

Network/IT™

TCPaccess 5.3

Release Notes

This documentation and related computer software program (hereinafter referred to as the "Documentation") is for the end user's informational purposes only and is subject to change or withdrawal by Computer Associates International, Inc. ("CA") at any time.

THIS DOCUMENTATION MAY NOT BE COPIED, TRANSFERRED, REPRODUCED, DISCLOSED OR DUPLICATED, IN WHOLE OR IN PART, WITHOUT THE PRIOR WRITTEN CONSENT OF CA. THIS DOCUMENTATION IS PROPRIETARY INFORMATION OF CA AND PROTECTED BY THE COPYRIGHT LAWS OF THE UNITED STATES AND INTERNATIONAL TREATIES.

TO THE EXTENT PERMITTED BY APPLICABLE LAW, CA PROVIDES THIS DOCUMENTATION "AS IS" WITHOUT WARRANTY OF ANY KIND, INCLUDING WITHOUT LIMITATION, ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON INFRINGEMENT. IN NO EVENT WILL CA BE LIABLE TO THE END USER OR ANY THIRD PARTY FOR ANY LOSS OR DAMAGE, DIRECT OR INDIRECT, FROM THE USE OF THIS DOCUMENTATION, INCLUDING WITHOUT LIMITATION, LOST PROFITS, BUSINESS INTERRUPTION, GOODWILL OR LOST DATA, EVEN IF CA IS EXPRESSLY ADVISED OF SUCH LOSS OR DAMAGE.

THE USE OF ANY PRODUCT REFERENCED IN THIS DOCUMENTATION AND THIS DOCUMENTATION IS GOVERNED BY THE END USER'S APPLICABLE LICENSE AGREEMENT.

The manufacturer of this documentation is Computer Associates International, Inc.

Provided with "Restricted Rights" as set forth in 48 C.F.R. Section 12.212, 48 C.F.R Sections 52.227-19(c)(1) and (2) or DFARS Sections 252.227.7013(c)(1)(ii) or applicable successor provisions.

© 2000 Computer Associates International, Inc. One Computer Associates Plaza, Islandia, New York 11788-7000. All rights reserved.

All product names referenced herein belong to their respective companies.

Network/IT™ TCPaccess 5.3 Release Notes

Table of Contents

Chapter 1 NetworkIT™ TCPaccess 5.3 Release Notes

New Features	1 - 2
TN3270E TELNET Server	1 - 2
Features of the TN3270E Server:	1 - 2
Configuring the TN3270E Server	1 - 2
Open Edition Telnet Server with Timer Daemon support	1 - 2
Configuration Wizard	1 - 3
Channel to Channel Driver Support	1 - 3
SMF Exit Point	1 - 3
Product Documentation	1 - 4
Reading the Documents with a Web Browser	1 - 4
Reading the Documents with Adobe Acrobat Reader	1 - 4
TCPaccess Documentation Library	1 - 5
Installation Pre-requisites	1 - 6
MVS Release Level	1 - 6
APF Authorizations	1 - 6
Link List Data Sets	1 - 6
Automated Operations Impact	1 - 6
Callable System Services Library and Language Environment	1 - 7
TSO/E	1 - 7
ACF/2	1 - 7
SAS/C Release Level	1 - 7
Common Inet Sockets in OpenEdition	1 - 7
Asynchronous Socket Support	1 - 8
RS6000	1 - 8
VTAM Release 3.3	1 - 8
MVS/ESA Release 5.1 Considerations	1 - 8
Novell LAN Workplace	1 - 8
Walker, Richer, & Quinn Software	1 - 8
Hardware Requirements	1 - 9
IP Hardware, Cisco IOS Software, and CIP Microcode Compatibility	1 - 9

Installing TCPAccess 5.3	1 - 10
FMID Description	1 - 10
Sample JCL to Unload the Tape	1 - 11
Installation Steps for TCPAccess 5.3	1 - 12
Usermods - Important Information	1 - 13
Usermods for 'FTP' Alias	1 - 13
Usermod UMFTP1A	1 - 13
Usermod UMFTP3A	1 - 13
Usermod UMOTELN	1 - 14
USERMOD Customization	1 - 14
Customization	1 - 15
Customization using the Wizard	1 - 15
Converting from TCPAccess 4.1 to TCPAccess 5.3	1 - 15
Starting TCPAccess 5.3	1 - 16
Diagnosis and Problem Reporting	1 - 18
Obtaining GTF Output	1 - 18
Obtaining a SNAP Dump	1 - 18
Obtaining a TCPEEP	1 - 18
Obtaining an SVC Dump	1 - 18
Obtaining JCL Output	1 - 18
SAS/C Usage Notes	1 - 19

Chapter 2 TCPAccess 5.3 Installation Verification Procedures

Executing RUNTCP as a Started Task	2 - 2
Sample Run JCL	2 - 2
Testing TCPAccess in Loopback Mode	2 - 4
Stopping the RUNTCP Started Task	2 - 4
Verifying the API Installation	2 - 5
Compiling and Testing Sample C Socket Applications	2 - 5
Compiling and Testing Sample C Basic Applications	2 - 5
Testing TTCP TSO Command Processor	2 - 5
Running the OpenEdition IVPs	2 - 6
Preparing to Run the IVPs	2 - 6
Running the IVPs in Batch Mode	2 - 6
Running the OpenEdition IVPs in the OMVS Shell	2 - 7

Chapter 3 Converting from TCPaccess 4.1 to TCPaccess 5.3

ACPCFG00 HOST Statement	3 - 2
Application Statement Mappings	3 - 4
TCP Protocol Parameters	3 - 5
UDP Protocol Parameters	3 - 7
RAW Protocol Parameters	3 - 8
NETWORK Parameters	3 - 9
ROUTE Parameters	3 - 10
Multiple Default ROUTE Statements	3 - 11
Driver Configuration	3 - 12
CETI Driver	3 - 12
LNICLAW	3 - 13
LNICDLC	3 - 14
LNIHYPR	3 - 15
LNILCS and LNILINK	3 - 16
IJTCFG00 Changes	3 - 17
LOGGING Statement	3 - 18

1

Network/IT™ TCPAccess 5.3 Release Notes

August, 2000

These release notes provide information for system administrators and installation managers who are responsible for installing and continuing support of the TCPAccess product. This document is not a user or administration guide; it details only information specifically related to TCPAccess 5.3.

Note: These release notes were current as of the printing of the Documentation CD.
Release notes are distributed in print version with your product and may include updates not available when the CD was created.

This document contains these sections:

- [“New Features” on page 1-2](#)
Provides a list of feature enhancements in this release of TCPAccess.
- [“Product Documentation” on page 1-4](#)
Describes the documentation for this release of TCPAccess.
- [“Installation Pre-requisites” on page 1-6](#)
Describes the environment needed prior to installing TCPAccess.
- [“Installing TCPAccess 5.3” on page 1-10](#)
Provides information on installing TCPAccess 5.3.
- [“Usermods - Important Information” on page 1-13](#)
Provides important new information about FTP aliases.
- [“Customization” on page 1-15](#)
Introduces TCPAccess customization.
- [“Starting TCPAccess 5.3” on page 1-16](#)
Describes how to start your newly installed TCPAccess product.
- [“Diagnosis and Problem Reporting” on page 1-18](#)
Describes how to use the diagnostic tools and how to report problems.
- [“SAS/C Usage Notes” on page 1-19](#)
Provides important information on the use of the SAS/C compiler.

New Features

This section describes the new product features of TCPAccess 5.3.

TN3270E TELNET Server

The TN3270E Telnet Server is a new telnet server, which provides full support for the TN3270E protocol, as defined by RFC2355 and extensions, including support for 3287 printers. The TN3270E server also supports "old-style" TN3270 connections for clients which do not support TN3270E, and can be configured to negotiate TN3270 by default.

