated Sockets AF_INET PFS is used for OS/390 TCP/IP OpenEdition; Common INET is not used in this configuration. The configuration is similar to Scenario A that was described briefly in Chapter 3 and is illustrated in Figure 18 on page 80.

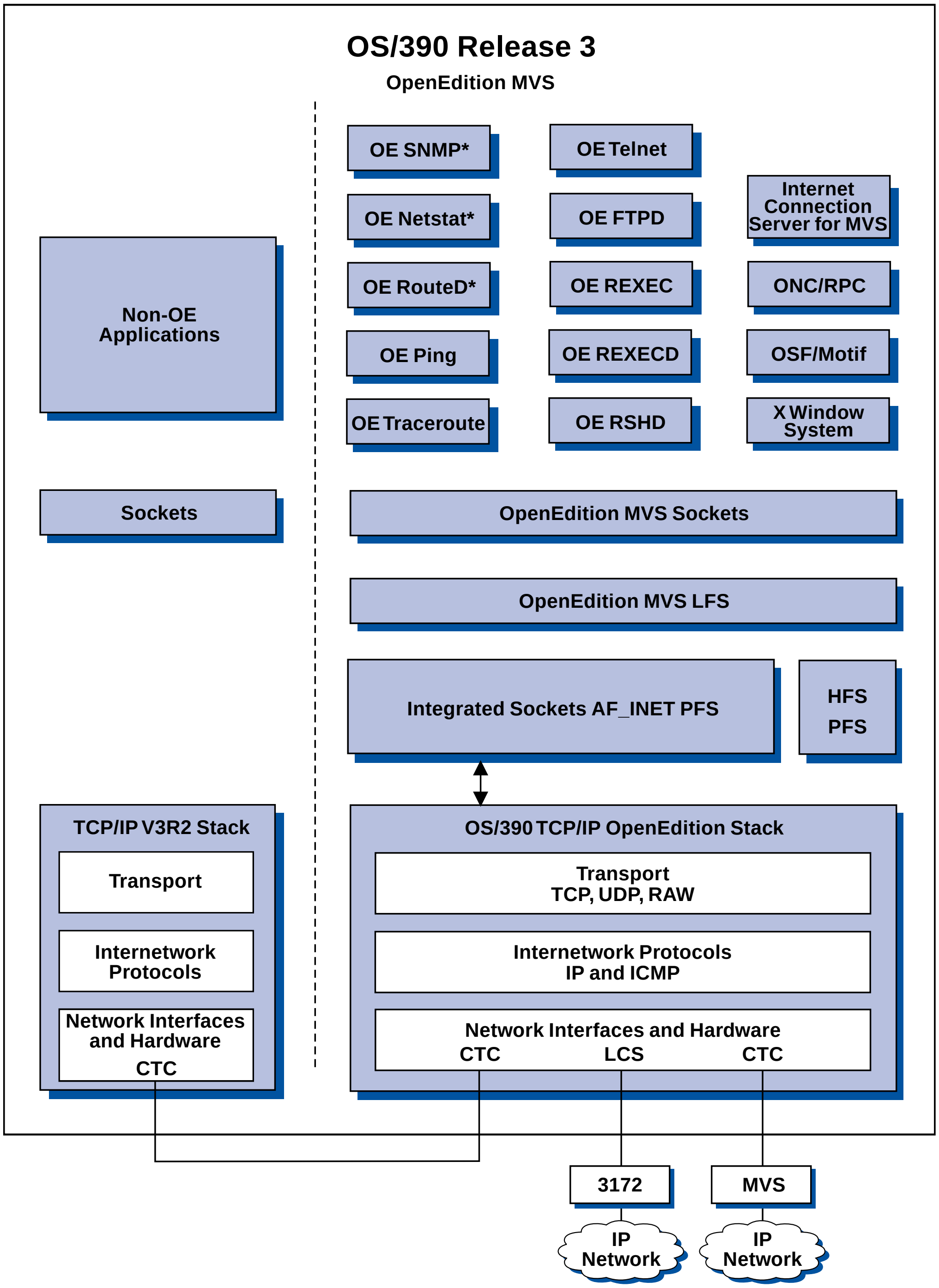


Figure 18. OS/390 TCP/IP OpenEdition Stack Selection by Integrated Sockets AF_INET and TCP/IP V3R2 Stack Selection by a Different PFS.

The connectivity of the OS/390 TCP/IP OpenEdition stack is illustrated in Figure 19 on page 81. This representation is based on the profile for TCPV33.

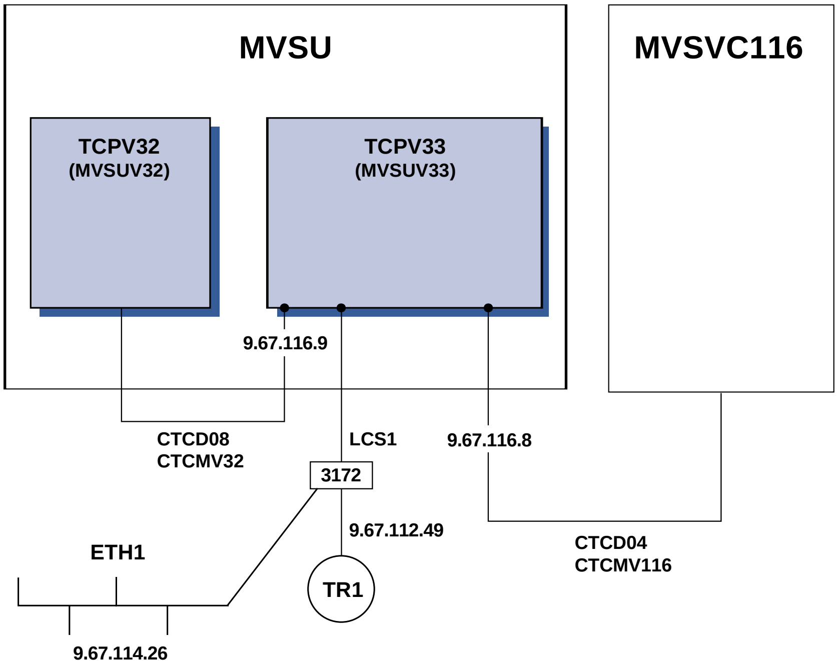


Figure 19. Connectivity of the OS/390 TCP/IP OpenEdition Stack in the Scenario A Sample

These samples provide a working configuration for OS/390 TCP/IP OpenEdition and TCP/IP V3R2, based on TCP/IP OpenEdition regression test. They are provided as samples only and require review and modification prior to use in other environments.

TCP/IP Modifications to SYS1.PARMLIB

The next four samples show the changes made to SYS1.PARMLIB members BPXPRMxx, SCHEDxx, LNKLSTxx, and LPALSTxx for OS/390 TCP/IP OpenEdition (shown in the examples as TCP/IP V3R3) and TCP/IP V3R2. The fifth SYS1.PARMLIB sample is for member CTIEZBxx which defines TCP/IP trace options.

BPXPRMxx

```
/********************************************************************/
/*                                                                  */
/*   This is a sample of the TCP/IP MVS V3R3 entries for            */
/*   the  BPXPRMxx member of SYS1.PARMLIB.                          */
/*                                                                  */
/********************************************************************/
/*                                                                  */
/*   Use the mount statement to define the file system required.    */
/*                                                                  */
/********************************************************************/
  MOUNT FILESYSTEM('OEA.TCPV33.HFS')
        TYPE(HFS)
        MODE(READ)
        MOUNTPOINT('/usr/lpp/tcpip')
/********************************************************************/
/*                                                                  */
/*   Use the Network statement to define the domain the specified   */
/*   file system supports.                                          */
/*                                                                  */
/********************************************************************/
  FILESYSTYPE TYPE(INET) ENTRYPOINT(EZBPFINI)
  SUBFILESYSTYPE NAME(TCPV33) TYPE(INET) ENTRYPOINT(EZBPFINI)
  NETWORK DOMAINNAME(AF_INET)
          DOMAINNUMBER(2)
          MAXSOCKETS(5000)
          TYPE(INET)
```

SCHEDxx

```
 /*                                                                  */
 /********************************************************************/
 /*                PROGRAM PROPERTIES TABLE ADDITIONS                */
 /********************************************************************/
 /*                                                                  */
 /*                  TCPIP V3R3 Program                              */
 /*                                                                  */
PPT PGMNAME(EZBTCPIP)      /* PROGRAM NAME that loads EZBTIINI       */
    NOCANCEL               /* NON-CANCELABLE                         */
    KEY(6)                 /* KEY 6                                  */
    NOSWAP                 /* NON-SWAPPABLE                          */
    PRIV                   /* PRIVILEGED                             */
    SYST                   /* SYSTEM TASK, NOT TIMED                 */
    SPREF                  /* ALL PRIVATE AREA SHORT-TERM FIXED      */
    LPREF                  /* ALL PRIVATE AREA LONG-TERM FIXED       */
```

LNKLSTxx

```
   . . . .
   TCPV33.SEZALINK,
   TCPV32.SEZALINK,
   TCPV32.SEZALNK2,
   TCPV32.SEZAMIG,
   . . . .
```

LPALSTxx

```
   . . . .
   TCPDS32L.SEZALPA,
   . . . .
```

CTIEZBxx

```
/********************************************************************/
/*                                                                  */
/*  TCP/IP for MVS                                                  */
/*  SMP/E Distribution Name: CTIEZB00                               */
/*                                                                  */
/*  MEMBER:  CTIEZB00                                               */
/*                                                                  */
/*     COPYRIGHT = Licensed Materials - Program Property of IBM.    */
/*               This product contains "Restricted Materials of IBM"*/
/*               5645-001 5655-HAL (C) Copyright IBM Corp. 1996.   */
/*               All rights reserved.                               */
/*               US Government Users Restricted Rights -            */
/*               Use, duplication or disclosure restricted          */
/*               by GSA ADP Schedule Contract with IBM Corp.        */
/*               See IBM Copyright Instructions                     */
/*                                                                  */
/*  DESCRIPTION = This parmlib member causes component trace for    */
/*                the TCP/IP product to be initialized with a       */
/*                trace buffer size of 512K                         */
/*                                                                  */
/*                This parmlib members only lists those TRACEOPTS   */
/*                value specific to TCP/IP.  For a complete list    */
/*                of TRACEOPTS keywords and their values see        */
/*                OS/390 MVS INITIALIZATION AND TUNING REFERENCE.   */
/*                                                                  */
/* $MAC(CTIEZB00) PROD(TCPIP): Component Trace SYS1.PARMLIB member  */
/*                                                                  */
/*  CHANGE-ACTIVITY =                                               */
/*  CFD List:                                                       */
/*                                                                  */
/*  $L0=D005509  HTCP330 960619 MWS: RAS DCR - Trace Enhancements   */
/*                                                                  */
/*  End CFD List:                                                   */
/********************************************************************/
 TRACEOPTS
          WTRSTART(TCPWTR1)
/* ---------------------------------------------------------------- */
/*   ON OR OFF: PICK 1                                              */
/* ---------------------------------------------------------------- */
          ON
/*        OFF                                                       */
          WTR(TCPWTR1)
/* ---------------------------------------------------------------- */
/*   BUFSIZE: A VALUE IN RANGE 128K TO 16M                          */
/* ---------------------------------------------------------------- */
          BUFSIZE(512K)
/* ---------------------------------------------------------------- */
/*   OPTIONS: NAMES OF FUNCTIONS TO BE TRACED, OR "ALL"             */
/* ---------------------------------------------------------------- */
          OPTIONS(
                  'ALL     '
/*               ,'MINIMUM '   */
/*               ,'INIT    '   */
/*               ,'SOCKET  '   */
/*               ,'OPMSGS  '   */
/*               ,'PFS     '   */
/*               ,'ENGINE  '   */
/*               ,'RAW     '   */
/*               ,'UDP     '   */
/*               ,'TCP     '   */
/*               ,'ICMP    '   */
/*               ,'ARP     '   */
/*               ,'CTC     '   */
/*               ,'CLAW    '   */
/*               ,'LCS     '   */
/*               ,'INTERNET'   */
/*               ,'QUEUE   '   */
/*               ,'MESSAGE '   */
/*               ,'IOCTL   '   */
/*               ,'VTAMDATA'   */
/*               ,'VTAM    '   */
                  )
```

OS/390 TCP/IP OpenEdition Initialization

The following samples are related to initializing OS/390 TCP/IP OpenEdition. These include the procedure for starting OS/390 TCP/IP OpenEdition (TCP/IP V3R3), the profile used in our regression test configuration, and the TCPDATA file.

Startup Procedure

The following procedure must reside in a procedure library such as SYS1.PROCLIB.

```
//TCPV33 PROC MODULE=EZBTCPIP,
//            PARMS='CTRACE(CTIEZBxx)',
//            TCPPRF=MVSU,
//            TCPDAT=TCPDATA,
//            TCPPDS='SYS1.TCPPARMS',
//            REG=7M
//*
//* OS/390 TCP/IP OpenEdition
//* SMP/E Distribution Name: EZBOPROC
//*
//*     5645-001 5655-HAL (C) Copyright IBM Corp. 1989, 1997.
//*     All rights reserved.
//*     US Government Users Restricted Rights -
//*     Use, duplication or disclosure restricted
//*     by GSA ADP Schedule Contract with IBM Corp.
//*     See IBM Copyright Instructions
//*
//TCPIP    EXEC PGM=&MODULE,
//             PARM='&PARMS',
//             REGION=&REG,
//             TIME=1440
//*
//*     The C runtime libraries should be in the system's link list
//*     or add them to the STEPLIB definition here.  If you add
//*     them to STEPLIB, they must be APF authorized.
//*
//*
//*     SYSPRINT contains run-time diagnostics from TCPIP. It may be
//*        a data set or SYSOUT.
//*     SYSERROR contains error messages from TCPIP that occurred
//*        while processing the PROFILE.
//*
//SYSPRINT DD SYSOUT=*,DCB=(RECFM=FB,LRECL=137,BLKSIZE=137)
//SYSOUT   DD SYSOUT=*,DCB=(RECFM=FB,LRECL=137,BLKSIZE=137)
//CEEDUMP  DD SYSOUT=*,DCB=(RECFM=FB,LRECL=137,BLKSIZE=137)
//SYSERROR DD SYSOUT=*
//*
//*     TCPIP reads the parameters from a data set with name
//*     TCPIP.nodename.TCPIP or with name TCPIP.PROFILE.TCPIP.
//*     See the chapter on "Configuring the TCPIP Address Space" in
//*     the Configuration Guide for more information. A sample of
//*     such a profile is included in member SAMOPROF of the
//*     SEZAINST data set.
//*
//PROFILE  DD DISP=SHR,DSN=&TCPPDS(&TCPPRF)
//*
//*     SYSTCPD explicitly identifies which data set is to be
//*        used to obtain the parameters defined by TCPIP.DATA.
//*        The SYSTCPD DD statement should be placed in the TSO logon
//*        procedure or in the JCL of any client or server executed
//*        as a background task.  The data set can be any sequential
//*        data set or a member of a partitioned data set (PDS).
//*
//*     For more information please see "Understanding TCP/IP Data
//*     Set Names" in the Configuration Guide.
//*
//SYSTCPD  DD DSN=&TCPPDS(&TCPDAT),DISP=SHR
```