Features of the TN3270E Server:

- All-new server, designed to be highly efficient and scalable. Benchmark tests show the TN3270E server uses 78% less virtual storage per TN3270 session than the old telnet server.
 - Supports the full TN3270E protocol (as defined in RFC2355 and its extensions), including TN3270E printer support.
 - Response-time metrics are captured to a new data space. A new NETSTAT command, TELNET, can be used to query the response-time data on a session level. See the *System Management Guide* for information on using NETSTAT.
 - SMF record subtype 23 is expanded for the TN3270E server to include more information, including response time figures. See the *System Management Guide* for details.
 - A new user exit point, VTAMBIND, is added with the TN3270E server, to allow a user exit program to examine the BIND image from a PLU and (optionally) reject a session. For details, read the *TCPAccess Planning Guide*. See the *TCPAccess Customization Guide* for information on configuring user exit programs.
-

Configuring the TN3270E Server

To use the TN3270E server, change the MODULE parameter of the SERVICE statement for the server telnet port to specify T01S3270. New parameters are added to the TELNET statement, to define the telnet protocol to be negotiated, and the TN3270E functions to be permitted. A new parameter is added to the TERMPROF statement to allow specification of logmodenames for TN3270E device types. The LUPOOL statement is enhanced to allow association of TN3270E printer LUs with their terminal LU counterparts. See the *TCPAccess Customization Guide* for details.

Open Edition Telnet Server with Timer Daemon support

This TCPAccess OpenEdition Telnet daemon is designed to interface with Open Edition or OS/390 UNIX applications on the mainframe. The T016TIDP Timer Daemon program provides a site-independent, machine-readable date and time. The use of time-servers makes it possible to quickly confirm or correct a system's time, by making a brief poll of several independent sites on the network.

Configuration Wizard

The Configuration Wizard allows you to set up all the configuration files that you will need to customize TCPAccess for your site and automatically generates a complete member set for your version of TCPAccess.

Channel to Channel Driver Support

Support for a CTC (channel-to-channel) interface has been added to TCPAccess 5.3.

SMF Exit Point

A new SMF exit point is defined to allow a user exit program to be called when an SMF record is about to be written by TCPAccess.

Product Documentation

Documentation provided with the TCPaccess product consists of a printed manual set, printed release notes, and a Compact Disc (CD). The documentation on the CD is provided in two formats:

- HTML, for use with a Web Browser
- PDF, for use with the Adobe Acrobat PDF Reader, version 3.0 or higher.

Note: Release Notes are included on the CD. Be sure to check the printed release notes that you received with the product as they may have been updated since the CD was created.

Reading the Documents with a Web Browser

For best viewing, we recommend that you use either Netscape Version 4.0 and above or Internet Explorer 4.0 and above.

From the browser, click on “File” and select “Open” (or “Open File”) and select the file named **index.htm**. That file is a menu with links to all the manuals. You can click on the manual name to bring up the manual and its table of contents. The menu also includes links to the PDF formatted manuals (noted with the Adobe Acrobat symbol). You can view and/or print the PDF files from your browser, but you will not be able to use the Acrobat full manual search index from the browser.

Reading the Documents with Adobe Acrobat Reader

If you do not have the Adobe Acrobat Reader, you can download it free from <http://www.adobe.com>. Be sure to select the option “Include option for searching PDF files”. This will allow you to perform searches over the entire TCPaccess manual set with the search index provided (tcp53ndx.pdx).

From the Acrobat reader, select “File”, then “Open” and select the file named **manuals.pdf**. This file contains a menu of the PDF files for the TCPaccess manual set. The search index is automatically linked for these files. Alternatively, you can open each file individually. PDF files are in the directory pdfs/tcp53. The following table lists each file name and its corresponding manual title:

PDF filename	Manual Title
ac.pdf	TCPaccess Assembler API Concepts
am.pdf	TCPaccess Assembler API Macro Reference
cg.pdf	TCPaccess Customization Guide
mp.pdf	TCPaccess Prefixed Messages
mu.pdf	TCPaccess Unprefixed Messages and Codes
pl.pdf	TCPaccess Planning Guide
rp.pdf	TCPaccess RPC/XDR Programmer’s Reference
sk.pdf	TCPaccess C/Socket Programmer’s Reference
sm.pdf	TCPaccess System Management Guide
tcpn.pdf	TCPaccess 5.3 Release Notes
ug.pdf	TCPaccess User’s Guide

TCPaccess Documentation Library

The complete documentation library for TCPaccess is listed below. Future updates will be detailed in technical notes that accompany the product.

- ***NetworkIT TCPaccess Planning Guide***

Describes the TCPaccess architecture, how to prepare for installation, customize system security, and how to use OpenEdition and user exits. It also includes information about editing tools, information to help configure TCPaccess for Cisco routers and plan storage usage, and about the CDLC driver. This manual is targeted for Computer Operations personnel.

- ***NetworkIT TCPaccess Customization Guide***

Complete guide to customizing TCPaccess for your site.

- ***NetworkIT System Management Guide***

Describes the operator commands (startup and shutdown of TCPaccess), how to use the System Management Facility (SMF), diagnostic procedures, diagnostic commands, and Inter-User Communications Vehicle (IUCV) sockets.

- ***NetworkIT TCPaccess User's Guide***

Guide to the TCPaccess Client FTP2, Client FTP3, Client FTP, Server FTP, TN3270E, Telnet, and Mail commands, USPOOL, remote executor, API and socket applications, and programmable batch interface for Client FTP2 and Client FTP3.

- ***NetworkIT TCPaccess Prefixed Messages***

Describes all messages that have an alphanumeric prefix and are printed to the console or the TCPaccess log files.

- ***NetworkIT TCPaccess Unprefixed Messages and Codes***

Describes messages from FTP, Telnet, Mail, and general API return codes.

- ***NetworkIT TCPaccess Assembler API Concepts***

Describes the concepts, flow, and operation of the TCPaccess API.

- ***NetworkIT TCPaccess Assembler API Macro Reference***

Guide to the Assembler API macros.

- ***NetworkIT TCPaccess C/Socket Programmer's Reference***

Describes calls for the basic C library and for the Socket API. This manual contains a chapter describing the OpenEdition (UNIX System Services) socket interface.

- ***NetworkIT TCPaccess RPC/XDR Programmer's Reference***

Describes the RPC and XDR API calls.

Installation Pre-requisites

This section details the information you will need before you install TCPAccess Version 5.3.

MVS Release Level

TCPAccess 5.3 requires MVS/ESA Release 5.1 or higher, including all release levels of OS/390. MVS/SP releases are not supported. OpenEdition support is available only at MVS/ESA Releases 5.x and above.

The MVS PTF associated with APAR **OW30322** must be applied prior to starting TCPAccess 5.3.

APF Authorizations

The PFSLOAD library is used with OpenEdition and contains Physical File System (PFS) modules. It must be APF-authorized to work correctly.

If you use the SASLINK data set as described in the ALLOCT00 section of these instructions you must also APF-authorize that library.

Link List Data Sets

TCPLOAD must *never* be in the link list.

Automated Operations Impact

As of TCPAccess Release 5.2 and later, the message

```
ACC210I Internet Protocol Task Started
```

no longer displays to indicate that startup has completed. A message will display indicating startup completion in TCPAccess 5.3. The message number will vary, depending on the device you are using. The message text will be:

```
Media media_name is now operational with one or more active interfaces.
```

Note: An additional WTOR message appears when a second P or P clear is issued.

Callable System Services Library and Language Environment

You must have the Callable System Services library, SYS1.CSSLIB, available for the installation. Modules from this library are linked with TCPaccess 5.3 for OpenEdition support.