Profile

The following profile resides in SYS1.TCPPARMS and is used at startup by the TCPV33 procedure.

```
;
DATASETPREFIX TCPV33
;
KEEPALIVEOPTIONS
INTERVAL 1
ENDKEEPALIVEOPTIONS
;
;SACONFIG DISABLE
;
; ------------------------------------------------------------------
; Hardware definitions:
; ------------------------------------------------------------------
;
; ------------------------------------------------------------------
; Channel to Channel Definition
; HOSTNAME:  MVSU-MVSUV32       (TCPIP V3R2 stack)
; ------------------------------------------------------------------
;
DEVICE CTCMV32   CTC    D08
LINK   CTCD08    CTC  0 CTCMV32
;
;
; ------------------------------------------------------------------
; Channel to Channel Definition
; HOSTNAME:  MVSU-MVSVC116
; ------------------------------------------------------------------
;
DEVICE CTCMV116  CTC    D04
LINK   CTCD04    CTC  0 CTCMV116
;
; ------------------------------------------------------------------
; UNITY__  4M TokenRing
;             Ethernet
; HOSTNAME:  MVSU
; ------------------------------------------------------------------
;
DEVICE LCS1      LCS      114
LINK   ETH1      ETHERNET   0 LCS1
LINK   TR1       IBMTR      0 LCS1
;
; ------------------------------------------------------------------
; ------------------------------------------------------------------
;
; ------------------------------------------------------------------
; Ports reserved for OMVS
; ------------------------------------------------------------------
;
PORT
     7 TCP OMVS                  ; OMVS ECHO
     7 UDP OMVS                  ; OMVS ECHO
     9 TCP OMVS                  ; OMVS Discard
     9 UDP OMVS                  ; OMVS Discard
    13 TCP OMVS                  ; OMVS Daytime
    13 UDP OMVS                  ; OMVS Daytime
    19 TCP OMVS                  ; OMVS Chargen
    19 UDP OMVS                  ; OMVS Chargen
    53 TCP OMVS                  ; Name Server
    53 UDP OMVS                  ; Name Server
;  111 UDP OMVS                  ; Portmap Server
;  111 TCP OMVS                  ; Portmap Server
   161 UDP OSNMPD                ; SNMPD Agent
   162 UDP OSNMPD                ; SNMPD Client for receipt of traps
   512 UDP OMVS                  ; Remote Execution Server
   513 UDP OMVS                  ; Remote Login
   514 TCP OMVS                  ; Remote Shell Server
   514 UDP OMVS                  ; Remote Shell Server
   620 TCP OMVS                  ; Ftpd
   621 TCP OMVS                  ; Ftpd
   623 TCP OMVS                  ; Telnetd
;
; Local host's Internet addresses
;
HOME
  9.67.116.9     CTCD08
  9.67.116.8     CTCD04
  9.67.114.26    ETH1
  9.67.112.49    TR1
;
PRIMARYINTERFACE TR1
;
GATEWAY
; Network  First hop   Driver  Packet size  Subnet mask    Subnet value
; Direct routes
9.67.116.7    =        CTCD04     4096      HOST
9.67.116.10   =        CTCD08     4096      HOST
9             =        ETH1       1500      0.255.255.128  0.67.114.0
9             =        TR1        2000      0.255.255.128  0.67.112.0
;
DEFAULTNET 9.67.112.1   TR1    DEFAULTSIZE    0
;
START CTCMV116
START CTCMV32
START LCS1
```

TCPDATA

The following TCPDATA file resides in “SYS1.TCPPARMS” and is used at startup by the TCPV33 procedure.

```
;***********************************************************************
;                                                                      *
;   Name of File:           TCPIP.DATA                                 *
;                                                                      *
;   This data set, TCPIP.DATA, is used to specify configuration        *
;   information required by TCP/IP client programs.                    *
;                                                                      *
;   Syntax Rules for the TCPIP.DATA configuration data set:            *
;                                                                      *
;   (a) All characters to the right of and including a ';' will be     *
;       treated as a comment.                                          *
;                                                                      *
;   (b) Blanks and <end-of-line> are used to delimit tokens.           *
;                                                                      *
;   (c) The format for each configuration statement is:                *
;                                                                      *
;       <SystemName||':'>  keyword  value                              *
;                                                                      *
;       where <SystemName||':'> is an optional label which may be      *
;       specified before a keyword; if present, then the keyword-      *
;       value pair will only be recognized if the SystemName matches   *
;       the node name of the system, as defined in the IEFSSNxx        *
;       PARMLIB member.  This optional label permits configuration     *
;       information for multiple systems to be specified in a single   *
;       TCPIP.DATA data set.                                           *
;                                                                      *
;***********************************************************************
; TCPIPuserid specifies the userid of the TCPIP address space.
; TCPIP is the default userid.
DATASETPREFIX TCPV33
;
TCPIPjobname TCPV33
;
; HostName specifies the TCP host name of this system.  If not
; specified, the default HostName will be the node name specified
; in the IEFSSNxx PARMLIB member.
;
; For example, if this TCPIP.DATA data set is shared between two
; systems, MVSXA and MVSESA, then the following two lines will define
; the HostName correctly on each system.
;
MVSU: HostName MVSU
; MVSESA: HostName mvsesa
;
; DomainOrigin specifies the domain origin that will be appended
; to host names passed to the resolver.  If a host name contains
; any dots, then the DomainOrigin will not be appended to the
; host name.
;
DomainOrigin TCP.RALEIGH.IBM.COM
;
; NSinterAddr specifies the internet address of the Name Server.
; Multiple Name Server addresses may be specified.  The Name Servers
; will be tried in the given order.
; If a Name Server will not be used, then do not code an NSinterAddr
; statement.  This will cause all names to be resolved via host table
; lookup. 14.0.0.0 defines a name server located on the local system.
;
; The format of the NSinterAddr specification is as follows.
; If a remote secondary name server is to be defined, the
; NSInterAddr comment line may be updated with the address
; of the host that is running a Name Server.
;
NSinterAddr  9.37.32.94
NSinterAddr  9.37.32.3
NSinterAddr  9.67.1.5
;
; NSportAddr specifies the Name Server port.
; 53 is the default value.
;
NSportAddr 53
;
; ResolveVia specifies how the Resolver is to communicate
; with the Name Server. TCP indicates use of TCP virtual circuits.
; UDP indicates use of UDP Datagrams.
; The default is UDP.
;
ResolveVia UDP
;
; ResolverTimeout specifies the time in seconds that the Resolver
; will wait while trying to open a TCP connection to the name server,
; or how long it will wait for a response when using UDP.
; Default is 30 seconds.
;
ResolverTimeout 30
;
; ResolverUdpRetries specifies the number of times the resolver should
; retry a query to the Nameserver when using UDP datagrams.
; Default is 1.
;
ResolverUdpRetries 1
;
; End of file.
```

OS/390 TCP/IP OpenEdition Application Procedures

The following samples are related to initializing OS/390 TCP/IP OpenEdition applications.

File Transfer Protocol (FTP)

FTP Procedure:

```
//FTPD   PROC MODULE='FTPD',PARMS='PORT 621 TRACE'
//*********************************************************************
//*                                                                   *
//*         TCP/IP FOR MVS                                            *
//*                                                                   *
//*         DESCRIPTIVE NAME:            FTP SERVER START PROCEDURE   *
//*                                                                   *
//*         FILE NAME:                   TCPIP.SEZAINST(EZAFTPAP)     *
//*                                      TCPIP.SEZAINST(FTPD)         *
//*                                                                   *
//*         SMP/E DISTRIBUTION NAME:     EZAFTPAP                     *
//*                                                                   *
//*                                                                   *
//*             LICENSED MATERIALS - PROPERTY OF IBM                  *
//*             THIS PRODUCT CONTAINS "RESTRICTED MATERIALS OF IBM"   *
//*             5645-001 5655-HAL (C) COPYRIGHT IBM CORP. 1995, 1997. *
//*             ALL RIGHTS RESERVED.                                  *
//*             US GOVERNMENT USERS RESTRICTED RIGHTS -               *
//*             5645-001 5655-HAL (C) COPYRIGHT IBM CORP. 1995, 1997. *
//*             ALL RIGHTS RESERVED.                                  *
//*             US GOVERNMENT USERS RESTRICTED RIGHTS -               *
//*             USE, DUPLICATION OR DISCLOSURE RESTRICTED BY          *
//*             GSA ADP SCHEDULE CONTRACT WITH IBM CORP.              *
//*             SEE IBM COPYRIGHT INSTRUCTIONS.                       *
//*                                                                   *
//*                                                                   *
//*********************************************************************
//FTPD   EXEC PGM=&MODULE,REGION=4096K,TIME=NOLIMIT,
//      PARM='POSIX(ON) ALL31(ON)/&PARMS'
//CEEDUMP  DD SYSOUT=*
//*
//*       SYSFTPD IS USED TO SPECIFY THE FTP.DATA FILE FOR THE FTP
//*       SERVER.  THE FILE CAN BE ANY SEQUENTIAL DATA SET, MEMBER
//*       OF A PARTITIONED DATA SET (PDS), OR HFS FILE.
//*
//*       THE SYSFTPD DD STATEMENT IS OPTIONAL.  THE SEARCH ORDER FOR
//*       FTP.DATA IS:
//*
//*           /ETC/FTP.DATA
//*           SYSFTPD DD STATEMENT
//*           JOBNAME.FTP.DATA
//*           SYS1.TCPPARMS(FTPDATA)
//*           TCPIP.FTP.DATA
//*
//*       IF NO FTP.DATA FILE IS FOUND, FTP DEFAULT VALUES ARE USED.
//*       FOR INFORMATION ON FTP DEFAULTS, SEE THE CUSTOMIZATION
//*       AND ADMINISTRATION GUIDE AND TCP/IP OE MVS APPLICATIONS
//*       FEATURE GUIDE.
//*SYSFTPD DD DISP=SHR,DSN=TCPIP.SEZAINST(FTPSDATA)
//*
//*      SYSTCPD EXPLICITLY IDENTIFIES WHICH FILE IS TO BE
//*      USED TO OBTAIN THE PARAMETERS DEFINED BY TCPIP.DATA.
//*      THE SYSTCPD DD STATEMENT SHOULD BE PLACED IN THE JCL OF
//*      THE SERVER.  THE FILE CAN BE ANY SEQUENTIAL DATA SET,
//*      MEMBER OF A PARTITIONED DATA SET (PDS), OR HFS FILE.
//*SYSTCPD DD DISP=SHR,DSN=TCPIP.SEZAINST(TCPDATA)
//*
//*
```

```
//*      SYSFTSX EXPLICITLY IDENTIFIES WHICH FILE IS TO BE USED
//*      FOR THE EBCDIC-ASCII TRANSLATION TABLE.  THE FILE CAN
//*      BE ANY SEQUENTIAL DATA SET, MEMBER OF A PARTITIONED DATA
//*      SET (PDS), OR HFS FILE.
//*SYSFTSX DD DISP=SHR,DSN=TCPIP.STANDARD.TCPXLBIN
```

INETD Procedure

INETD is required for the execution of OE TELNET, Remote Execution (REXEC), and Remote Shell (RSH). The following procedure was used on the TCP/IP test systems for INETD.

```
//INETD  PROC
//*********************************************************************
//*        Sample procedure to execute INETD
//*********************************************************************
//*
//*        5655-HAL (C) Copyright IBM Corp. 1996, 1997.
//*        All rights reserved.
//*        US Government Users Restricted Rights -
//*        Use, duplication or disclosure restricted
//*        by GSA ADP Schedule Contract with IBM Corp.
//*        See IBM Copyright Instructions
//*
//*********************************************************************
//INETD EXEC PGM=INETD,REGION=0K,TIME=NOLIMIT,
//      PARM='POSIX(ON) ALL31(ON)/ /etc/inetd.conf'
//SYSPRINT DD SYSOUT=*
```

OSNMPD Procedure

The following procedure was used on the TCP/IP test systems for OSNMPD.

```
//OSNMPD    PROC
//*
//* Sample procedure for running the OE SNMP agent
//*
//* TCP/IP for MVS
//* SMP/E Distribution Name: AEZASMP1
//*
//*        Licensed Materials - Property of IBM
//*        This product contains "Restricted Materials of IBM"
//*        5645-001 5655-HAL (C) Copyright IBM Corp. 1996.
//*        All rights reserved.
//*        US Government Users Restricted Rights -
//*        Use, duplication or disclosure restricted by
//*        GSA ADP Schedule Contract with IBM Corp.
//*        See IBM Copyright Instructions.
//*
//OSNMPD EXEC PGM=EZASNMPD,REGION=4096K,TIME=NOLIMIT,
// PARM='POSIX(ON) ALL31(ON)/ -c public '
//*
//*   The C runtime libraries should be in the system's link list
//*   or this sample procedure will need to STEPLIB to them.
//*
//*   TCP/IP runtime libraries should also be in the system's link
//*   list.
//*
//*   OSNMPD must find the name (TCPIPJOBNAME in TCPIP.DATA) that
//*   it should be associated with. The OE function __iptcpn() is
//*   used to find this name. It is suggested that the parmlist
//*   be modified to set the environment variable
//*   RESOLVER_CONFIG to point to the correct resolver file when
//*   multiple INET Physical File Systems are started:
//*
//*   //      PARM=('POSIX(ON) ALL31(ON)',
//*   //      'ENVAR("RESOLVER_CONFIG=/etc/tcpv33a.data")/')
//*
//*   If only one INET PFS will be started then it is
//*   recommended that /etc/resolv.conf be used.
//*
//SYSPRINT DD SYSOUT=*,DCB=(RECFM=F,LRECL=80,BLKSIZE=80)
//SYSIN    DD DUMMY
//SYSMDUMP DD SYSOUT=*
//SYSERR   DD SYSOUT=*
//SYSOUT   DD SYSOUT=*,DCB=(RECFM=F,LRECL=80,BLKSIZE=80)
//CEEDUMP  DD SYSOUT=*
```

PORTMAP Procedure

PORTMAP is required for the execution of RPCINFO and RPCGEN. The following procedure was used on the TCP/IP test systems for PORTMAP.

```
//PORTMAP  PROC
//*
//*   TCP/IP for MVS
//*   SMP/E distribution name: EZBOPORT
//*
//*   5645-001 5655-HAL (C) Copyright IBM Corp. 1997.
//*   Licensed Materials - Property of IBM
//*   This product contains "Restricted Materials of IBM"
//*   All rights reserved.
//*   US Government Users Restricted Rights -
//*   Use, duplication or disclosure restricted by
//*   GSA ADP Schedule Contract with IBM Corp.
//*   See IBM Copyright Instructions.
//*
//*   Function: OpenEdition MVS Portmapper Server main process
//*
//PORTMAP  EXEC PGM=OPORTMAP,REGION=4096K,TIME=1440,
//           PARM='POSIX(ON),ALL31(ON)/'
//*
//STDOUT   DD SYSOUT=*
//STDERR   DD SYSOUT=*
//SYSOUT   DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//         PEND
```

SYSLOGD Procedure

SYSLOGD is required for the recording of TCP/IP V3R3 OpenEdition application events. The following procedure was used on the TCP/IP test systems for SYSLOGD.

```
//SYSLOGD PROC OPTS=''
//*********************************************************************
//*        SAMPLE PROCEDURE TO EXECUTE SYSLOGD
//*********************************************************************
//*
//*        5655-HAL (C) COPYRIGHT IBM CORP. 1996, 1997.
//*        ALL RIGHTS RESERVED.
//*        US GOVERNMENT USERS RESTRICTED RIGHTS -
//*        USE, DUPLICATION OR DISCLOSURE RESTRICTED
//*        BY GSA ADP SCHEDULE CONTRACT WITH IBM CORP.
//*        SEE IBM COPYRIGHT INSTRUCTIONS
//*
//*********************************************************************
//SYSLOGD EXEC PGM=SYSLOGD,REGION=0K,TIME=NOLIMIT,PARM=&OPTS
//SYSPRINT DD SYSOUT=*
```

OS/390 TCP/IP OpenEdition HFS Files

The following samples show the OS/390 TCP/IP OpenEdition HFS files used in the test system configuration.

Services

The following file resides in the HFS /etc/services.

```
#
#     /etc/services file
#
echo            7/tcp
echo            7/udp
discard         9/tcp           sink null
discard         9/udp           sink null
systat          11/tcp          users
daytime         13/tcp
daytime         13/udp
netstat         15/tcp
qotd            17/tcp          quote
chargen         19/tcp          ttytst source
chargen         19/udp          ttytst source
otelnet         623/tcp
smtp            25/tcp          mail
time            37/tcp          timserver
time            37/udp          timserver
rlp             39/udp          resource        # resource location
nameserver      42/tcp          name            # IEN 116
whois           43/tcp          nicname
domain          53/tcp          nameserver      # name-domain server
domain          53/udp          nameserver
mtp             57/tcp                          # deprecated
tftp            69/udp
rje             77/tcp          netrjs
finger          79/tcp
link            87/tcp          ttylink
supdup          95/tcp
hostnames       101/tcp         hostname        # usually from sri-nic
pop             109/tcp         postoffice
sunrpc          111/tcp
sunrpc          111/udp
auth            113/tcp         authentication
sftp            115/tcp
uucp-path       117/tcp
nntp            119/tcp         readnews untp   # USENET News Transfer Protocol
snmp            161/udp                         # snmp request port
snmp-trap       162/udp                         # snmp monitor trap port
exec            512/tcp
biff            512/udp         comsat
login           513/tcp
who             513/udp         whod
shell           514/tcp         cmd             # no passwords used
shell           514/udp         cmd             # no passwords used
syslog          529/udp
printer         515/tcp         spooler         # line printer spooler
talk            517/udp
ntalk           518/udp
efs             520/tcp                         # for LucasFilm
route           520/udp         router routed
timed           525/udp         timeserver
tempo           526/tcp         newdate
courier         530/tcp         rpc
conference      531/tcp         chat
```

```
netnews         532/tcp         readnews
netwall         533/udp                         # -for emergency broadcasts
uucp            540/tcp         uucpd           # uucp daemon
remotefs        556/tcp         rfs_server rfs  # Brunhoff remote filesystem
ncprout         580/udp         ncproute
ftp             621/tcp                         # OE ftpd
otelnet         623/tcp                         # OE telnetd
ingreslock      1524/tcp
rvd-control     531/udp                         # rvd control port
filesrv         2001/tcp
console         2018/udp
venus.itc       2106/tcp
client          2030/tcp
vexec           712/tcp         vice-exec
vlogin          713/tcp         vice-login
vshell          714/tcp         vice-shell
venus.itc       2106/tcp
rauth2          2001/udp
rfilebulk               2002/udp
rfilesrv                2003/udp
ropcons         2115/udp
rupdsrv         2131/udp
rupdbulk                2132/udp
rupdsrv1                2133/udp
rupdbulk1               2134/udp
klogin          543/tcp           # Kerberos aythenticated rlogin
kerberos        750/udp      kdc  # Kerberos aythentication--udp
kerberos        750/tcp      kdc  # Kerberos aythentication--tcp
kerboros_master 751/udp           # Kerberos aythentication
kerberos_master 751/tcp           # Kerberos aythentication
passwd_server   752/udp           # Kerberos passwd server
userreg_server  753/tcp           # Kerberos userreg serrver
kpop            1109/tcp          # Pop with Kerberos
knetd         2053/tcp                # Kerberos de-multiplexor
kshell          544/tcp         cmd     # and remote shell
eklogin         2105/tcp                # Kerberos encrypted rlogin
krb_prop        754/tcp                 # Kerboros slpave propagation
erlogin         888/tcp                 # Login and environment passing
sample        906/tcp                 # Kerberos sample app server
sample                        906/udp                #for kerberos simple test
```

INETD.CONF

The following file resides in the HFS /etc/inetd.conf.

```
#
#          /etc/inetd.conf
#
#          Internet server configuration database
#
#======================================================================
# service | socket | protocol | wait/ | user | server  | server program
# name    | type   |          | nowait|      | program |   arguments
#======================================================================
#
otelnet  stream tcp nowait bpxroot /usr/sbin/otelnetd otelnetd -t -D all
login     stream tcp nowait BPXROOT /usr/sbin/rlogind rlogind -m
shell   stream  tcp  nowait  bpxroot /usr/sbin/orshd      orshd -dLV
exec    stream  tcp  nowait  bpxroot /usr/sbin/orexecd    orexecd -dLV
echo     stream tcp nowait root internal
discard  stream tcp nowait root internal
chargen  stream tcp nowait root internal
daytime  stream tcp nowait root internal
time     stream tcp nowait root internal
echo     dgram  udp wait   root internal
discard  dgram  udp wait   root internal
chargen  dgram  udp wait   root internal
daytime  dgram  udp wait   root internal
time     dgram  udp wait   root internal
```

HOSTS

The following file resides in the HFS /etc/hosts.

```
#
#  /etc/hosts file
#
# This file contains the hostnames and their address for hosts in the
# network.  This file is used to resolve a hostname into an Internet
# address.
#
# The format of this file is:
# Internet Address Hostname # Comments
#
# Internet   Hostname                      # Comments
# Address
127.0.0.1     localhost
9.67.112.49   MVSU mvsu
9.67.43.62    ralvm12
9.67.113.53   mvsa MVSA
9.67.116.7    mvsu-mvsc116
9.67.116.8    mvsc116-mvsu
9.67.116.9    mvsu-mvsuv32
9.67.116.10   mvsuv32-mvsu
9.67.114.26   mvsu-e
```

RESOLV.CONF

The following file resides in the HFS /etc/resolv.conf.

```
TCPIPjobname  TCPV33
Datasetprefix TCPV33
Messagecase mixed
HostName MVSU
Domain   tcp.raleigh.ibm.com
nameserver  9.37.32.94
nameserver  9.37.32.3
nameserver  9.67.5.44
```

SYSLOG.CONF

The following file resides in the HFS /etc/syslog.conf.

```
local1.debug   /tmp/telnetd.syslog
local2.debug   /tmp/rcmds.syslog
daemon.info    /tmp/daemon.log
daemon.debug   /tmp/daemon.trace
```

OSNMPD.DATA

The following file resides in the HFS /etc/osnmpd.data.

```
# This is a test osnmpd.data file to set MIB variables to non-default
#  values.
  sysDescr  "SNMP agent on MVSU"
  sysContact  "MVSU Operations   "
  sysLocation "RTP NC         "
  sysName "MVSU"
  sysObjectID "1.3.6.1.4.1.2.14"
  sysServices 3
  snmpEnableAuthenTraps  2
  dpiPathNameForUnixStream "/tmp/dpi_new_socket"
  saMaxTimeout 1000
  saAllowDuplicateIDs 2
```

Scenario B Samples

This section contains samples used in OS/390 TCP/IP OpenEdition regression testing for Common INET. The test system configuration consists of two OS/390 TCP/IP OpenEdition stacks executing with a TCP/IP V3R2 stack. The two OS/390 TCP/IP OpenEdition stacks are illustrated as TCPV33 and TCPV33A, which are the stack names used in the following examples. These samples are provided as samples only and require review and modification prior to use in other environments.

The following Scenario B samples are different from the Scenario A samples:

- BPXPRMxx in SYS1.PARMLIB
- RESOLV.CONF

The following samples were added to Scenario B for OE RouteD:

- OROUTED Procedure
- OROUTED.ENV
- GATEWAYS

The configuration is similar to Scenario B that was described briefly in Chapter 3 and is illustrated in Figure 20 on page 99.

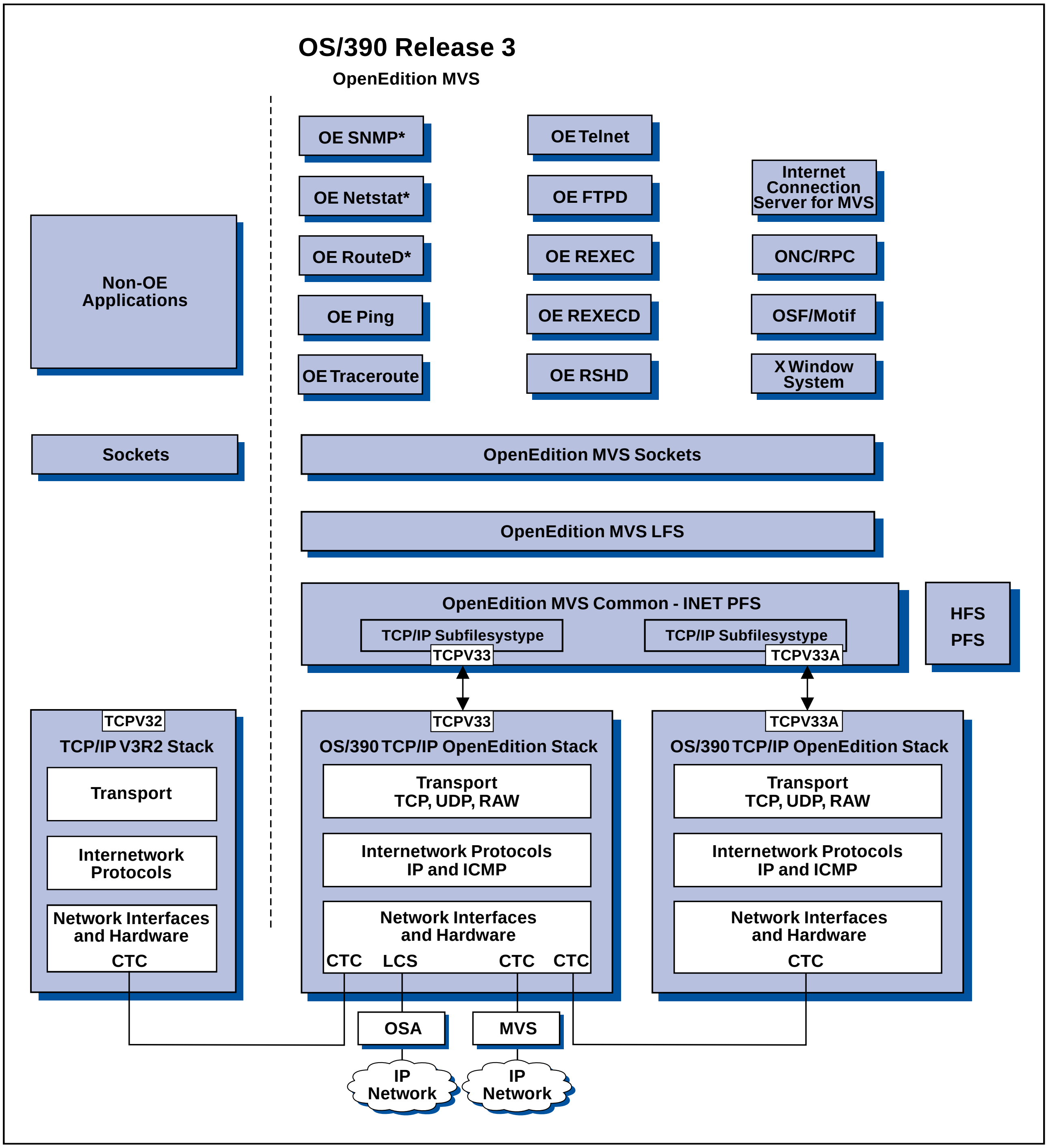


Figure 20. OS/390 TCP/IP OpenEdition Stack Selections by Common-INET and TCP/IP V3R2 Stack Selection by a Different PFS.

The connectivity to one OS/390 TCP/IP OpenEdition stack, TCPV33, is illustrated in Figure 21 on page 100. This representation is based on the profile for TCPV33.

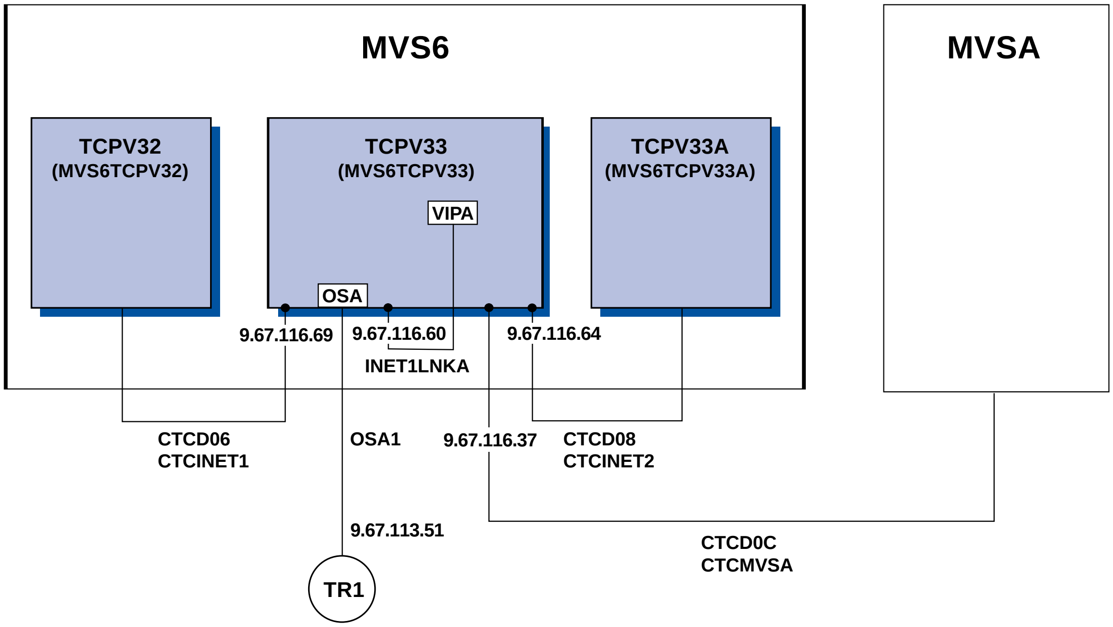


Figure 21. Connectivity of the OS/390 TCP/IP OpenEdition Stack in the Scenario B Sample

TCP/IP Modifications to SYS1.PARMLIB

The next four samples show the changes made to SYS1.PARMLIB members BPXPRMxx, SCHEDxx ,LNKLSTxx, and LPALSTxx for OS/390 TCP/IP OpenEdition (shown in the examples as TCP/IP V3R3) and TCP/IP V3R2. The fifth SYS1.PARMLIB sample is for member CTIEZBxx which defines TCP/IP trace options.