IBM's Language Environment link-time library, SCEELKED, is now also a required SMP/E data set used by the CALLLIB facility.

CAUTION ! TCPaccess will not install properly without these libraries.

TSO/E

New users must verify that the IBM program product TSO/E is installed. TCPaccess configuration files are parsed using TSO/E parsing routines and S0C4 ABENDs can result if these routines are not available.

ACF/2

For some users of CA-ACF2, the CA-ACF2 fixes listed here affect the way the TCPaccess SAF security interface to CA-ACF2 works. These fixes should be reviewed by the CA-ACF2 Systems Programmer to determine if they are still applicable or have been superseded by later maintenance.

Table 1-1 CA-ACF2 Fixes

ACF2 R5.2 Fixes	ACF2 R6.0 Fixes
TW87092	G016544
TW95626	G025907
TW95673	G023824
CO96846	G026166
CO97436	G026702
	G031029
	TA1294B

SAS/C Release Level

TCPaccess 5.3 is designed to work with SAS/C Release 6.5, which is included with the product. If you are not running with the correct SAS/C libraries, unpredictable results may occur.

Common Inet Sockets in OpenEdition

You must install the APAR **OW20620** (for OS/390 R1.3 only) in order to use Common Inet sockets in OpenEdition.

Asynchronous Socket Support

TCPAccess 5.3 supports asynchronous socket calls from OpenEdition (Unix System Services). The following system levels are required to run this option:

- OS/390 V1.3 requires PTF level 9705 or higher
 - DB2 V5 requires PTF level 9708 or higher
 - DB2/CONNECT V5 requires Fix Pack WR09014. **Do not apply Fix Pack WR09024.**
-

RS6000

You must install the RS/6000 patch **U432867** to use the CLAW driver. With earlier RS/6000 maintenance, MVS error recovery problems can occur.

VTAM Release 3.3

VTAM Release 3.3 users must apply PTF **UY67100** or **UY69746**, or their equivalent. If this is not possible, contact Customer Support and request APAR **MB18602**. If the PTFs or APAR are not applied, problems can occur handling control vector X'5F' and cause connection errors.

MVS/ESA Release 5.1 Considerations

Interlink 3722 users must install the latest release of the CETI microcode. Interlink 3762 users must install the latest release of the ECA CETI microcode. The CETI driver may terminate at startup if you are using earlier releases.

Novell LAN Workplace

It is recommended that you apply Novell fixes **LWP41N** and **LWP168** (to correct an FTP hang condition). These fixes are available on NetWire.

Walker, Richer, & Quinn Software

If you are using software from Walker, Richer, & Quinn, it is recommended that you install version 2.1.2 or above of Reflection 8+ (for DOS using IBM 3270) and Version 2.2 of WRQNET TCP/IP. Users of earlier releases can experience a tn3270 hang condition.

Hardware Requirements

These minimum hardware levels are required for TCPaccess 5.3, including TCPaccess Fault Tolerant:

Interlink 3762 ECA CETI 3.0 Checksum B030

Interlink 3762 TCA 4.0 Checksum 5841

Interlink 3762 FCA 2.5 Checksum 6273

Interlink 3722 ELC2 CETI 2.2 Checksum 67FE

Interlink 3722 ELC2 8232 3.1 Checksum 4132

IBM 2216 MAS V1 R1.1

IBM 3172 Model 3 ICP 3.3

IBM 3172 Model 1 ICP 1.1 and higher (ICP 1.1 with patch PLXXX48)

NSC HYPERchannel DX CDA 7.2

IP Hardware, Cisco IOS Software, and CIP Microcode Compatibility

Note: We recommend that you refer to the CCO service and your Systems Engineer to choose the most appropriate level for your configuration.

Table 1-2 Hardware/Software/Microcode Compatibility

CIP Hardware Version	Minimum Cisco IOS Release Required	Minimum CIP Microcode Version Recommended
CIP 4.4 or later	11.0(14a)	cip21-14
CIP 2 5.x or later	11.0(13a)BT	cip22-17
	11.1(9)	cip22-14
	11.2(4)	cip22-14
	11.2(9)BC	cip24-1
	11.3(0.6)	cip25-2
	11.3(3)T	cip26-0
CPA	11.3(3)T	xcpa26-0

Note: For the CLAW PACKING feature, you must use IOS level 11.2(14)BC and CIP microcode version cip24-6 and above.

Installing TCPAccess 5.3

Use these instructions to install TCPAccess 5.3 and to upgrade from a previous release of TCPAccess.

Note: There is no CNTL library for TCPAccess 5.3. All members from that library have been moved to the SAMP data set.

Note: In order to avoid a potential CSA overlay problem, **do not** use the same subsystem ID for TCPAccess 4.1 and TCPAccess 5.3 stack address spaces. To determine which subsystems are in use on your system, execute the MVS command **D SSI**.

FMID Description

The FMID's for TCPAccess 5.3 have been renamed. The FMID's are summarized in the table below.

Table 1-3 TCPAccess 5.3 FMID Description

FMID	Function	Description
TCP5300	Base function	Infrastructure
WCP530S	Dependent function	Support
WCP530A	Dependent function	Applications
WCP530D	Dependent function	Development
WCP530K	Dependent function	TCP/IP Stack
WCP530T	Dependent function	Trace
WCP530X	Dependent function	Translate Tables
WCP650C	Dependent function	SAS

Sample JCL to Unload the Tape

Copy and execute the JCL below to unload the control file from which you will be able to install and customize TCPAccess 5.3. This control file, INSTJCL, is on your TCPAccess 5.3 base tape.

```
//UNLDTCP JOB (TCPAccess),'UNLOAD TCP SAMP',MSGCLASS=X
//*
//UNLOAD EXEC PGM=IEBCOPY
//INDD DD DISP=SHR,DSN=INSTJCL,VOL=SER=vol_ser,
// LABEL=(1,SL,EXPDT=98000),UNIT=UNITNAME
//OUTDD DD DSN=trgindx.CNTL,DISP=(NEW,CATLG,DELETE),
// VOL=SER=trgvol,SPACE=(TRK,(30,2,25)),UNIT=trgunit,
// DCB=(DSORG=PO,RECFM=FB,LRECL=80,BLKSIZE=6160)
//SYSPRINT DD SYSOUT=holdcl
//SYSIN DD *
COPY INDD=((INDD,R)),OUTDD=OUTDD
```

Installation Steps for TCPAccess 5.3

Member names referenced in the following instructions are located in the CNTL file that you loaded onto your system in the JCL above, "[Sample JCL to Unload the Tape](#)" on page 1-11.

1. You will need to edit the data set name symbolics to be consistent with the naming conventions of your site.

Note: You can edit the data set names in the following steps manually or you can use TCPNAMES, an edit macro supplied with TCPAccess. To use TCPNAMES, copy the TCPNAMES member to a data set listed in the SYSPROC concatenation of your TSO logon procedure. You will also need to create a jobcard (named JOBCARD) for TCPNAMES to use.