BPXPRMxx

This sample shows the changes made to SYS1.PARMLIB member BPXPRMxx for Common INET. Because there are two OS/390 TCP/IP OpenEdition stacks on the C-INET, TCPV33 and TCPV33A, the subfilesystype names are included.

```
/*******************************************************************/
/*                                                                 */
/*   This is a sample of the TCP/IP MVS V3R3 entries for           */
/*   the  BPXPRMxx member of SYS1.PARMLIB.                         */
/*                                                                 */
/*******************************************************************/
/*                                                                 */
/*   Use the mount statement to define the file system required.   */
/*                                                                 */
/*******************************************************************/
  MOUNT FILESYSTEM('OEA.TCPV33.HFS')
        TYPE(HFS)
        MODE(READ)
        MOUNTPOINT('/usr/lpp/tcpip')
/*******************************************************************/
/*                                                                 */
/*   Use the Network statement to define the domain the specified  */
/*   file system supports.                                         */
/*                                                                 */
/*******************************************************************/
  FILESYSTYPE TYPE(CINET) ENTRYPOINT(BPXTCINT)
  SUBFILESYSTYPE NAME(TCPV33) TYPE(CINET) ENTRYPOINT(EZBPFINI) DEFAULT
  SUBFILESYSTYPE NAME(TCPV33A) TYPE(CINET) ENTRYPOINT(EZBPFINI)
  NETWORK DOMAINNAME(AF_INET)
          DOMAINNUMBER(2)
          MAXSOCKETS(10000)
          INADDRANYPORT(10000)
          INADDRANYCOUNT(1000)
          TYPE(CINET)
```

SCHEDxx

```
 /*                                                                 */
 /*******************************************************************/
 /*                PROGRAM PROPERTIES TABLE ADDITIONS               */
 /*******************************************************************/
 /*                                                                 */
 /*                  TCPIP V3R3 Program                             */
 /*                                                                 */
PPT PGMNAME(EZBTCPIP)      /* PROGRAM NAME that loads EZBTIINI      */
    NOCANCEL               /* NON-CANCELABLE                        */
    KEY(6)                 /* KEY 6                                 */
    NOSWAP                 /* NON-SWAPPABLE                         */
    PRIV                   /* PRIVILEGED                            */
    SYST                   /* SYSTEM TASK, NOT TIMED                */
    SPREF                  /* ALL PRIVATE AREA SHORT-TERM FIXED     */
    LPREF                  /* ALL PRIVATE AREA LONG-TERM FIXED      */
```

LNKLSTxx

```
   . . . .
   TCPV33.SEZALINK,
   TCPV32.SEZALINK,
   TCPV32.SEZALNK2,
   TCPV32.SEZAMIG,
   . . . .
```

LPALSTxx

```
   . . . .
   TCPDS32L.SEZALPA,
   . . . .
```

CTIEZBxx

```
/********************************************************************/
/*                                                                  */
/*  TCP/IP for MVS                                                  */
/*  SMP/E Distribution Name: CTIEZB00                               */
/*                                                                  */
/*  MEMBER:  CTIEZB00                                               */
/*                                                                  */
/*     COPYRIGHT = Licensed Materials - Program Property of IBM.    */
/*               This product contains "Restricted Materials of IBM"*/
/*               5645-001 5655-HAL (C) Copyright IBM Corp. 1996.    */
/*               All rights reserved.                               */
/*               US Government Users Restricted Rights -            */
/*               Use, duplication or disclosure restricted          */
/*               by GSA ADP Schedule Contract with IBM Corp.        */
/*               See IBM Copyright Instructions                     */
/*                                                                  */
/*  DESCRIPTION = This parmlib member causes component trace for    */
/*                the TCP/IP product to be initialized with a       */
/*                trace buffer size of 512K                         */
/*                                                                  */
/*                This parmlib members only lists those TRACEOPTS   */
/*                value specific to TCP/IP.  For a complete list    */
/*                of TRACEOPTS keywords and their values see        */
/*                OS/390 MVS INITIALIZATION AND TUNING REFERENCE.   */
/*                                                                  */
/* $MAC(CTIEZB00) PROD(TCPIP): Component Trace SYS1.PARMLIB member  */
/*                                                                  */
/*  CHANGE-ACTIVITY =                                               */
/*  CFD List:                                                       */
/*                                                                  */
/*  $L0=D005509  HTCP330 960619 MWS: RAS DCR - Trace Enhancements   */
/*                                                                  */
/*  End CFD List:                                                   */
/********************************************************************/
 TRACEOPTS
          WTRSTART(TCPWTR1)
/* ---------------------------------------------------------------- */
/*   ON OR OFF: PICK 1                                              */
/* ---------------------------------------------------------------- */
          ON
/*        OFF                                                       */
          WTR(TCPWTR1)
/* ---------------------------------------------------------------- */
/*   BUFSIZE: A VALUE IN RANGE 128K TO 16M                          */
/* ---------------------------------------------------------------- */
          BUFSIZE(512K)
/* ---------------------------------------------------------------- */
/*   OPTIONS: NAMES OF FUNCTIONS TO BE TRACED, OR "ALL"             */
/* ---------------------------------------------------------------- */
          OPTIONS(
                  'ALL     '
/*               ,'MINIMUM '   */
/*               ,'INIT    '   */
/*               ,'SOCKET  '   */
/*               ,'OPMSGS  '   */
/*               ,'PFS     '   */
/*               ,'ENGINE  '   */
/*               ,'RAW     '   */
/*               ,'UDP     '   */
/*               ,'TCP     '   */
/*               ,'ICMP    '   */
/*               ,'ARP     '   */
/*               ,'CTC     '   */
/*               ,'CLAW    '   */
/*               ,'LCS     '   */
/*               ,'INTERNET'   */
/*               ,'QUEUE   '   */
/*               ,'MESSAGE '   */
/*               ,'IOCTL   '   */
/*               ,'VTAMDATA'   */
/*               ,'VTAM    '   */
                  )
```

OS/390 TCP/IP OpenEdition Initialization

The following samples are related to initializing OS/390 TCP/IP OpenEdition. These include the procedure for starting OS/390 TCP/IP OpenEdition (TCP/IP V3R3), the profile used in our regression test configuration, and the TCPDATA file.

Startup Procedure

The following procedure must reside in a procedure library such as SYS1.PROCLIB.

```
//TCPV33 PROC MODULE=EZBTCPIP,
//            PARMS='CTRACE(CTIEZBxx)',
//            TCPPRF=MVS6,
//            TCPDAT=TCPDATA,
//            TCPPDS='SYS1.TCPPARMS',
//            REG=7M
//*
//* OS/390 TCP/IP OpenEdition
//* SMP/E Distribution Name: EZBOPROC
//*
//*     5645-001 5655-HAL (C) Copyright IBM Corp. 1989, 1997.
//*     All rights reserved.
//*     US Government Users Restricted Rights -
//*     Use, duplication or disclosure restricted
//*     by GSA ADP Schedule Contract with IBM Corp.
//*     See IBM Copyright Instructions
//*
//TCPIP    EXEC PGM=&MODULE,
//             PARM='&PARMS',
//             REGION=&REG,
//             TIME=1440
//*
//*     The C runtime libraries should be in the system's link list
//*     or add them to the STEPLIB definition here.  If you add
//*     them to STEPLIB, they must be APF authorized.
//*
//*
//*     SYSPRINT contains run-time diagnostics from TCPIP. It may be
//*        a data set or SYSOUT.
//*     SYSERROR contains error messages from TCPIP that occurred
//*        while processing the PROFILE.
//*
//SYSPRINT DD SYSOUT=*,DCB=(RECFM=FB,LRECL=137,BLKSIZE=137)
//SYSOUT   DD SYSOUT=*,DCB=(RECFM=FB,LRECL=137,BLKSIZE=137)
//CEEDUMP  DD SYSOUT=*,DCB=(RECFM=FB,LRECL=137,BLKSIZE=137)
//SYSERROR DD SYSOUT=*
//*
//*     TCPIP reads the parameters from a data set with name
//*     TCPIP.nodename.TCPIP or with name TCPIP.PROFILE.TCPIP.
//*     See the chapter on "Configuring the TCPIP Address Space" in
//*     the Configuration Guide for more information. A sample of
//*     such a profile is included in member SAMOPROF of the
//*     SEZAINST data set.
//*
//PROFILE  DD DISP=SHR,DSN=&TCPPDS(&TCPPRF)
//*
//*     SYSTCPD explicitly identifies which data set is to be
//*        used to obtain the parameters defined by TCPIP.DATA.
//*        The SYSTCPD DD statement should be placed in the TSO logon
//*        procedure or in the JCL of any client or server executed
//*        as a background task.  The data set can be any sequential
//*        data set or a member of a partitioned data set (PDS).
//*
//*     For more information please see "Understanding TCP/IP Data
//*     Set Names" in the Configuration Guide.
//*
//SYSTCPD  DD DSN=&TCPPDS(&TCPDAT),DISP=SHR
```

Profile

The following profile resides in SYS1.TCPPARMS and is used at startup by the TCPV33 procedure.

```
;
; -------------------------------------------------------------------
; Profile for TCP/IP Version 3.3 Regression Test.
; -------------------------------------------------------------------
;
ITRACE ON CONFIG 1
; -------------------------------------------------------------------
;
; -------------------------------------------------------------------
; Specify a prefix that will be used instead of the default
; high-level qualifier for dynamic allocation of TCPIP datasets.
; -------------------------------------------------------------------
;
DATASETPREFIX TCPV33
;
; -------------------------------------------------------------------
; Hardware definitions
; -------------------------------------------------------------------
;
; -------------------------------------------------------------------
;  OSA Card  16M Tokenring
;  HOSTNAME:   MVS6
; -------------------------------------------------------------------
;
DEVICE OSA1  LCS    2E02
LINK   TR1   IBMTR     1 OSA1
;
; -------------------------------------------------------------------
;  Channel to Channel Definition
;  HOSTNAME:   MVS6TCPV33-MVSA
; -------------------------------------------------------------------
DEVICE CTCMVSA  CTC  D0C AUTORESTART
LINK   CTCD0C   CTC    1 CTCMVSA
;
; -------------------------------------------------------------------
;  Channel to Channel Definition
;  HOSTNAME:   MVS6TCPV33-MVS6TCPV32
; -------------------------------------------------------------------
;
DEVICE CTCINET1 CTC  0D06 IOBUFFERSIZE 32K
LINK   CTCD06   CTC    1 CTCINET1
;
; -------------------------------------------------------------------
;  Channel to Channel Definition
;  HOSTNAME:   MVS6TCPV33-MVS6TCPV32
; -------------------------------------------------------------------
;
DEVICE CTCINET1 CTC  0D06 IOBUFFERSIZE 32K
LINK   CTCD06   CTC    1 CTCINET1
;
; -------------------------------------------------------------------
;  Virtual Device and Link Definition
;  HOSTNAME:   MVS6TCPV33-VLINKA
; -------------------------------------------------------------------
```

```
;
DEVICE INET1VIPA  VIRTUAL 0
LINK   INET1VLNKA VIRTUAL 1 INET1VIPA
;
; -------------------------------------------------------------------
;  Channel to Channel Definition
;  HOSTNAME:   MVS6TCPV33-MVS6TCPV33A
; -------------------------------------------------------------------
;
DEVICE CTCINET2 CTC  D08
LINK   CTCD08   CTC    0 CTCINET2
;
; -------------------------------------------------------------------
; Local Host's Internet Addresses
; -------------------------------------------------------------------
;
HOME
9.67.113.51 TR1
9.67.116.37 CTCD0C
9.67.116.60 INET1VLNKA
9.67.116.69 CTCD06
9.67.116.64 CTCD08
;
; -------------------------------------------------------------------
;
PRIMARYINTERFACE TR1
;
; -------------------------------------------------------------------
;
GATEWAY
; Network  First hop   Driver  Packet size  Subnet mask     Subnet value
; Direct routes
;
9               =      TR1            2000  0.255.255.128   0.67.113.0
9.67.116.38     =      CTCD0C         8192  HOST
9.67.116.68     =      CTCD06         4096  HOST
;
DEFAULTNET 9.67.113.1  TR1            2000  0
;
; -------------------------------------------------------------------
;
BSDROUTINGPARMS false
;
;  LINK         maxmtu    metric    subnet mask   destination address
;
INET1VLNKA        2000         0    255.255.255.0 0
CTCD08            4096         0    255.255.255.0 9.67.116.67
;
ENDBSDROUTINGPARMS
;
UDPCONFIG NOUDPCHKSUM
;
KEEPALIVEOPTIONS
INTERVAL 5
ENDKEEPALIVEOPTIONS
;
; -------------------------------------------------------------------
; Started Devices
; --------------------------------------------------------------------
;
START CTCMVSA
START CTCINET1
START OSA1
```

TCPDATA

The following TCPDATA file resides in “SYS1.TCPPARMS” and is used at startup by the TCPV33 procedure.

```
;***********************************************************************
;                                                                      *
;   Name of File:           TCPIP.DATA                                 *
;                                                                      *
;   This data set, TCPIP.DATA, is used to specify configuration        *
;   information required by TCP/IP client programs.                    *
;                                                                      *
;   Syntax Rules for the TCPIP.DATA configuration data set:            *
;                                                                      *
;   (a) All characters to the right of and including a ';' will be     *
;       treated as a comment.                                          *
;                                                                      *
;   (b) Blanks and <end-of-line> are used to delimit tokens.           *
;                                                                      *
;   (c) The format for each configuration statement is:                *
;                                                                      *
;       <SystemName||':'>  keyword  value                              *
;                                                                      *
;       where <SystemName||':'> is an optional label which may be      *
;       specified before a keyword; if present, then the keyword-      *
;       value pair will only be recognized if the SystemName matches   *
;       the node name of the system, as defined in the IEFSSNxx        *
;       PARMLIB member.  This optional label permits configuration     *
;       information for multiple systems to be specified in a single   *
;       TCPIP.DATA data set.                                           *
;                                                                      *
;***********************************************************************
; TCPIPuserid specifies the userid of the TCPIP address space.
; TCPIP is the default userid.
DATASETPREFIX TCPV33
;
TCPIPjobname TCPV33
;
; HostName specifies the TCP host name of this system.  If not
; specified, the default HostName will be the node name specified
; in the IEFSSNxx PARMLIB member.
;
; For example, if this TCPIP.DATA data set is shared between two
; systems, MVSXA and MVSESA, then the following two lines will define
; the HostName correctly on each system.
;
MVS6: HostName MVS6
; MVSESA: HostName mvsesa
;
; DomainOrigin specifies the domain origin that will be appended
; to host names passed to the resolver.  If a host name contains
; any dots, then the DomainOrigin will not be appended to the
; host name.
;
DomainOrigin TCP.RALEIGH.IBM.COM
;
; NSinterAddr specifies the internet address of the Name Server.
; Multiple Name Server addresses may be specified.  The Name Servers
; will be tried in the given order.
; If a Name Server will not be used, then do not code an NSinterAddr
; statement.  This will cause all names to be resolved via host table
; lookup. 14.0.0.0 defines a name server located on the local system.
;
; The format of the NSinterAddr specification is as follows.
; If a remote secondary name server is to be defined, the
; NSInterAddr comment line may be updated with the address
; of the host that is running a Name Server.
;
NSinterAddr  9.37.32.94
NSinterAddr  9.37.32.3
NSinterAddr  9.67.1.5
;
; NSportAddr specifies the Name Server port.
; 53 is the default value.
;
NSportAddr 53
;
; ResolveVia specifies how the Resolver is to communicate
; with the Name Server. TCP indicates use of TCP virtual circuits.
; UDP indicates use of UDP Datagrams.
; The default is UDP.
;
ResolveVia UDP
;
; ResolverTimeout specifies the time in seconds that the Resolver
; will wait while trying to open a TCP connection to the name server,
; or how long it will wait for a response when using UDP.
; Default is 30 seconds.
;
ResolverTimeout 30
;
; ResolverUdpRetries specifies the number of times the resolver should
; retry a query to the Nameserver when using UDP datagrams.
; Default is 1.
;
ResolverUdpRetries 1
;
; End of file.
```

OS/390 TCP/IP OpenEdition Application Procedures

The following sample is related to initializing OS/390 TCP/IP OpenEdition application OROUTED.

OROUTED Procedure:

```
//*
//* TCP/IP for MVS
//* SMP/E Distribution Name: EZBRDORP
//*
//*       5655-HAL (C) Copyright IBM Corp. 1997
//*       All rights reserved.
//*       US Government Users Restricted Rights -
//*       Use, duplication or disclosure restricted
//*       by GSA ADP Schedule Contract with IBM Corp.
//*       See IBM Copyright Instructions
//*
//*       orouted can be started with a variety of parameters.
//*       In this example, the "-ep" parameter enables displaying
//*       program output to standard out, and the "-t -t" parameter
//*       enables 2 levels of tracing.
//*
//OROUTED  EXEC PGM=BPXBATCH,REGION=4M,TIME=NOLIMIT,
//        PARM='PGM /usr/lpp/tcpip/sbin/orouted -ep -t -t'
//*
//*       To save the program output to standard out to
//*       an HFS file, define STDOUT as show below:
//*
//STDOUT   DD PATH='/tmp/orouted.stdout',
//        PATHOPTS=(OWRONLY,OCREAT,OAPPEND),
//        PATHMODE=(SIRUSR,SIWUSR,SIRGRP,SIWGRP)
//*
//*       To capture standard error messages in a file, define
//*       STDERR
//*
//STDERR   DD PATH='/tmp/orouted.stderr',
//        PATHOPTS=(OWRONLY,OCREAT,OAPPEND),
//        PATHMODE=(SIRUSR,SIWUSR,SIRGRP,SIWGRP)
//*
//*       orouted can use certain environmental variables, such
//*       as NLSPATH (to determine the location of the message
//*       catalog), RESOLVER_CONFIG (to determine the location
//*       of the file that contains parameters TCPIPjobname and
//*       DATASETPrefix), and GATEWAYS_FILE (to determine the
//*       location of the gateways file).  Define STDENV with
//*       the name of the file that contains the environmental
//*       variables to be used for this invocation of orouted.
//*
//STDENV   DD PATH='/etc/orouted.env'
//*
//CEEDUMP  DD SYSOUT=*,DCB=(RECFM=FB,LRECL=132,BLKSIZE=132)
```

The following set of samples are related to initializing OS/390 TCP/IP OpenEdition applications.