2. Verify that there is adequate space on the volume you've specified. TCPAccess 5.3 requires 250 cylinders of DASD. The LNKINDX data set is allocated as a SYS1 data set in job ALLOCT00.
3. Use ALLOCSMP to allocate the data sets for the TCPAccess SMP/E install.
TCPAccess 5.3 must be installed into a new CSI. Do **not** install over an existing release.
4. Edit ALLOCT00 to allocate the TCPAccess target and distribution libraries.
5. Edit INSTSMPE to install the base product.
SMP/E requires an 8 Meg region to install TCPAccess.
If you are using a tape management system such as CA1, you must modify the label parameter on your DD statements to include EXPDT=98000.

```
LABEL=( 2 , NL , , EXPDT=98000 )
```
6. Verify HOLDDATA.
Before installing TCPAccess 5.3 maintenance, verify that you have the most recent HOLDDATA file. Use this HOLDDATA file as your SMPHOLD file in the SMPHOLD DD statement. You can find the HOLDDATA and the latest PTFs on <http://www.cai.com> and follow the links for Interlink/Sterling support. Otherwise, you can contact Customer Support for the latest HOLDDATA file.
7. Use SMPECUMT to apply maintenance to the base product.
The maintenance tape (VOL=SER=PTFTAP) distributed with the product contains the latest maintenance for the TCPAccess product. Apply this maintenance prior to beginning any customization, as configuration files or parameters may have changed. You should also check the web site <http://www.cai.com> for any updated PTFs not included on the tape.
8. Before submitting the INSTSMPE member, consider the following:
You may submit only the RECEIVE portion of this job first. This will let you review the HOLDDATA, especially any hold action that may appear, and take appropriate action. It will also let you add additional BYPASS HOLD keywords to your APPLY statement.

Usermods - Important Information

CAUTION ! The EXECutable FTP is no longer supplied as a default with this release of TCPAccess. FTP aliases are no longer supplied and the FTPLOAD library is not created. You **MUST** use one of the usermods described below to have FTP as an EXECutable.

Usermods for 'FTP' Alias

Previous releases of TCPAccess supplied an alias of 'FTP' for FTP1 in the LINK library and an alias for 'FTP' for FTP3 in the FTPLOAD library. With the 5.3 release, no aliases are supplied and the FTPLOAD library is not created.

You can now choose which FTP to alias (FTP1 or FTP3, or both) and link them into a library of your choice. By using the supplied usermods, UMFTP1A and UMFTP3A, the load module and its alias will remain SMP/E maintainable through supplied maintenance.

You **must** choose either the UMFTP1A or UMFTP3A usermods described below.

Usermod UMFTP1A

Usermod UMFTP1A, located in the SAMP library, will assign an alias of 'FTP' to FTP1. This usermod defaults to linking the alias 'FTP' into the TCPLINK library as in previous releases.

Update the usermod with the current prerequisite for the module name in the '++MOD' statement. If you need to apply future maintenance to FTP1, this usermod will have to be restored for the PTF(s) to be applied. It can then be applied to assign the 'FTP' alias.

Usermod UMFTP3A

Usermod UMFTP3A, located in the SAMP library, will assign an alias of 'FTP' to FTP3. This usermod defaults to linking the alias 'FTP' into the FTPLOAD library as in previous releases.

To install this usermod, you will need to either create a new load library, or use an existing load library of your choice, and add a DDDEF to SMP/E defining this library using the DDDEF name PTFLOAD. You **cannot** install both UMFTP1A and UMFTP3A into the TCPLINK library.

Update the usermod with the current prerequisite for the module name in the '++MOD' statement. If you need to apply future maintenance to FTP3, this usermod will have to be restored for the PTF(s) to be applied, and it can then be applied again to assign the 'FTP' alias.

CAUTION ! The FTP usermods **must be applied separately – they cannot be applied at the same time.** Since CALLLIBS is in effect, the modules are first linked out to SMPLTS and then to the target library. If the usermods are applied together, one 'FTP' alias will overlay the other in the SMPLTS. This results in one 'FTP' alias for the last module linked.

Usermod UMOTELN

Use the UMOTELN usermod only if you want to run OTELNET from HFS and not PFSLOAD. You will need to supply a DDDEF using the 'PATH' parameter for the HFS library. You may need a 'PRE' for the module T016TNDP. If future maintenance is applied to this module, UMOTELN will have to be restored and re-applied. After UMOTELN is applied, the permission bits must be set correctly. To set permissions correctly, issue the following command:

```
chmod 111 /your/path/T016TNDP
```

Note that the module name is uppercase.

USERMOD Customization

The statements described in this section are all described in the *TCPaccess Customization Guide*. These members all ship with the product.

UMODTRAN: Read the section "Maintaining Translation Tables Using UMODTRAN" in the "Telnet and tn3270 Configuration Chapter" of the *TCPaccess Customization Guide*.

UMODAPPL: This is the USERMOD that you will most likely apply. In order for UMODAPPL to run, you must modify member APPLNAME in the SAMP library with your correct VTAM APPL.

UMODPOOL: UMODPOOL uses member ACCPOOL in the SAMP library. Old style 3174 controllers did not let things start with 01, so the starting name is ACSVLT02.

Customization

You will need to customize the configuration files for TCPAccess for your network. Refer to the ***TCPAccess Customization Guide*** for complete information.

Customization using the Wizard

We recommend that you use the new Configuration Wizard when installing the TCPAccess product for the first time, or when converting from TCPAccess 4.1. For more information about the Wizard, refer to the ***TCPAccess Customization Guide***.

Converting from TCPAccess 4.1 to TCPAccess 5.3

If you are upgrading from TCPAccess 4.1, refer to the chapter ["Converting from TCPAccess 4.1 to TCPAccess 5.3"](#) in these Release Notes.

Starting TCPaccess 5.3

The startup member, RUNTCP, is located in the SAMP library. This member will invoke TCPaccess 5.3. You will have a choice to submit RUNTCP as either a started task or as a batch job.

If you will be using OpenEdition, be sure that the PFSLOAD library is allocated before submitting the RUNTCP member. The PFSLOAD library contains Physical File System (PFS) modules needed by OpenEdition. It must be APF-authorized to work as specified in the section [“APF Authorizations” on page 1-6](#) earlier in this document. If you are not using OpenEdition, the library is not required.

Note: TCPLOAD must *never* be in the linklist.

Edit the RUNTCP member.

- Do *not* execute TCPNAMES.
- Change the symbolic 'TRGINDX' to the value you specified in member TCPNAMES.
- Verify that the symbolic SSN=ACSS is available for use. The MVS command **D SSI** will display subsystem names that have been invoked during the life of the current IPL. You can use ACSS if it either does not display or if it shows as inactive. If ACSS is active, you must choose a different name for your subsystem.
- Verify that the symbolic 'SRC=%' (command character) is not being used by another application.
- If you are using a user VTAMLIB, you must add it to your STEPLIB concatenation to make sure RUNTCP can find your USSTAB.
- If you have a line limit, add one of the following lines immediately after the JOBCARD to support diagnostics.

- If you are using JES2 add:

```
/*JOBPARM LINES=9999
```

- If you are using JES3, add:

```
//*MAIN LINES=(999,W)
```

- To prepare RUNTCP as a batch job:
 - Complete your RUNTCP JOBCARD and add the following JCL statements to the end of the RUNTCP JCL stream:

```
// PEND
//label EXEC TCPIP
```

- To prepare RUNTCP as a started task:
 - Delete the RUNTCP JOBCARD and comments that appear prior to the PROC statement.
 - Copy member RUNTCP into a system PROCLIB.

Submit RUNTCP.

- To submit RUNTCP as a batch job, submit member RUNTCP.
- To submit RUNTCP as a started task, issue the MVS command **S RUNTCP**.
- The message ACC210I Internet Protocol Task Started no longer displays to indicate that startup has completed. A message will display indicating startup completion in TCPaccess 5.3. The message number will vary, depending on the device you are using. The message text will be

Media *media_name* is now operational with one or more active interfaces.

To stop RUNTCP, issue the MVS command **F RUNTCP,P CLEAR** and answer YES to the reply. This reply is generated if you have the PROMPT parameter specified in your **IJTCFGxx** member. For more information on stopping and starting TCPAccess, refer to the TCPAccess System Management Guide.