File Transfer Protocol (FTP)

FTP Procedure:

```
//FTPD   PROC MODULE='FTPD',PARMS='PORT 621 TRACE'
//*********************************************************************
//*                                                                   *
//*         TCP/IP FOR MVS                                            *
//*                                                                   *
//*         DESCRIPTIVE NAME:            FTP SERVER START PROCEDURE   *
//*                                                                   *
//*         FILE NAME:                   TCPIP.SEZAINST(EZAFTPAP)     *
//*                                      TCPIP.SEZAINST(FTPD)         *
//*                                                                   *
//*         SMP/E DISTRIBUTION NAME:     EZAFTPAP                     *
//*                                                                   *
//*                                                                   *
//*             LICENSED MATERIALS - PROPERTY OF IBM                  *
//*             THIS PRODUCT CONTAINS "RESTRICTED MATERIALS OF IBM"   *
//*             5645-001 5655-HAL (C) COPYRIGHT IBM CORP. 1995, 1997. *
//*             ALL RIGHTS RESERVED.                                  *
//*             US GOVERNMENT USERS RESTRICTED RIGHTS -               *
//*             5645-001 5655-HAL (C) COPYRIGHT IBM CORP. 1995, 1997. *
//*             ALL RIGHTS RESERVED.                                  *
//*             US GOVERNMENT USERS RESTRICTED RIGHTS -               *
//*             USE, DUPLICATION OR DISCLOSURE RESTRICTED BY          *
//*             GSA ADP SCHEDULE CONTRACT WITH IBM CORP.              *
//*             SEE IBM COPYRIGHT INSTRUCTIONS.                       *
//*                                                                   *
//*                                                                   *
//*********************************************************************
//FTPD   EXEC PGM=&MODULE,REGION=4096K,TIME=NOLIMIT,
//      PARM='POSIX(ON) ALL31(ON)/&PARMS'
//CEEDUMP  DD SYSOUT=*
//*
//*       SYSFTPD IS USED TO SPECIFY THE FTP.DATA FILE FOR THE FTP
//*       SERVER.  THE FILE CAN BE ANY SEQUENTIAL DATA SET, MEMBER
//*       OF A PARTITIONED DATA SET (PDS), OR HFS FILE.
//*
//*       THE SYSFTPD DD STATEMENT IS OPTIONAL.  THE SEARCH ORDER FOR
//*       FTP.DATA IS:
//*
//*           /ETC/FTP.DATA
//*           SYSFTPD DD STATEMENT
//*           JOBNAME.FTP.DATA
//*           SYS1.TCPPARMS(FTPDATA)
//*           TCPIP.FTP.DATA
//*
//*       IF NO FTP.DATA FILE IS FOUND, FTP DEFAULT VALUES ARE USED.
//*       FOR INFORMATION ON FTP DEFAULTS, SEE THE CUSTOMIZATION
//*       AND ADMINISTRATION GUIDE AND TCP/IP OE MVS APPLICATIONS
//*       FEATURE GUIDE.
//*SYSFTPD DD DISP=SHR,DSN=TCPIP.SEZAINST(FTPSDATA)
//*
//*      SYSTCPD EXPLICITLY IDENTIFIES WHICH FILE IS TO BE
//*      USED TO OBTAIN THE PARAMETERS DEFINED BY TCPIP.DATA.
//*      THE SYSTCPD DD STATEMENT SHOULD BE PLACED IN THE JCL OF
//*      THE SERVER.  THE FILE CAN BE ANY SEQUENTIAL DATA SET,
//*      MEMBER OF A PARTITIONED DATA SET (PDS), OR HFS FILE.
//*SYSTCPD DD DISP=SHR,DSN=TCPIP.SEZAINST(TCPDATA)
//*
//*
```

```
//*      SYSFTSX EXPLICITLY IDENTIFIES WHICH FILE IS TO BE USED
//*      FOR THE EBCDIC-ASCII TRANSLATION TABLE.  THE FILE CAN
//*      BE ANY SEQUENTIAL DATA SET, MEMBER OF A PARTITIONED DATA
//*      SET (PDS), OR HFS FILE.
//*SYSFTSX DD DISP=SHR,DSN=TCPIP.STANDARD.TCPXLBIN
```

INETD Procedure

INETD is required for the execution of OE TELNET, Remote Execution (REXEC), and Remote Shell (RSH). The following procedure was used on the TCP/IP test systems for INETD.

```
//INETD  PROC
//*******************************************************************
//*       Sample procedure to execute INETD
//*******************************************************************
//*
//*       5655-HAL (C) Copyright IBM Corp. 1996, 1997.
//*       All rights reserved.
//*       US Government Users Restricted Rights -
//*       Use, duplication or disclosure restricted
//*       by GSA ADP Schedule Contract with IBM Corp.
//*       See IBM Copyright Instructions
//*
//*******************************************************************
//INETD EXEC PGM=INETD,REGION=0K,TIME=NOLIMIT,
//      PARM='POSIX(ON) ALL31(ON)/ /etc/inetd.conf'
//SYSPRINT DD SYSOUT=*
```

OSNMPD Procedure

The following procedure was used on the TCP/IP test systems for OSNMPD.

```
//OSNMPD    PROC
//*
//* Sample procedure for running the OE SNMP agent
//*
//* TCP/IP for MVS
//* SMP/E Distribution Name: AEZASMP1
//*
//*       Licensed Materials - Property of IBM
//*       This product contains "Restricted Materials of IBM"
//*       5645-001 5655-HAL (C) Copyright IBM Corp. 1996.
//*       All rights reserved.
//*       US Government Users Restricted Rights -
//*       Use, duplication or disclosure restricted by
//*       GSA ADP Schedule Contract with IBM Corp.
//*       See IBM Copyright Instructions.
//*
//OSNMPD EXEC PGM=EZASNMPD,REGION=4096K,TIME=NOLIMIT,
// PARM='POSIX(ON) ALL31(ON)/ -c public '
//*
//*   The C runtime libraries should be in the system's link list
//*   or this sample procedure will need to STEPLIB to them.
//*
//*   TCP/IP runtime libraries should also be in the system's link
//*   list.
//*
//*   OSNMPD must find the name (TCPIPJOBNAME in TCPIP.DATA) that
//*   it should be associated with. The OE function __iptcpn() is
//*   used to find this name. It is suggested that the parmlist
//*   be modified to set the environment variable
//*   RESOLVER_CONFIG to point to the correct resolver file when
//*   multiple INET Physical File Systems are started:
//*
//*   //      PARM=('POSIX(ON) ALL31(ON)',
//*   //      'ENVAR("RESOLVER_CONFIG=/etc/tcpv33a.data")/')
//*
//*   If only one INET PFS will be started then it is
//*   recommended that /etc/resolv.conf be used.
//*
//SYSPRINT DD SYSOUT=*,DCB=(RECFM=F,LRECL=80,BLKSIZE=80)
//SYSIN    DD DUMMY
//SYSMDUMP DD SYSOUT=*
//SYSERR   DD SYSOUT=*
//SYSOUT   DD SYSOUT=*,DCB=(RECFM=F,LRECL=80,BLKSIZE=80)
//CEEDUMP  DD SYSOUT=*
```

PORTMAP Procedure

PORTMAP is required for the execution of RPCINFO and RPCGEN. The following procedure was used on the TCP/IP test systems for PORTMAP.

```
//PORTMAP  PROC
//*
//*   TCP/IP for MVS
//*   SMP/E distribution name: EZBOPORT
//*
//*   5645-001 5655-HAL (C) Copyright IBM Corp. 1997.
//*   Licensed Materials - Property of IBM
//*   This product contains "Restricted Materials of IBM"
//*   All rights reserved.
//*   US Government Users Restricted Rights -
//*   Use, duplication or disclosure restricted by
//*   GSA ADP Schedule Contract with IBM Corp.
//*   See IBM Copyright Instructions.
//*
//*   Function: OpenEdition MVS Portmapper Server main process
//*
//PORTMAP  EXEC PGM=OPORTMAP,REGION=4096K,TIME=1440,
//           PARM='POSIX(ON),ALL31(ON)/'
//*
//STDOUT   DD SYSOUT=*
//STDERR   DD SYSOUT=*
//SYSOUT   DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//         PEND
```

SYSLOGD Procedure

SYSLOGD is required for the recording of TCP/IP V3R3 OpenEdition application events. The following procedure was used on the TCP/IP test systems for SYSLOGD.

```
//SYSLOGD PROC OPTS=''
//*********************************************************************
//*        SAMPLE PROCEDURE TO EXECUTE SYSLOGD
//*********************************************************************
//*
//*        5655-HAL (C) COPYRIGHT IBM CORP. 1996, 1997.
//*        ALL RIGHTS RESERVED.
//*        US GOVERNMENT USERS RESTRICTED RIGHTS -
//*        USE, DUPLICATION OR DISCLOSURE RESTRICTED
//*        BY GSA ADP SCHEDULE CONTRACT WITH IBM CORP.
//*        SEE IBM COPYRIGHT INSTRUCTIONS
//*
//*********************************************************************
//SYSLOGD EXEC PGM=SYSLOGD,REGION=0K,TIME=NOLIMIT,PARM=&OPTS
//SYSPRINT DD SYSOUT=*
```

OS/390 TCP/IP OpenEdition HFS Files

The following samples show the changed OS/390 TCP/IP OpenEdition HFS files used in the test system configuration and the HFS files required for OROUTED.

Services

The following file resides in the HFS /etc/services.

```
#
#     /etc/services file
#
echo            7/tcp
echo            7/udp
discard         9/tcp           sink null
discard         9/udp           sink null
systat          11/tcp          users
daytime         13/tcp
daytime         13/udp
netstat         15/tcp
qotd            17/tcp          quote
chargen         19/tcp          ttytst source
chargen         19/udp          ttytst source
otelnet         623/tcp
smtp            25/tcp          mail
time            37/tcp          timserver
time            37/udp          timserver
rlp             39/udp          resource        # resource location
nameserver      42/tcp          name            # IEN 116
whois           43/tcp          nicname
domain          53/tcp          nameserver      # name-domain server
domain          53/udp          nameserver
mtp             57/tcp                          # deprecated
tftp            69/udp
rje             77/tcp          netrjs
finger          79/tcp
link            87/tcp          ttylink
supdup          95/tcp
hostnames       101/tcp         hostname        # usually from sri-nic
pop             109/tcp         postoffice
sunrpc          111/tcp
sunrpc          111/udp
auth            113/tcp         authentication
sftp            115/tcp
uucp-path       117/tcp
nntp            119/tcp         readnews untp   # USENET News Transfer Protocol
snmp            161/udp                         # snmp request port
snmp-trap       162/udp                         # snmp monitor trap port
exec            512/tcp
biff            512/udp         comsat
login           513/tcp
who             513/udp         whod
shell           514/tcp         cmd             # no passwords used
shell           514/udp         cmd             # no passwords used
syslog          529/udp
printer         515/tcp         spooler         # line printer spooler
talk            517/udp
ntalk           518/udp
efs             520/tcp                         # for LucasFilm
route           520/udp         router routed
timed           525/udp         timeserver
tempo           526/tcp         newdate
courier         530/tcp         rpc
conference      531/tcp         chat
```

```
netnews         532/tcp         readnews
netwall         533/udp                         # -for emergency broadcasts
uucp            540/tcp         uucpd           # uucp daemon
remotefs        556/tcp         rfs_server rfs  # Brunhoff remote filesystem
ncprout         580/udp         ncproute
ftp             621/tcp                         # OE ftpd
otelnet         623/tcp                         # OE telnetd
ingreslock      1524/tcp
rvd-control     531/udp                         # rvd control port
filesrv         2001/tcp
console         2018/udp
venus.itc       2106/tcp
client          2030/tcp
vexec           712/tcp         vice-exec
vlogin          713/tcp         vice-login
vshell          714/tcp         vice-shell
venus.itc       2106/tcp
rauth2          2001/udp
rfilebulk               2002/udp
rfilesrv                2003/udp
ropcons         2115/udp
rupdsrv         2131/udp
rupdbulk                2132/udp
rupdsrv1                2133/udp
rupdbulk1               2134/udp
klogin          543/tcp           # Kerberos aythenticated rlogin
kerberos        750/udp      kdc  # Kerberos aythentication--udp
kerberos        750/tcp      kdc  # Kerberos aythentication--tcp
kerboros_master 751/udp           # Kerberos aythentication
kerberos_master 751/tcp           # Kerberos aythentication
passwd_server   752/udp           # Kerberos passwd server
userreg_server  753/tcp           # Kerberos userreg serrver
kpop            1109/tcp          # Pop with Kerberos
knetd         2053/tcp                # Kerberos de-multiplexor
kshell          544/tcp         cmd     # and remote shell
eklogin         2105/tcp                # Kerberos encrypted rlogin
krb_prop        754/tcp                 # Kerboros slpave propagation
erlogin         888/tcp                 # Login and environment passing
sample        906/tcp                 # Kerberos sample app server
sample                        906/udp                #for kerberos simple test
```

INETD.CONF

The following file resides in the HFS /etc/inetd.conf.

```
#
#          /etc/inetd.conf
#
#          Internet server configuration database
#
#======================================================================
# service | socket | protocol | wait/ | user | server   | server program
# name    | type   |          | nowait|      | program |   arguments
#======================================================================
#
otelnet  stream tcp nowait bpxroot /usr/sbin/otelnetd otelnetd -t -D all
login     stream tcp nowait BPXROOT /usr/sbin/rlogind rlogind -m
shell   stream  tcp  nowait  bpxroot /usr/sbin/orshd      orshd -dLV
exec    stream  tcp  nowait  bpxroot /usr/sbin/orexecd    orexecd -dLV
echo     stream tcp nowait root internal
discard  stream tcp nowait root internal
chargen  stream tcp nowait root internal
daytime  stream tcp nowait root internal
time     stream tcp nowait root internal
echo     dgram  udp wait   root internal
discard  dgram  udp wait   root internal
chargen  dgram  udp wait   root internal
daytime  dgram  udp wait   root internal
time     dgram  udp wait   root internal
```

HOSTS

The following file resides in the HFS /etc/hosts.

```
#
#  /etc/hosts file
#
# This file contains the hostnames and their address for hosts in the
# network.  This file is used to resolve a hostname into an Internet
# address.
#
# The format of this file is:
# Internet Address Hostname # Comments
#
# Internet   Hostname                      # Comments
# Address
127.0.0.1    localhost
9.67.113.51  MVS6 mvs6
9.67.116.37  mvs6-mvsa
9.37.38.7    coco
9.67.43.62   ralvm12
9.67.116.38  mvsa-mvs6
9.67.113.53  mvsa MVSA
```

RESOLV.CONF

The following file resides in the HFS /etc/resolv.conf.

```
TCPIPjobname  TCPV33
Datasetprefix TCPV33
Messagecase mixed
HostName MVS6
Domain   tcp.raleigh.ibm.com
nameserver  9.37.32.94
nameserver  9.37.32.3
nameserver  9.67.5.44
```

OROUTED.ENV

The following file resides in the HFS /etc/resolv.env

```
RESOLVER_CONF=/etc/resolv.conf
GATEWAYS_FILE=/etc/gateways
```

GATEWAYS

The following file resides in the HFS /etc/gateways.

```
host 9.67.116.63 gateway 9.67.116.62 metric 1 passive
```

SYSLOG.CONF

The following file resides in the HFS /etc/syslog.conf.

```
local1.debug   /tmp/telnetd.syslog
local2.debug   /tmp/rcmds.syslog
daemon.info    /tmp/daemon.log
daemon.debug   /tmp/daemon.trace
```

OSNMPD.DATA

The following file resides in the HFS /etc/osnmpd.data.

```
#
#    /etc/osnmpd.data
#
   sysDescr  "SNMP agent on MVS6"
   sysContact  "MVS6 Operations   "
   sysLocation "RTP NC         "
   sysName "MVS6"
   sysObjectID "1.3.6.1.4.1.2.14"
   sysServices 3
   snmpEnableAuthenTraps  2
   dpiPathNameForUnixStream "/tmp/dpi_new_socket"
   saMaxTimeout 700
   saAllowDuplicateIDs 2
```

Appendix H. Related Protocol Specifications (RFCs)

This appendix lists the related protocol specifications for TCP/IP for MVS. The Internet suite of protocols is still evolving through Requests for Comments (RFC). New protocols are being designed and implemented by researchers, and are brought to the attention of the Internet community in the form of RFCs. Some of these are so useful that they become recommended protocols. That is, all future implementations for TCP/IP are recommended to implement this particular function or protocol. These become the *de facto* standards, on which the TCP/IP protocol suite is built.