You can use RUNIUCV to start IUCV as a separate started task. You don't need RUNIUCV if you will be using IUCV in the TCPAccess address space. If RUNIUCV is to be stopped, it must be done after RUNTCP has terminated.

Diagnosis and Problem Reporting

Generally, Customer Support needs the following documentation to help diagnose problems thoroughly:

- TCP SNAP
- TCPEEP
- SVC dump of the TCPaccess and other related address spaces
- JCL output of the TCPaccess job
- Case record from Customer Support

Refer to the *TCPaccess System Management Guide* for a complete description of diagnostic tools.

Obtaining GTF Output

The “Command Scripts” section of Chapter 2, “TCPaccess Operation” in the *TCPaccess System Management Guide* provides a sample command script to start GTF and turn on tracing.

Obtaining a SNAP Dump

As soon as a problem occurs, enter the following MVS command to your TCPaccess address space:

```
F RUNTCP,TCP SNAP ALL
```

Obtaining a TCPEEP

Read the *TCPaccess System Management* for information to help you set up and submit your trace address space and submit the TCPEEP TSO command.

Obtaining an SVC Dump

Read the IBM document *MVS/ESA System Commands Reference Summary* for the syntax of the required **DUMP** commands. In particular, include all jobs involved in the problem using the JOBNAME parameter of the **DUMP** command.

Obtaining JCL Output

Copy the JCL output of RUNTCP to a file using the following procedure:

- If you are using JES2/SDSF, type **XDC** beside the job listed in SDSF and follow the panel instructions. This will let you copy to a data set of your choice. Generally, for every thousand lines of job output you need three 3390 tracks.
 - If you are using JES3, you may use FLASHER or a similar product to copy the RUNTCP output to a data set in a similar fashion.
-

SAS/C Usage Notes

You **must** use the SAS 6.5 compiler, header files, object files, and runtime libraries to COMPILE and LINK using the TCPaccess 5.3 socket library.

You can run a down-level (SAS 5.5) C program against TCPaccess 5.3, and it can be compiled and linked using the TCPaccess 4.1 (or lower) socket library. However, it **must** be run using the SAS 5.5 runtime library. This program can then run over the TCPaccess 5.3 stack.

- Mixing any SAS 6.5 modules/headers with SAS 5.5 modules/headers will cause problems.
- Mixing any TCPaccess 5.3 socket modules/headers with TCPaccess 4.1 socket modules/headers will cause problems.
- Mixing runtime modules between releases will cause problems.

In addition, in SAS 6.5, LSCNCOM is always dynamically loaded. If linked with the application program in 6.5, LSCNCOM will cause ABENDs.

2

TCPaccess 5.3 Installation Verification Procedures

This section includes the Installation Verification Procedures (IVP) that you should run to ensure that the TCPaccess software installed correctly.

Sections include:

- [“Executing RUNTCP as a Started Task” on page 2-2](#)
- [“Testing TCPaccess in Loopback Mode” on page 2-4](#)
- [“Verifying the API Installation” on page 2-5](#)
- [“Running the OpenEdition IVPs” on page 2-6](#)

Executing RUNTCP as a Started Task

No network interface device is required to execute in loopback mode. Data is not written to or read from the channel. Outbound data is copied into the input buffers by the Local Network Interface (LNI). You can establish connections only to your own host when operating in software loopback mode.

1. Using SAMP member RUNTCP as a model, create a started task JCL procedure in a JCL procedure library at your site.
2. Substitute appropriate values for the TRGINDX, SSN, SRC, SOUT, and CMND parameters.
3. Issue an MVS START command to start the RUNTCP started task.
4. From an MVS console, issue this MVS command:

S RUNTCP

Sample Run JCL

Here is sample JCL to run TCPaccess in Loopback:

```
***** Top of Data *****
//RUNTCP JOB /*JOBPARM LINES=999
//**MAIN LINES=(999,W)
//*
//* SAMPLE JCL PROCEDURE TO RUN TCP/IP
//* THIS JCL CAN BE USED WITH ANY INTERFACE
//*
//* EDIT THE TRGINDX, SSN, SRC, SOUT, CMND SYMBOLIC
//* PARAMETERS
//*
//* VERIFY THAT THE JOB CARD AND NAMING CONVENTIONS MEET
//* YOUR SITE'S JCL REQUIREMENTS, THEN SUBMIT THIS JOB.
//*
//TCPIP PROC TRGINDX='TRGINDX', TARGET LIBRARIES DSN INDEX
//          SSN=ACSS,          DFLT SUBSYSTEM NAME
//          SRC='% ',          DFLT SUBSYSTEM RECOGNITION CHAR
//          SOUT='* ',          CHOOSE A HOLD NONPURGE SYSOUT CLASS
//          CMND=START00,      DFLT STARTUP COMMAND SCRIPT NAME
//          CNFG=00           IJTCFGxx SUFFIX
//*
//TCPIP EXEC PGM=IFSSTART,REGION=6144K,TIME=1440,
// PARM=' IFSINIT,U=&SSN,P=T01,SR=&SRC,SO=&SOUT,CM=&CMND,CF=&CNFG'
//*
//STEPLIB DD DISP=SHR,DSN=&TRGINDX..LOAD
//          DD DISP=SHR,DSN=&TRGINDX..SASLINK
//*
//* WARNING: THE LOAD DATA SET MUST NEVER BE ADDED TO THE LINK LIST.
//*          TCPACCESS' ELEMENT NAMES ARE NOT UNIQUE AND COULD AFFECT
//*          THE OPERATIONS OF OTHER SOFTWARE. THE LOAD DATA SET
//*          SHOULD ALWAYS BE REFERENCED THROUGH A STEPLIB OR JOBLIB
//*          STATEMENT.
//*
```

```
//*          CONFIGURATION DATA SETS
//*
//SYSPARM DD  DISP=SHR,DSN=&TRGINDX..PARM
//SYSPROC DD  DISP=SHR,DSN=&TRGINDX..PARM
//*
//*          LOG DATA SETS
//*
//T01LOG DD  SYSOUT=&SOUT
//SYSPRINT DD  SYSOUT=&SOUT
//DNRLOG DD  SYSOUT=&SOUT
//DNRERR DD  SYSOUT=&SOUT
//GTDLOG DD  SYSOUT=&SOUT
//GTDERR DD  SYSOUT=&SOUT
//GTDTRC DD  SYSOUT=&SOUT
//MAPLOG DD  SYSOUT=&SOUT
//MAPERR DD  SYSOUT=&SOUT
//SNMLOG DD  SYSOUT=&SOUT
//*
//*          DUMP DATA SETS
//*
//SYSUDUMP DD  SYSOUT=&SOUT
//*
//*          MISC DATA SETS
//*
//ARPAHELP DD  DISP=SHR,DSN=&TRGINDX..HELP
//SYSHelp DD  DISP=SHR,DSN=&TRGINDX..HELP
//ABNLIGNR DD  DUMMY          /* DISABLE ABEND-AID PROCESSING */
//          PEND
//TCPIP EXEC TCPIP
```

Testing TCPaccess in Loopback Mode

The local Internet address for loopback is 127.0.0.1. Read the *TCPaccess Customization Guide* for additional information on testing TCPaccess in loopback mode.

1. Test the VTAM/TELNET interface by entering this VTAM LOGON command:

LOGON APPLID(ACCES) DATA(127.0.0.1)

Successful loopback returns the Server TELNET screen.

You might need to use the Assembler syntax for LOGON. Specify:

LOGON APPLID=ACCES, DATA=127.0.0.1

2. Enter bye to exit.
3. Test FTP and TELNET TSO commands by logging on to TSO and entering these commands:
 - TELNET - Returns TSO TELNET Screen. Enter bye to exit.
 - FTP - Returns FTP prompt. Enter end to exit.
 - FTP2 - Returns FTP2 login prompt. Enter bye to exit.

CAUTION ! You must use a Usermod to create the FTP aliases. Read [“Usermods for ‘FTP’ Alias” on page 1-13.](#)

To use the FTP, FTP2, and TELNET commands, the LINK library must be available to your TSO STEPLIB concatenation or in the MVS link list.

Note: If you plan to include FTP data transfer in the IVP, you must issue a SITE UNIT(disk) where disk is a valid DASD unit name at your installation.

Stopping the RUNTCP Started Task

Enter this MVS command from an MVS console to terminate the started task:

P RUNTCP

If TCPaccess displays a WTOR prompt, enter a Y to continue termination.

Verifying the API Installation

This section describes the Installation Verification Procedures (IVPs) you run to ensure the TCPAccess API installed correctly.

Compiling and Testing Sample C Socket Applications

If you have the IBM C/370 or SAS SAS/C C compiler, use the CLGSASC or CLGIBMC job in the SAMP data set to compile, link, and execute one of the C Socket sample applications found in the SAMP data set — SHELLO, SWHOIS, or FINGERC.

The *TCPAccess User's Guide* describes how to run ACSHELLO or FINGER after compiling and linking.

Compiling and Testing Sample C Basic Applications

If you have the IBM C/370 or SAS SAS/C C compiler, use the CLGSASC or CLGIBMC job in the SAMP data set to compile, link, and execute one of the C basic sample applications found in the SAMP data set — BHELLO or BWHOIS.

The *TCPAccess User's Guide* describes how to run ACSHELLO or FINGER after creating it.

Testing TTCP TSO Command Processor

Execute the TTCP TSO command processor to exercise the TCPAccess API components. The *TCPAccess User's Guide* describes how to run TTCP.

1. Execute this TTCP command in transmit mode to send data to the TCPAccess TCP discard port:

TTCP TRANS PORT(9) HOST(127.0.0.1)

This command sends 1024 buffers of length 1024 to the discard port. You receive a message on successful completion.

2. Execute TTCP in two (2) TSO user address spaces so that one TTCP transmits data to the other receiving TTCP.

In one TSO user address space, enter this command:

TTCP RECV PORT(2000)

In the other TSO user address space, enter this command:

TTCP TRANS PORT(2000) HOST(127.0.0.1)

TTCP sends 1024 buffers of length 1024 from the TTCP in transmit mode to the TTCP in receive mode. You receive messages from each TTCP on successful completion. The TTCP in receive mode should be stopped via TSO attention.

Running the OpenEdition IVPs

OpenEdition must be operational and you must have TCPAccess defined in the BPXPRMxx PARMLIB member before you can run the IVPs. Refer to the *TCPAccess Planning Guide* for the required statements. The OMVS shell does not have to be set up to run the IVPs as batch jobs.

Preparing to Run the IVPs

Before you begin to run the OpenEdition IVPs, you might need to make some changes. Review this list to identify any changes required.

Is TCPAccess defined to OpenEdition with an address family other than the AF_INET default of 2? If so, complete these steps:

- For batch execution, add a GPARM parameter indicating the number being used. The GPARM parameter is required for both MAKESERV and MAKECLNT.

GPARM=11

- For interactive execution, indicate the number in the program call.

ivpserv 11 & and ivpclnt 11

Do you want to improve performance of the IVP in loopback mode?

- You can improve performance in loopback mode d by increasing the LOOPMTU value in the ACPCFGxx HOST statement. The default is 8192.

Are you going to run one or both programs on UNIX? If so, you need to

- Change the #define statement for ERRNO2 to map to 0
- Comment out the #include for manifest.h

Are you running the IVP in other than loopback mode?

- If running in other than loopback mode, modify the SERVADDR address in IVPCLNT to point to the IP address where the server will be executing.

Running the IVPs in Batch Mode

1. Modify the MAKESERV JCL stream to conform to any standards you in place at your installation. Make sure trgindx is replaced with the TCPAccess high-level qualifier being used. Submit the job.
2. When the MAKESERV job is in the GO step, modify and submit the MAKECLNT job.
3. The MAKECLNT job ends automatically. The following message in the SYSPRINT output from the GO step indicates successful completion:
4. IVPCLNT: total bytes read = 52428800
5. MVS cancels the MAKESERV job.

Running the OpenEdition IVPs in the OMVS Shell

1. Use the TSO/E command, `oput`, to copy `IVPSERV`, `IVPCLNT`, and `IVPMAKE` to an appropriate HFS directory under the names
 - `ivpserv.c`
 - `ivpclnt.c`
 - `Makefile`.

Use the UNIX `chmod` command to make any necessary permission changes.

2. To compile and link both programs, enter
 - `make`**
3. To start the server in the background, enter
 - `ivpserv &`**

4. Write down the process ID that is returned.

5. The program will return the message

```
IVPSERV: listening on port 2300
```

Enter this command:

`ivpclnt`

6. Several messages will appear, indicating the data transfer is occurring, including this message

```
IVPCLNT: total bytes read = 52428800
```
7. When the command prompt reappears the client program has terminated.
8. Issue the kill command to terminate the server.

3

Converting from TCPAccess 4.1 to TCPAccess 5.3

This technical note gives information on how to convert your TCPAccess 4.1 configuration files to the configuration files used for Network/IT TCPAccess 5.3. There is also a discussion on the use of the LOGGING statement.

Several diagrams are given to show the parameter changes. In the diagrams, parameters not available in TCPAccess 5.3 are shown crossed out (~~OLDPARAM~~). New parameters are shown in **boldface type (BOLD)**.

For complete information on the TCPAccess product, please refer to your documentation set.

This document contains these sections:

- [“ACPCFG00 HOST Statement” on page 3-2](#)
- [“Application Statement Mappings” on page 3-4](#)
- [“TCP Protocol Parameters” on page 3-5](#)
- [“UDP Protocol Parameters” on page 3-7](#)
- [“RAW Protocol Parameters” on page 3-8](#)
- [“NETWORK Parameters” on page 3-9](#)
- [“ROUTE Parameters” on page 3-10](#)
- [“Multiple Default ROUTE Statements” on page 3-11](#)
- [“Driver Configuration” on page 3-12](#)
- [“IJTCFG00 Changes” on page 3-17](#)
- [“LOGGING Statement” on page 3-18](#)

ACPCFG00 HOST Statement

Many of the ACPCFG00 HOST statement parameters were distributed into new and existing configuration files for 5.3. Parameters that affect applications have been moved to APPCFG00, system parameters have been moved to IJTCFG00, and protocol parameters have been moved to TCPCFG00.

These 4.1 parameters have been deleted:

- LOGSTAMP
- LOOPBUF
- MVS
- SYSDUMP
- LOOPMTU

Parameters renamed for 4.1:

- HOST SSN -> GLOBAL API

New statements/parameters for 5.3:

- SECURITY - SECURITY CLASS, PROFILE, APPLNAME, XSEC
- LOGGING

The following diagram illustrates the changes.

Application Statement Mappings

The application statements in the former ACPCFG00 file can be mapped to the same statements in the APPCFG00 member, although the SMF statement has been moved to IJT CFG00.