Many features of TCP/IP for MVS are based on the following RFCs:

RFC	**Title and Author**
768	*User Datagram Protocol*, J.B. Postel
791	*Internet Protocol*, J.B. Postel
792	*Internet Control Message Protocol*, J.B. Postel
793	*Transmission Control Protocol*, J.B. Postel
821	*Simple Mail Transfer Protocol*, J.B. Postel
822	*Standard for the Format of ARPA Internet Text Messages*, D. Crocker
823	*DARPA Internet Gateway*, R.M. Hinden, A. Sheltzer
826	*Ethernet Address Resolution Protocol: or Converting Network Protocol Addresses to 48.Bit Ethernet Address for Transmission on Ethernet Hardware*, D.C. Plummer
854	*Telnet Protocol Specification*, J.B. Postel, J.K. Reynolds
856	*Telnet Binary Transmission*, J.B. Postel, J.K. Reynolds
857	*Telnet Echo Option*, J.B. Postel, J.K. Reynolds
862	*Echo Protocol*, J.B. Postel
863	*Discard Protocol*, J.B. Postel
864	*Character Generator Protocol*, J.B. Postel
877	*Standard for the Transmission of IP Datagrams over Public Data Networks*, J.T. Korb
885	*Telnet End of Record Option*, J.B. Postel
903	*Reverse Address Resolution Protocol*, R. Finlayson, T. Mann, J.C. Mogul, M. Theimer
904	*Exterior Gateway Protocol Formal Specification*, D.L. Mills
919	*Broadcasting Internet Datagrams*, J.C. Mogul
922	*Broadcasting Internet Datagrams in the Presence of Subnets*, J.C. Mogul
950	*Internet Standard Subnetting Procedure*, J.C. Mogul, J.B. Postel
952	*DoD Internet Host Table Specification*, K. Harrenstien, M.K. Stahl, E.J. Feinler
959	*File Transfer Protocol*, J.B. Postel, J.K. Reynolds

974 *Mail Routing and the Domain Name System*, C. Partridge

1009 *Requirements for Internet Gateways*, R.T. Braden, J.B. Postel

1013 *X Window System Protocol, Version 11: Alpha Update*, R.W. Scheifler

1014 *XDR: External Data Representation Standard*, Sun Microsystems** Incorporated

1027 *Using ARP to Implement Transparent Subnet Gateways*, S. Carl-Mitchell, J.S. Quarterman

1032 *Domain Administrators Guide*, M.K. Stahl

1033 *Domain Administrators Operations Guide*, M. Lottor

1034 *Domain Names—Concepts and Facilities*, P.V. Mockapetris

1035 *Domain Names—Implementation and Specification*, P.V. Mockapetris

1042 *Standard for the Transmission of IP Datagrams over IEEE 802 Networks*, J.B. Postel, J.K. Reynolds

1044 *Internet Protocol on Network System's HYPERchannel**: Protocol Specification*, K. Hardwick, J. Lekashman

1055 *Nonstandard for Transmission of IP Datagrams over Serial Lines: SLIP*, J.L. Romkey

1057 *RPC: Remote Procedure Call Protocol Version 2 Specification*, Sun Microsystems Incorporated

1058 *Routing Information Protocol*, C.L. Hedrick

1091 *Telnet Terminal-Type Option*, J. VanBokkelen

1094 *NFS**: Network File System Protocol Specification*, Sun Microsystems Incorporated

1118 *Hitchhikers Guide to the Internet*, E. Krol

1122 *Requirements for Internet Hosts—Communication Layers*, R.T. Braden

1123 *Requirements for Internet Hosts—Application and Support*, R.T. Braden

1155 *Structure and Identification of Management Information for TCP/IP-Based Internets*, M.T. Rose, K. McCloghrie

1156 *Management Information Base for Network Management of TCP/IP-based Internets*, K. McCloghrie, M.T. Rose

1157 *Simple Network Management Protocol (SNMP),* J.D. Case, M. Fedor, M.L. Schoffstall, C. Davin

1179 *Line Printer Daemon Protocol*, The Wollongong** Group, L. McLaughlin III

1180 *TCP/IP Tutorial,* T.J. Socolofsky, C.J. Kale

1183 *New DNS RR Definitions*, C.F. Everhart, L.A. Mamakos, R. Ullmann, P.V. Mockapetris, (Updates RFC 1034, RFC 1035)

1187 *Bulk Table Retrieval with the SNMP*, M.T. Rose, K. McCloghrie, J.R. Davin

1188 *Proposed Standard for the Transmission of IP Datagrams over FDDI Networks*, D. Katz

1198 *FYI on the X Window System*, R.W. Scheifler

1207 *FYI on Questions and Answers: Answers to Commonly Asked "Experienced Internet User" Questions*, G.S. Malkin, A.N. Marine, J.K. Reynolds

1208 *Glossary of Networking Terms*, O.J. Jacobsen, D.C. Lynch

1213 *Management Information Base for Network Management of TCP/IP-Based Internets: MIB-II,* K. McCloghrie, M.T. Rose

1215 *Convention for Defining Traps for Use with the SNMP*, M.T. Rose

1228 *SNMP-DPI Simple Network Management Protocol Distributed Program Interface*, G.C. Carpenter, B. Wijnen

1229 *Extensions to the Generic-Interface MIB*, K. McCloghrie

1230 *IEEE 802.4 Token Bus MIB IEEE 802 4 Token Bus MIB*, K. McCloghrie, R. Fox

1231 *IEEE 802.5 Token Ring MIB IEEE 802.5 Token Ring MIB*, K. McCloghrie, R. Fox, E. Decker

1267 *A Border Gateway Protocol 3 (BGP-3)*, K. Lougheed, Y. Rekhter

1268 *Application of the Border Gateway Protocol in the Internet*, Y. Rekhter, P. Gross

1269 *Definitions of Managed Objects for the Border Gateway Protocol (Version 3)*, S. Willis, J. Burruss

1270 *SNMP Communications Services*, F. Kastenholz, ed.

1325 *FYI on Questions and Answers: Answers to Commonly Asked "New Internet User" Questions*, G.S. Malkin, A.N. Marine

1340 *Assigned Numbers*, J.K. Reynolds, J.B. Postel

1348 *DNS NSAP RRs*, B. Manning

1350 *TFTP Protocol*, K.R. Sollins

1351 *SNMP Administrative Model*, J. Davin, J. Galvin, K. McCloghrie

1352 *SNMP Security Protocols*, J. Galvin, K. McCloghrie, J. Davin

1353 *Definitions of Managed Objects for Administration of SNMP Parties*, K. McCloghrie, J. Davin, J. Galvin

1354 *IP Forwarding Table MIB*, F. Baker

1356 *Multiprotocol Interconnect on X.25 and ISDN in the Packet Mode*, A. Malis, D. Robinson, R. Ullmann

1374 *IP and ARP on HIPPI*, J. Renwick, A. Nicholson

1381 *SNMP MIB Extension for X.25 LAPB*, D. Throop, F. Baker

1382 *SNMP MIB Extension for the X.25 Packet Layer*, D. Throop

1387 *RIP Version 2 Protocol Analysis*, G. Malkin

1388 *RIP Version 2 — Carrying Additional Information*, G. Malkin

1389 *RIP Version 2 MIB Extension*, G. Malkin

1390 *Transmission of IP and ARP over FDDI Networks*, D. Katz

1393 *Traceroute Using an IP Option*, G. Malkin

1397 *Default Route Advertisement In BGP2 And BGP3 Versions of the Border Gateway Protocol*, D. Haskin

1398 *Definitions of Managed Objects for the Ethernet-like Interface Types*, F. Kastenholz

1540 *IAB Official Protocol Standards*, J.B. Postel

1592 *Simple Network Management Protocol Distributed Protocol Interface Version 2.0*, B. Wijnen, G. Carpenter, K. Curran, A. Sehgal, G. Waters

1901 *Introduction to Community-based SNMPv2*, J. Case, K. McCloghrie, M. Rose, S. Waldbusser

1902 *Structure of Management Information for Version 2 of the Simple Network Management Protocol (SNMPv2)*, J. Case, K. McCloghrie, M. Rose, S. Waldbusser

1903 *Textual Conventions for Version 2 of the Simple Network Management Protocol (SNMPv2)*, J. Case, K. McCloghrie, M. Rose, S. Waldbusser

1904 *Conformance Statements for Version 2 of the Simple Network Management Protocol (SNMPv2)*, J. Case, K. McCloghrie, M. Rose, S. Waldbusser

1905 *Protocol Operations for Version 2 of the Simple Network Management Protocol (SNMPv2)*, J. Case, K. McCloghrie, M. Rose, S. Waldbusser

1906 Transport Mappings for Version 2 of the Simple Network Management Protocol (SNMPv2), J. Case, K. McCloghrie, M. Rose, S. Waldbusser

1907 *Information Base for Version 2 of the Simple Network Management Protocol (SNMPv2)*, J. Case, K. McCloghrie, M. Rose, S. Waldbusser

1908 *Coexistence between Version 1 and Version 2 of the Internet -standard Network Management Framework*, J. Case, K. McCloghrie, M. Rose, S. Waldbusser

1909 *An Administrative Infrastructure for SNMPv2*, K. McCloghrie

1910 *User-based Security Model for SNMPv2* G. Waters

These documents can be obtained from:

Government Systems, Inc.
Attn: Network Information Center
14200 Park Meadow Drive
Suite 200
Chantilly, VA 22021

Many RFCs are available online. Hard copies of all RFCs are available from the NIC, either individually or on a subscription basis. Online copies are available using FTP from the NIC at `nic.ddn.mil`. Use FTP to download the files, using the following format:

```
RFC:RFC-INDEX.TXT
RFC:RFCnnnn.TXT
RFC:RFCnnnn.PS
```

Where:

nnnn Is the RFC number.
TXT Is the text format.
PS Is the PostScript** format.

You can also request RFCs through electronic mail, from the automated NIC mail server, by sending a message to `service@nic.ddn.mil` with a subject line of `RFC` *nnnn* for text versions or a subject line of `RFC` *nnnn*`.PS` for PostScript versions. To request a copy of the RFC index, send a message with a subject line of `RFC INDEX`.

For more information, contact `nic@nic.ddn.mil`.

Appendix I. Notices

References in this publication to IBM products, programs, or services do not imply that IBM intends to make them available in all countries in which IBM operates. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Subject to IBM's valid intellectual property or other legally protectable rights, any functionally equivalent product, program, or service may be used instead of the IBM product, program, or service. The evaluation and verification of operation in conjunction with other products, except those expressly designated by IBM, are the responsibility of the user.

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
USA

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

Site Counsel
IBM Corporation
P.O. Box 12195
3039 Cornwallis Road
Research Triangle Park, NC 27709-2195
USA

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 document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement.

This document is not intended for production use and is furnished as is without any warranty of any kind, and all warranties are hereby disclaimed including the warranties of merchantability and fitness for a particular purpose.

IBM is required to include the following statements in order to distribute portions of this document and the software described herein to which contributions have been made by Sun Microsystems, Massachusetts Institute of Technology, Digital Equipment Corporation, and The University of California.

Portions herein © Copyright 1979, 1980, 1983, 1986, Regents of the University of California. Reproduced by permission. Portions herein were developed at the Electrical Engineering and Computer Sciences Department at the Berkeley campus of the University of California under the auspices of the Regents of the University of California.

Portions of this publication relating to RPC are Copyright © Sun Microsystems, Inc., 1988, 1989.

Some portions of this publication relating to X Window System** are Copyright © 1987, 1988 by Digital Equipment Corporation, Maynard, Massachusetts, and the Massachusetts Institute Of Technology, Cambridge, Massachusetts. All Rights Reserved.

Some portions of this publication relating to X Window System are Copyright © 1986, 1987, 1988 by Hewlett-Packard Corporation.

Permission to use, copy, modify, and distribute the M.I.T., Digital Equipment Corporation, and Hewlett-Packard Corporation portions of this software and its documentation for any purpose without fee is hereby granted, provided that the above copyright notice appears in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the names of M.I.T., Digital, and Hewlett-Packard not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. M.I.T., Digital, and Hewlett-Packard make no representation about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

Trademarks

The following terms are trademarks of the IBM Corporation in the United States or other countries or both:

ACF/VTAM
AD/Cycle
AIX
AIX/ESA
BookManager
C/370
CICS
DB2
DFSMS
DFSMS/MVS
ESCON
ES/9000
EtherStreamer
Extended Services
GDDM
Hardware Configuration Definition
IBM
LANStreamer
Library Reader
MVS/ESA
MVS/SP
MVS/XA
NetView
OpenEdition
OS/2
OS/390
PS/2
RACF
RISC System/6000
RS/6000
SAA
S/390
System/370
System/390
TURBOWAYS
VTAM
3090

The following terms are trademarks of other companies:

UNIX is a registered trademark in the United States and other countries licensed exclusively through X/Open Company Limited.

Other company, product, and service names, which may be denoted by a double asterisk (**), may be trademarks or service marks of others.

Glossary, Bibliography, and Index

Glossary

The IBM Networking Software Glossary is now available in HTML format as well as PDF. You can access it directly at the following URL:

```
http://www.networking.ibm.com/nsg/nsggls.htm
```

This glossary includes terms and definitions from:

- The *American National Standards Dictionary for Information Systems*, ANSI X3.172-1990, copyright 1990 by the American National Standards Institute (ANSI). Copies may be purchased from the American National Standards Institute, 11 West 42nd Street, New York, New York 10036. Definitions are identified by the symbol (A) after the definition.
- The ANSI/EIA Standard—440-A, *Fiber Optic Terminology*. Copies may be purchased from the Electronic Industries Association, 2001 Pennsylvania Avenue, N.W., Washington, DC 20006. Definitions are identified by the symbol (E) after the definition.
- The *Information Technology Vocabulary*, developed by Subcommittee 1, Joint Technical Committee 1, of the International Organization for Standardization and the International Electrotechnical Commission (ISO/IEC JTC1/SC1). Definitions of published parts of this vocabulary are identified by the symbol (I) after the definition; definitions taken from draft international standards, committee drafts, and working papers being developed by ISO/IEC JTC1/SC1 are identified by the symbol (T) after the definition, indicating that final agreement has not yet been reached among the participating National Bodies of SC1.
- The *IBM Dictionary of Computing*, New York: McGraw-Hill, 1994.
- Internet Request for Comments: 1208, *Glossary of Networking Terms*.
- Internet Request for Comments: 1392, *Internet Users' Glossary*.
- The *Object-Oriented Interface Design: IBM Common User Access Guidelines*, Carmel, Indiana: Que, 1992.

Bibliography

This bibliography lists the publications for IBM TCP/IP products.

IBM TCP/IP Publications

The following sections describe the books associated with IBM TCP/IP products.

OS/390 TCP/IP OpenEdition Publications

- *OS/390 TCP/IP OpenEdition Configuration Guide*, SC31-8304.

  This book is for people who want to configure, customize, administer, and maintain OS/390 TCP/IP OpenEdition. Familiarity with MVS operating system, TCP/IP protocols, and IBM Time Sharing Option (TSO) is recommended.

- *OS/390 TCP/IP OpenEdition Diagnosis Guide*, SC31-8492.

  This book explains how to diagnose TCP/IP problems and how to determine whether a specific problem is in the IBM OS/390 TCP/IP OpenEdition product code. It explains how to gather information for and describe problems to the IBM Software Support Center.

- *OS/390 TCP/IP OpenEdition Messages and Codes*, SC31-8307.

  This book explains the informational and error messages issued by IBM OS/390 TCP/IP OpenEdition. It can help users, operators, or system programmers to diagnose and fix problems identified by OS/390 TCP/IP OpenEdition error messages.

- *OS/390 TCP/IP OpenEdition Planning and Release Guide*, SC31-8303.

  This book is intended to help you plan for OS/390 TCP/IP OpenEdition for MVS whether you are migrating from a previous version or installing TCP/IP for the first time. This book also identifies the suggested and required modifications needed to enable you to use the enhanced functions provided with OS/390 TCP/IP OpenEdition.

- *OS/390 TCP/IP OpenEdition Programmer's Reference*, SC31-8308.

  This book describes the syntax and semantics of a set of high-level application functions that you can use to program your own applications in a TCP/IP environment. These functions provide support for application facilities, such as user authentication, distributed databases, distributed processing, network management, and device sharing.

  This book is for people who want to use the supplied interfaces while writing application programs that access OS/390 TCP/IP OpenEdition. Familiarity with the MVS operating system, TCP/IP protocols, and IBM Time Sharing Option (TSO) is recommended.

- *OS/390 TCP/IP OpenEdition User's Guide*, GC31-8305.

  This book is for people who want to use OS/390 TCP/IP OpenEdition for data communication. Familiarity with MVS operating system and IBM Time Sharing Option (TSO) is recommended.

TCP/IP for MVS Publications

- *TCP/IP Version 3 for OpenEdition MVS: Applications Feature Guide*, SC31-8069

  This book explains how to plan for, install, customize, and use the OpenEdition MVS Applications Feature. The Feature consists of applications and interfaces for direct access to the OpenEdition MVS environment. For example, users of the Feature can use MVS, UNIX, or AIX commands to transfer files, log in to the OpenEdition environment without going through TSO, and run commands remotely. This book also explains how to improve performance and diagnose problems when using the Feature.

- *TCP/IP for MVS: Application Programming Interface Reference*, SC31-7187

  This book describes the syntax and semantics of program source code necessary to write your own application programming interface (API) into TCP/IP. You can use this interface as the communication base for writing your own client or server application. You can also use this book to adapt your existing applications to communicate with each other using sockets over TCP/IP.

- *TCP/IP for MVS: CICS TCP/IP Socket Interface Guide and Reference*, SC31-7131

  This book is for people who want to set up, write application programs for, and diagnose problems with the socket interface for CICS using TCP/IP for MVS.

- *TCP/IP for MVS: Customization and Administration Guide*, SC31-7134.

  This book is for people who want to configure, customize, administer, and maintain TCP/IP for MVS. Familiarity with MVS operating system, TCP/IP protocols, and IBM Time Sharing Option (TSO) is recommended.

- *TCP/IP for MVS: Diagnosis Guide*, LY43-0105

  This book explains how to diagnose TCP/IP problems and how to determine whether a specific problem is in the IBM TCP/IP for MVS product code. It explains how to gather information for and describe problems to the IBM Software Support Center.

- *TCP/IP for MVS: IMS TCP/IP Application Development Guide and Reference*, SC31-7186

  This book is for programmers who want to write application programs that use the IMS TCP/IP application development services provided by IBM TCP/IP for MVS.

- *TCP/IP for MVS: Messages and Codes*, SC31-7132.

  This book explains the informational and error messages issued by IBM TCP/IP for MVS. It can help users, operators, or system programmers to diagnose and fix problems identified by TCP/IP for MVS error messages.

- *TCP/IP for MVS: Network Print Facility*, SC31-8074

  This book is for system programmers and network administrators who need to prepare their network to route VTAM, JES2, or JES3 printer output to remote printers using TCP/IP for MVS.

- *TCP/IP for MVS: Offloading TCP/IP Processing*, SC31-7133

  This book is for people who want to install and configure the Offload feature on IBM 3172 Model 3 Interconnect Controllers. This book is also for people who want to use and customize the Offload feature of TCP/IP for MVS.

- *TCP/IP for MVS: Planning and Migration Guide*, SC31-7189

  This book is intended to help you plan for TCP/IP for MVS whether you are migrating from a previous version or installing TCP/IP for MVS for the first time. This book also identifies the suggested and required modifications needed to enable you to use the enhanced functions provided with TCP/IP for MVS.

- *TCP/IP: Performance Tuning Guide*, SC31-7188

  This book describes how to improve the performance of your network operations.

- *TCP/IP for MVS: Programmer's Reference*, SC31-7135

  This book describes the syntax and semantics of a set of high-level application functions that you can use to program your own applications in a TCP/IP environment. These functions provide support for application facilities, such as user authentication, distributed databases, distributed processing, network management, and device sharing.

  This book is for people who want to use the supplied interfaces while writing application programs that access TCP/IP for MVS. Familiarity with the MVS operating system, TCP/IP protocols, and IBM Time Sharing Option (TSO) is recommended.

- *TCP/IP for MVS: User's Guide*, SC31-7136

  This book is for people who want to use TCP/IP for MVS for data communication. Familiarity with MVS operating system and IBM Time Sharing Option (TSO) is recommended.

TCP/IP for VM Publications

The following list describes books in the IBM TCP/IP for VM library.

- *IBM TCP/IP Version 2 Release 4 for VM: Messages and Codes*, SC31-6151.

  This book is for system programmers who want to diagnose and fix problems identified by TCP/IP for VM error messages.

- *IBM TCP/IP Version 2 Release 4 for VM: Planning and Customization*, SC31-6082.

  This book is for system programmers who want to plan and customize the TCP/IP for VM environment.

- *IBM TCP/IP Version 2 Release 4 for VM: Programmer’s Reference*, SC31-6084.

  This book is for application and system programmers who want to write application programs that use TCP/IP for VM. Application programmers should know the VM operating system.

- *IBM TCP/IP Version 2 Release 4 for VM: User’s Guide*, SC31-6081.

  This book is for people who want to use TCP/IP for VM for data communication. Familiarity with VM operating system, IBM Command Processor (CP), and IBM Conversational Monitor System (CMS) is recommended.

TCP/IP for OS/2 Publication

IBM TCP/IP Version 3.0 for OS/2: Programmer’s Reference, SC31-6077.

This book provides application and system programmers with the information required to write application programs that use TCP/IP for OS/2. Programmers should know the OS/2 operating system.

TCP/IP for DOS Publications

The following list describes books in the IBM TCP/IP for DOS library.

- *IBM TCP/IP Version 2.1.1 for DOS: Command Reference*, SX75-0083.

  This book is for people who use a workstation with TCP/IP for DOS, such as end users and system programmers. The people who use this book should be familiar with DOS and the workstation, understand DOS operating system concepts, and be familiar with the *IBM TCP/IP Version 2.1.1 for DOS: User's Guide*.

- *IBM TCP/IP Version 2.1.1 for DOS: Installation and Administration*, SC31-7047.

  This book provides system programmers, network administrators, and workstation users responsible for installing TCP/IP for DOS with the information required to plan and implement the installation of TCP/IP for DOS. The topics include hardware and software requirements, pre-installation system performance considerations, instructions for installing TCP/IP for DOS, instructions for customizing the TCP/IP for DOS environment, and installation examples.

- *IBM TCP/IP Version 2.1.1 for DOS: Programmer's Reference*, SC31-7046.

  This book is for application and system programmers to aid them in writing application programs that use TCP/IP for DOS on a workstation. Application programmers should know the DOS operating system and multitasking operating system concepts. Application programmers should be knowledgeable in the C programming language.

- *IBM TCP/IP Version 2.1.1 for DOS: User's Guide*, SC31-7045.

  This book is for people who use a workstation with TCP/IP for DOS, such as end users and system programmers. The people who use this book should be familiar with DOS and the workstation, and also understand DOS operating system concepts.

TCP/IP for AIX (RS/6000, PS/2, RT, 370) Publications

The following list shows books in the TCP/IP for AIX library.

- *AIX Operating System TCP/IP User's Guide*, SC23-2309.
- *AIX PS/2 TCP/IP User's Guide*, SC23-2047.
- *TCP/IP for IBM X-Windows on DOS 2.1*, SC23-2349.

TCP/IP for AS/400 Publications

The following list shows books in the TCP/IP for AS/400 library.

- *IBM AS/400 Communications: TCP/IP Guide*, SC41-9875.
- *IBM AS/400 Communications: User's Guide*, SC21-9601.

Other IBM TCP/IP Publications

The following list shows other available IBM TCP/IP books.

- *IBM Local Area Network Technical Reference*, SC30-3383.
- *IBM TCP/IP for VM and MVS: Diagnosis Guide*, LY43-0013.
- *TCP/IP and National Language Support*, GG24-3840.
- *TCP/IP Introduction*, GC31-6080.
- *TCP/IP Tutorial and Technical Overview*, GG24-3376.

IBM Operating System Publications

The following lists show books about various IBM operating systems.

AIX Publications

- *AIX Communications Concepts and Procedures for IBM RISC System/6000*, GC23-2203.
- *AIX Communications Programming Concepts*, SC23-2206.
- *IBM AIX Operating System Technical Reference, Volume 1*, SC23-2300.
- *IBM AIX Operating System Technical Reference, Volume 2*, SC23-2301.

AS/400 Publications

- *IBM AS/400 CL Reference Manual Volume 1*, SC21-9775.
- *IBM AS/400 CL Reference Manual Volume 2*, SC21-9776.
- *IBM AS/400 CL Reference Manual Volume 3*, SC21-9777.
- *IBM AS/400 CL Reference Manual Volume 4*, SC21-9778.
- *IBM AS/400 CL Reference Manual Volume 5*, SC21-9779.

- *IBM AS/400 Communications: APPN Network User's Guide*, SC21-8188.
- *IBM AS/400 Communications: Programmer's Guide*, SC21-9590.
- *IBM AS/400 Communications: User's Guide*, SC21-9601.
- *IBM AS/400 Device Configuration Guide*, SC21-8106.
- *IBM AS/400 Programming: Command Reference Summary*, SC21-8076.
- *IBM AS/400 Programming: Data Management Guide*, SC21-9658.
- *IBM AS/400 System Operations: Database Coordinator' Guide*, SC21-8086.
- *IBM AS/400 System Operations: Operator's Guide*, SC21-8082.

DOS Publications

- *DOS Getting Started Version 5.00*, SA40-0637.
- *DOS 5.02 Technical Reference*, S16G-4559.
- *DOS/Windows Client Getting Started*, SC09-3000.
- *PC DOS 6.1 Command Reference*, S71G-3634.

MVS Publications

For a complete description of the library for MVS/ESA Version 5, see *OS/390 Information Roadmap*, GC28-1727. See also "JES Publications" on page 135.

OS/2 Publications

- *IBM OS/2 Warp Server Up and Running!*, S25H-8004
- *IBM Official Guide to Using OS/2 Warp*, ISBN 1-56884-466-2 (Karla Stagray and Linda S. Rogers; Foster City, CA: An IBM Press Book published by IDG Books Worldwide, Inc., 1995)
- *IBM OS/2 Warp Internet Connection: Your Key to Cruising the Internet and the World Wide Web*, ISBN 1-56884-465-4 (Deborah Morrison; Foster City, CA: An IBM Press Book published by IDG Books Worldwide, Inc., 1995)

OS/390 Publications

- *OS/390 Information Roadmap*, GC28-1727.

  This book describes the documentation for the specific elements included in OS/390.
- *OS/390 Planning for Installation Release 3*, GC28-1726.

  This book is intended to help you plan for the installation of OS/390. It describes migration, installation, hardware and software requirements, and coexistence considerations.
- *OS/390 OpenEdition Introduction*, GC28-1889.
- *OS/390 OpenEdition Planning*, SC28-1890.
- *OS/390 OpenEdition User's Guide*, SC28-1891.
- *OS/390 OpenEdition Command Reference*, SC28-1892.
- *OS/390 OpenEdition Messages and Codes*, SC28-1908.
- *OS/390 Language Environment Programming Guide*, SC28-1939.
- *OS/390 Language Environment Programming Reference*, SC28-1940.
- *OS/390 OpenEdition Programming: Assembler Callable Services Reference*, GC23-3870.
- *OS/390 Open Systems Adapter Support Facility User's Guide*, SC28-1855.
- *Planning for the System/390 Open Systems Adapter Feature*, GC23-3870.

VM Publications

- *VM/ESA CMS Command Reference Summary*, SX24-5249.
- *VM/ESA CP Planning and Administration for 370*, SC24-5430.
- *VM/ESA CP Programming Services for 370*, SC24-5435.
- *VM/ESA Group Control System Reference for 370*, SC24-5426.
- *VM/ESA: Library Guide and Master Index*, GC23-0367.
- *VM/ESA: Master Index for 370*, GC24-5436.
- *VM/ESA Service Introduction and Reference*, SC24-5444.
- *VM/SP CMS Command Reference*, ST00-1981.
- *VM/SP Group Control System Macro Reference*, SC24-5250.
- *VM/SP Installation Guide*, SC24-5237.
- *VM/SP High Performance Option: Library Guide and Master Index*, GC23-0187.
- *VM/SP System Facilities for Programming*, SC24-5288.
- *VM/XA CP Programming Services*, SC23-0370.
- *VM/XA Diagnosis Reference*, LY27-8054.
- *VM/XA Installation and Service*, SC23-0364.

- *VM/XA SP Group Control System Command and Macro Reference*, SC23-0433.

IBM Software Publications

The following sections describe the books associated with IBM software products.

ACF/VTAM Publications

The following list shows books in the VTAM Version 4 Release 4 library.

- *VTAM Installation and Migration Guide*, GC31-8367.
- *VTAM Release Guide*, GC31-6545.
- *VTAM Network Implementation Guide*, SC31-8370.
- *VTAM Resource Definition Reference*, SC31-8377.
- *VTAM Resource Definition Samples*, SC31-8378.
- *VTAM Customization*, LY43-0075.
- *VTAM Operation*, SC31-8372.
- *VTAM Messages*, GC31-8368.
- *VTAM Codes*, GC31-8369
- *VTAM Programming*, SC31-8373.
- *VTAM Guide to Programming for LU 6.2*, SC31-8374.
- *VTAM Programming Reference for LU 6.2*, SC31-8375.
- *VTAM Programming for CSM*, SC31-8420.
- *VTAM CMIP Services and Topology Agent Programming Guide*, SC31-8365.
- *VTAM Diagnosis*, LY43-0078.
- *VTAM Data Areas for MVS/ESA Volume 1*, LY43-0076.
- *VTAM Data Areas for MVS/ESA Volume 2*, LY40-0077.
- *APPC Application Suite User's Guide*, SC31-6532.
- *APPC Application Suite Administration*, SC31-6533.
- *APPC Application Suite Programming*, SC31-6534.
- *VTAM AnyNet Guide to Sockets over SNA*, SC31-8371.
- *VTAM AnyNet Guide to SNA over TCP/IP*, SC31-8376.
- *VTAM Glossary*, GC31-8366.
- *Planning for NetView, NCP, and VTAM*, SC31-8063
- *Planning for Integrated Networks*, SC31-8062
- *VTAM Licensed Program Specifications*, GC31-8379.
- *VTAM Operation Quick Reference*, SX75-0208.