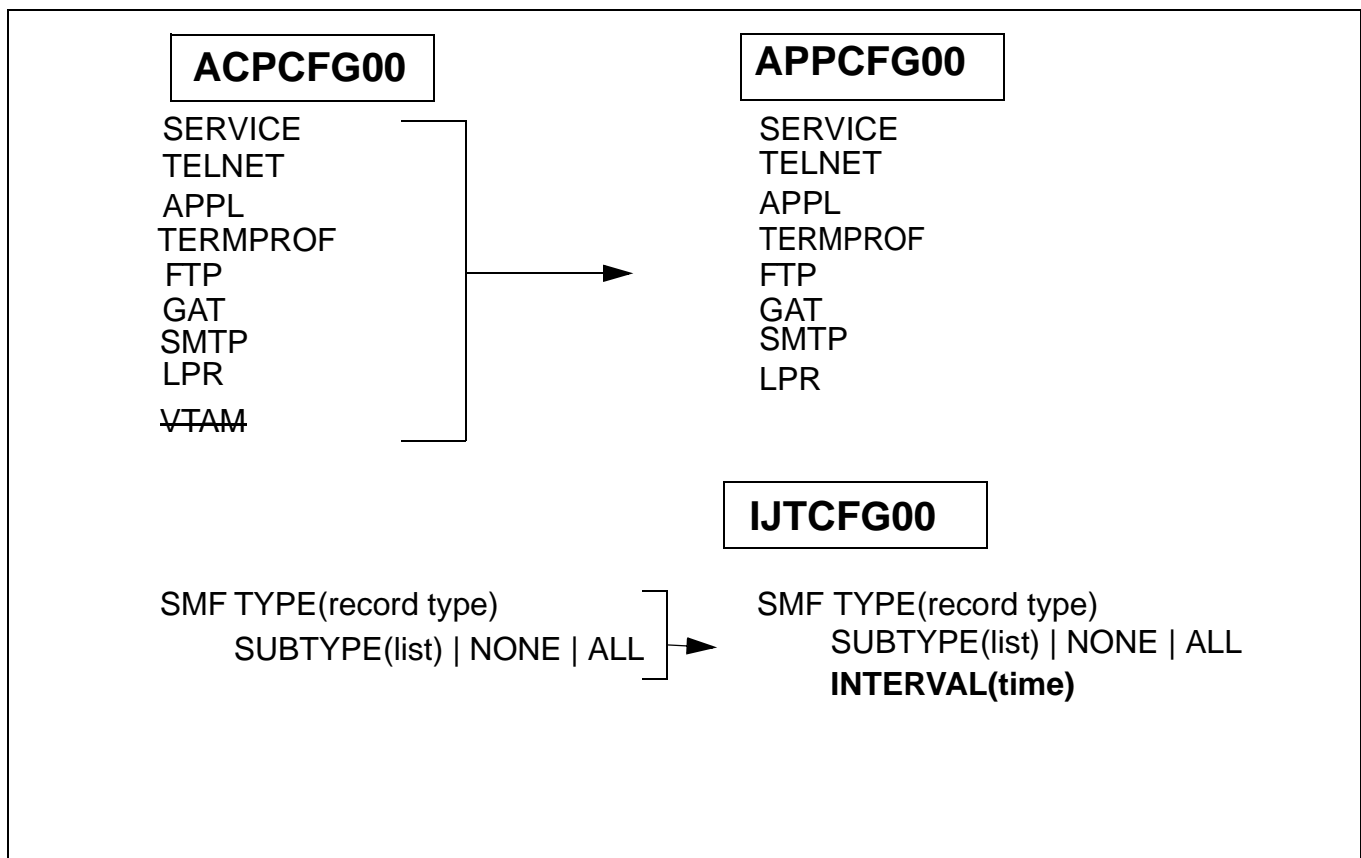
These 4.1 statements have been removed:

- VTAM

New parameters for 5.3:

- SMF INTERVAL

Figure 3-2 Application Parameters



TCP Protocol Parameters

The TIB, TCP and UDP parameters from ACPCFG00 have been moved to TCPCFG00. These parameters are now defined under each protocol, TCP, UDP, IP, and RAW. There was only one parameter name change. TADDRASSIGN and TADDRUSE have been renamed to PORTASSIGN and PORTUSE, respectively. Aliases have been added to support former names, including the TADDRASSIGN and TADDRUSE parameters.

Parameters removed from 4.1:

- PROTOCOL
- WRELIM
- RABLIM

Parameters renamed for 4.1:

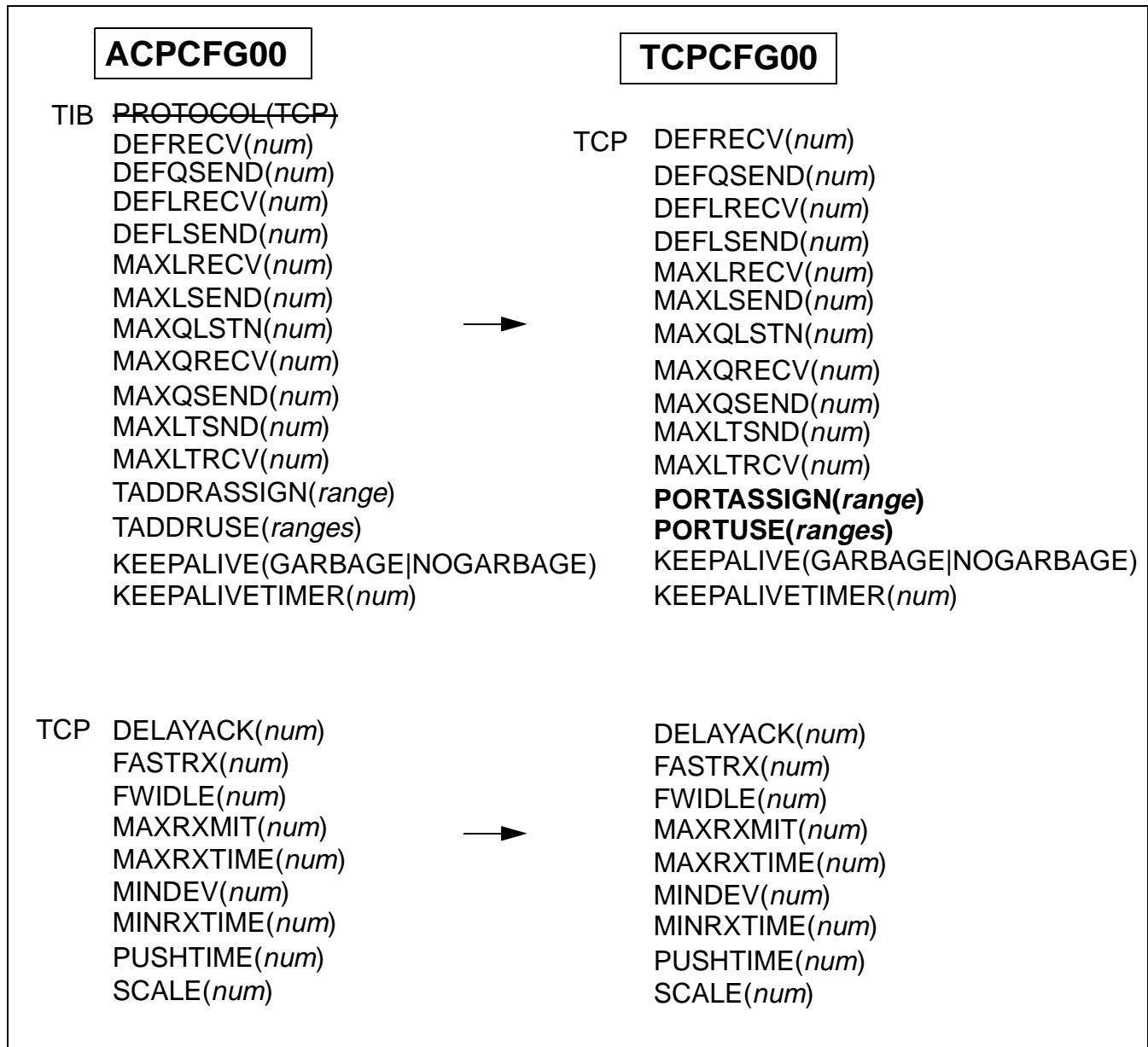
- TADDRASSIGN -> PORTASSIGN
- TADDRUSE -> PORTUSE

There are new parameters for the TCP statement in 5.3:

- IPNOTIFY
- KEEPALIVECOUNT
- RTD
- RTO
- TIMEWAIT
- TIMEWAIT

The following diagram shows the parameters.