DATABASE 2 Publications

The following lists show books in the DATABASE 2 library.

DATABASE 2 Version 2

- *IBM DATABASE 2 Version 2: Administration Guide*, SC26-4374.
- *IBM DATABASE 2 Version 2: Application Programming and SQL Guide*, SC26-4377.
- *IBM DATABASE 2 Version 2: Messages and Codes*, SC26-4379.
- *IBM DATABASE 2 Version 2: Reference Summary*, SX26-3771.
- *IBM DATABASE 2 Version 2: SQL Reference*, SC26-4380.

DATABASE 2 Version 3

- *IBM DATABASE 2 Version 3: DB2 Administration Guide*, SC26-4888.
- *IBM DATABASE 2 Version 3: DB2 Application Programming and SQL Guide*, SC26-4889.
- *IBM DATABASE 2 Version 3: DB2 Messages and Codes*, SC26-4892.
- *IBM DATABASE 2 Version 3: DB2 Reference Summary*, SX26-3801.
- *IBM DATABASE 2 Version 3: DB2 SQL Reference*, SC26-4890.

ISPF Publication

ISPF Dialog Management Guide and Reference, SC34-4266.

JES Publications

- *MVS/ESA Library Guide with JES2*, SC28-1423.
- *MVS/ESA Library Guide with JES3*, SC28-1424.

MVS/DFP Publications

- *MVS/DFP Version 3 Release 3: Customizing and Operating the Network File System Server*, SC26-4832.
- *MVS/DFP Version 3 Release 3: Macro Instructions for Data Sets*, S26-4747.
- *MVS/DFP Version 3 Release 3: Using Data Sets*, SC26-4749.
- *MVS/DFP Version 3 Release 3: Using the Network File System Server*, SC26-4732.

Network Control Program (NCP) Publications

- *ACF/NCP V7R1 IP Router Planning and Installation Guide*, GG24-3974.
- *NCP and EP Reference*, LY43-0029.
- *NCP, SSP, and EP Generation and Loading Guide*, SC31-6221.
- *NCP, SSP, and EP Resource Definition Guide*, SC31-6223.
- *NCP, SSP, and EP Resource Definition Reference*, SC31-6224.

TME 10 NetView for OS/390 Publications

For a complete listing of TME 10 NetView for OS/390 Publications, refer to the *TME 10 NetView for OS/390 Library Reference,* SC31-8249.

Networking Systems Cross-Product Library

The following list shows books in the Networking Systems cross-product library.

- *Planning Aids: Pre-Installation Planning Checklist for NetView, NCP, and VTAM*, SX75-0092.
- *Planning for Integrated Networks*, SC31-8062.
- *Planning for NetView, NCP, and VTAM*, SC31-8063.

OpenEdition MVS Publications

The following list shows selected books in the OpenEdition MVS library.

- *OS/390 OpenEdition Introduction*, GC28-1889
- *OS/390 OpenEdition Planning*, SC28-1890

Programming Publications

The following list shows books about various programming applications.

- *IBM C/370 Diagnosis Guide and Reference*, LY09-1804 (feature 8082).
- *IBM C/370 General Information Manual*, GC09-1386.
- *IBM C/370 Installation and Customization Guide Version 2 Release 1.0*, GC09-1387.
- *IBM C/370 Programming Guide*, SC09-1384.
- *IBM C/370 Reference Summary*, SX09-1211.
- *IBM C/370 User's Guide*, SC09-1264.
- *OS/390 C/C++ Run-Time Library Reference*, SC28-1663.
- *IBM TSO Extensions CLISTs*, SC28-1876.
- *IBM TSO Extensions Command Language Reference*, GX23-0015.
- *IBM TSO Extensions Interactive Data Transmission Facility: User's Guide*, SC28-1104.
- *IMS/ESA V3R1 Application Programming: DL/I Calls*, SC26-4274.
- *HiPPI User's Guide and Programmer's Reference*, SA23-0369.
- *Parallel I/O Access Methods Programmer's Guide*, SC26-4648.
- *VS Pascal Application Programming Guide*, SC26-4319.
- *VS Pascal Diagnosis Guide and Reference*, LY27-9525.
- *VS Pascal General Information*, GT00-2664.
- *VS Pascal Installation and Customization for MVS*, SC26-4321.
- *VS Pascal Installation and Customization for VM*, SC26-4342.
- *VS Pascal Language Reference*, SC26-4320.

RACF Publications

The following list shows books in the RACF library.

- *IBM Resource Access Control Facility (RACF): General Information Manual*, GT00-2820.
- *IBM Resource Access Control Facility (RACF): User's Guide*, SC28-1341.
- *External Security Interface (RACROUTE) Macro Reference*, GC28-1366.
- *RACF Publications Order Guide*, GX22-0002.
- *Resource Access Control Facility (RACF) Security Administrator's Guide*, SC28-1340.
- *System Programming Library: RACF*, SC28-1343.

SMP/E Publications

The following list shows books in the SMP/E Release 8 library.

- *SMP/E Diagnosis Guide*, SC23-3130.
- *SMP/E Messages and Codes*, SC28-1107.
- *SMP/E Reference*, SC28-1107.
- *SMP/E Reference Summary*, SX22-0006.
- *SMP/E User's Guide*, SC28-1302.

VSAM Publication

MVS/370 VSAM Administration Guide, GC26-4066.

X.25 NPSI Publication

- *X.25 Network Control Program Packet Switching Interface Diagnosis, Customization, and Tuning Version 3*, LY30-5610.
- *X.25 Network Control Program Packet Switching Interface Host Programming*, SC30-3502.
- *X.25 Network Control Program Packet Switching Interface Planning and Installation*, SC30-3470.

IBM Hardware Publications

The following sections describe the books associated with IBM hardware products.

System/370 and System/390 Publications

The following list shows the principles of operation manuals for the System/370 and System/390 processors.

- *IBM ESA/370 Principles of Operation*, SA22-7200.
- *IBM ESA/390 Principles of Operation*, SA22-7201.
- *IBM System/370 Extended Architecture Principles of Operation*, SA22-7085.
- *IBM System/370 Principles of Operation*, GA22-7000.
- *S/360, S/370, and S/390 I/O Interface Channel to Channel Control Unit OEMI*, GA22-6974.

3172 Interconnect Controller Publications

The following list shows books in the IBM 3172 Interconnect Controller library.

- *IBM Interconnect Controller Program User's Guide*, SC30-3525.
- *IBM 3172 Interconnect Controller Installation and Service Guide*, GA27-3861.
- *IBM 3172 Interconnect Controller Operator's Guide*, GA27-3860.
- *IBM 3172 Interconnect Controller Planning Guide*, GA27-3867.
- *IBM 3172 Interconnect Controller Status Codes*, GA27-3951.

3270 Information Display System Publication

3270 Information Display System: 3270 Data Stream Programmer's Reference, GA23-0059.

8232 LAN Channel Station Publications

The following list shows books in the IBM 8232 LAN Channel Station library.

- *IBM LAN Channel Support Program: Version 1.0 User's Guide*, SC30-3458.
- *IBM 8232 LAN Channel Station: Installation and Testing*, GA27-3796.
- *IBM 8232 LAN Channel Station: Operating Guide*, GA27-3785.

9370 Publications

The following list shows books in the 9370 library.

- *IBM 9370 Information System: Using the X.25 Communications Subsystem*, SA09-1742.
- *IBM 9370 Information System X.25 Communications Subsystem Description*, SA09-1743.
- *VM/ESA: Connectivity Planning, Administration, and Operation Release 1*, SC24-5448.

Other TCP/IP-Related Publications

The following sections describe other books associated with TCP/IP.

- *The Art of Distributed Application: Programming Techniques for Remote Procedure Calls*, John R. Corbin, Springer-Verlog, 1991.
- *CAE Specification: X/Open Transport Interface (XTI),* X/Open Company Ltd., U. K., 1992, SC31-8005.
- *IEEE Network Magazine*, July 1990.
- *TCP/IP Illustrated Volume I: The Protocols*, W. Richard Stevens, Addison-Wesley Publishing Company, Inc., 1994, SR28-5586.
- *TCP/IP Illustrated Volume II: The Implementation*, Gary R. Wright and W. Richard Stevens, Addison-Wesley Publishing Company, Inc., 1995, SR28-5630.
- *TCP/IP Illustrated Volume III*, W. Richard Stevens, Addison-Wesley Publishing Company, Inc., 1996, SR23-7289
- *Interoperability Report*, Volume 3, No. 3, March 1989.
- "MIB II Extends SNMP Interoperability," C. Vanderberg, *Data Communications*, October 1990.
- *"Network Management and the Design of SNMP,"* J.D. Case, J.R. Davin, M.S. Fedor, M.L. Schoffstall.

- *“Network Management of TCP/IP Networks: Present and Future,”* A. Ben-Artzi, A. Chandna, V. Warrier.
- *The Simple Book: An Introduction to Management of TCP/IP-based Internets*, Marshall T Rose, Prentice Hall, Englewood Cliffs, New Jersey, 1993.
- “Special Issue: Network Management and Network Security,” *ConneXions-The Interoperability Report*, Volume 4, No. 8, August 1990.
- *UNIX Programmer’s Reference Manual*, (4.3 Berkeley Software Distribution, Virtual VAX-11 Version). Department of Electrical Engineering and Computer Science. University of California, Berkeley, 1988.

OSF/Motif Publications

The following list shows OSF/Motif books.

- *OSF/Motif Application Environment Specifications*, (AES), Open Software Foundation, Prentice Hall, Inc., 1990, ISBN 0-13-640483-9.
- *OSF/Motif Programmer’s Guide*, Open Software Foundation, Prentice Hall, Inc., 1990, ISBN 0-13-640509-6.
- *OSF/Motif Programmer’s Reference*, Open Software Foundation, Prentice Hall, Inc., 1990, ISBN 0-13-640517-7.
- *OSF/Motif Style Guide*, Open Software Foundation, Prentice Hall, Inc., 1990, ISBN 0-13-640491-X.
- *OSF/Motif User’s Guide*, Open Software Foundation, Prentice Hall, Inc., 1990, ISBN 0-13-640525-8.

Sun (RPC) Publications

The following list shows Sun Microsystems books.

- *Networking on the Sun Workstation: Remote Procedure Call Programming Guide*, (800-1324-03), Sun Microsystems, Inc.
- *Network Programming*, (800-1779-10), Sun Microsystems, Inc.

X Window System Publications

The following list shows X Window System books.

- *Introduction to the X Window System*, Oliver Jones, Prentice-Hall, 1988, ISBN 0-13-499997-5.
- *PEXlib Specification and C Language Binding*, Jeff Stevenson, Hewlett-Packard Company, 1992, SR28-5116.
- *The X Window System Series* (6 volumes), O’Reilly & Associates, 1988, 1989, 1990, ISBN 0-937175-40-4, 0-937175-27-7, 0-937175-28-5, 0-937175-35-6, 0-937175-33-1, 0-937175-35-8.
- *X Protocol Reference Manual*, Adrian Nye, ed. O’Reilly & Associates, Inc., 1990, ISBN 0-937175-50-1.
- *X Window System: C Library and Protocol Reference*, Robert Scheifler, James Gettys, and Ron Newman, DEC Press, 1988, ISBN 1-55558-012-2.
- *X Window System: Programming and Applications with Xt*, Douglas A. Young, Prentice-Hall, 1989, ISBN 0-13-972167-3.
- *X Window System: Programming and Applications with Xt, OSF/Motif Edition*, Douglas A. Young, Prentice-Hall, 1990, ISBN 0-13-497074-8.
- *X Window System Technical Reference*, Steven Mikes, Addison-Wesley, 1990, ISBN 0-201-52370-1.
- *X Window System User’s Guide*, Valerie Quercia and Tim O’Reilly, O’Reilly & Associates, Inc., 1990, ISBN 0-937175-14-5.

Network Architecture Publications

The following sections list books associated with network architecture.

Open Systems Interconnection (OSI) Publication

Open Systems Interconnection, Z320-9757.

Systems Network Architecture (SNA) Publications

The following list shows books in the SNA library.

- *Systems Network Architecture: Sessions between Logical Units*, GC20-1868.
- *Systems Network Architecture Format and Protocol Reference Manual: Architecture Logic*, SC30-3112.
- *Systems Network Architecture Format and Protocol Reference Manual: Management Services*, SC30-3346.
- *Systems Network Architecture Formats*, GA27-3136.
- *Systems Network Architecture Network Product Formats*, LY43-0081.

Index

Communicating Your Comments to IBM

OS/390 TCP/IP OpenEdition
Planning and Release Guide

Publication No. SC31-8303-00

If you especially like or dislike anything about this book, please use one of the methods listed below to send your comments to IBM. Whichever method you choose, make sure you send your name, address, and telephone number if you would like a reply.

Feel free to comment on specific errors or omissions, accuracy, organization, subject matter, or completeness of this book. However, the comments you send should pertain to only the information in this manual and the way in which the information is presented. To request additional publications, or to ask questions or make comments about the functions of IBM products or systems, you should talk to your IBM representative or to your IBM authorized remarketer.

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.

If you are mailing a readers' comment form (RCF) from a country other than the United States, you can give the RCF to the local IBM branch office or IBM representative for postage-paid mailing.

- If you prefer to send comments by mail, use the RCF at the back of this book.
- If you prefer to send comments by FAX, use this number:

  United States and Canada: **1-800-227-5088**
- If you prefer to send comments electronically, use this network ID:
  - IBM Mail Exchange: **USIB2HPD at IBMMAIL**
  - IBMLink: **CIBMORCF at RALVM13**
  - Internet: **USIB2HPD@VNET.IBM.COM**

Make sure to include the following in your note:

- Title and publication number of this book
- Page number or topic to which your comment applies.

Help us help you!

OS/390 TCP/IP OpenEdition
Planning and Release Guide

Publication No. SC31-8303-00

If your concern is service related, you can reach Service at 1-800-992-4777 in the United States. Outside the United States, please check your phone listing for the IBM Service Center nearest you.

We hope you find this publication useful, readable and technically accurate, but only you can tell us! Please take a few minutes to let us know what you think by completing this form.

Overall, how satisfied are you with the information in this book?	Satisfied	Dissatisfied
	□	□

How satisfied are you that the information in this book is:	Satisfied	Dissatisfied
Accurate	□	□
Complete	□	□
Easy to find	□	□
Easy to understand	□	□
Well organized	□	□
Applicable to your task	□	□

Specific Comments or Problems:

Please tell us how we can improve this book:

Thank you for your response. When you send information to IBM, you grant IBM the right to use or distribute the information without incurring any obligation to you. You of course retain the right to use the information in any way you choose.

Your Internet Address: ____________________

Name

Address

Company or Organization

Phone No.

Help us help you!
SC31-8303-00

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

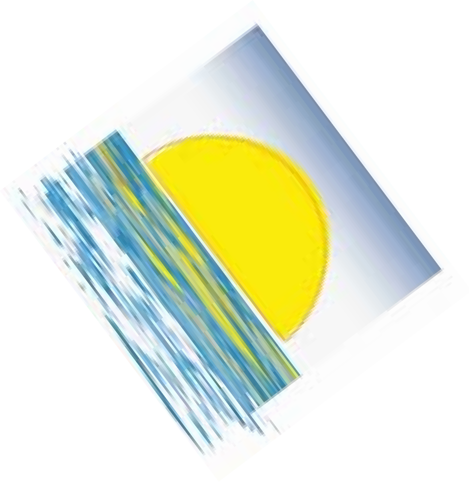
Information Development
Department CGMD
International Business Machines Corporation
PO BOX 12195
RESEARCH TRIANGLE PARK NC 27709-9990

Fold and Tape **Please do not staple** Fold and Tape

Cut or Fold
Along Line

File Number: S390-50
Program Number: 5645-001

Printed in the United States of America
on recycled paper containing 10%
recovered post-consumer fiber.



SC31-8303-00