Figure 3-3 TCP Parameters



UDP Protocol Parameters

The ACPCFG00 TIB and UDP mappings have been moved to TPPCFG00.

Parameters removed from 4.1:

- PROTOCOL(UDP)
- UDP IDLE, RABLIM, WRELIM

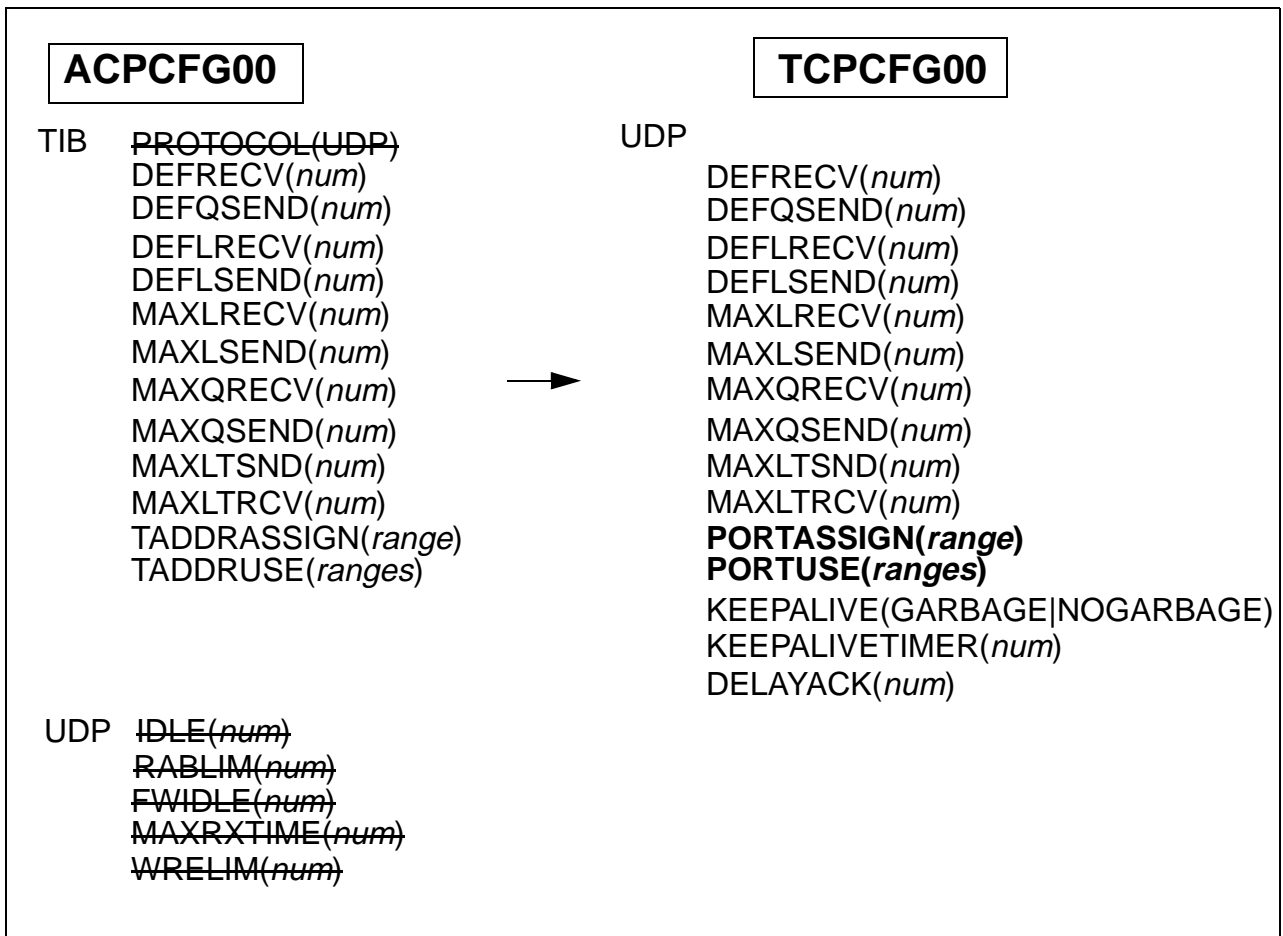
Parameters renamed for 4.1:

- TADDRASSIGN -> PORTASSIGN
- TADDRUSE -> PORTUSE

New parameters for 5.3.

- (None)

Figure 3-4 UDP Parameters



RAW Protocol Parameters

The ACPCFG00 TIB parameters for the RAW protocol have been moved to TCPCFG00.

Parameters removed from 4.1:

- PROTOCOL(RAW)

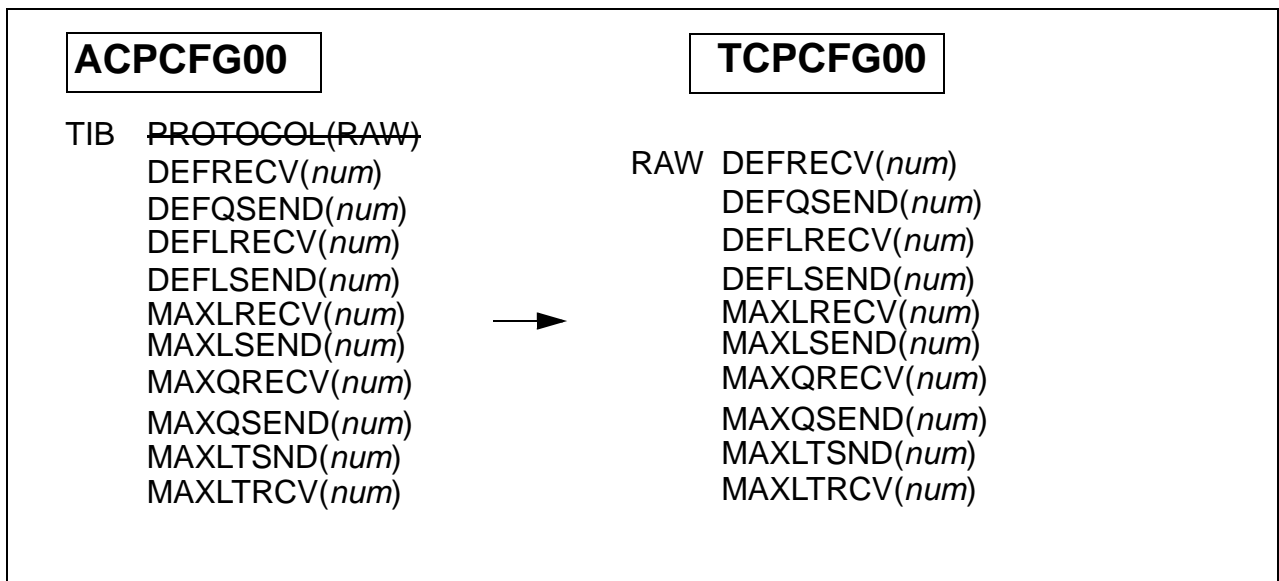
Parameters renamed for 4.1:

- (None)

New parameters for 5.3.

- (None)

Figure 3-5 RAW Protocol Parameters



NETWORK Parameters

NETWORK parameters have been moved to TCPCFG00 NETWORK and MEDIA statements.

Parameters removed from 4.1:

- ARPTABLE
- LNID
- LOOP

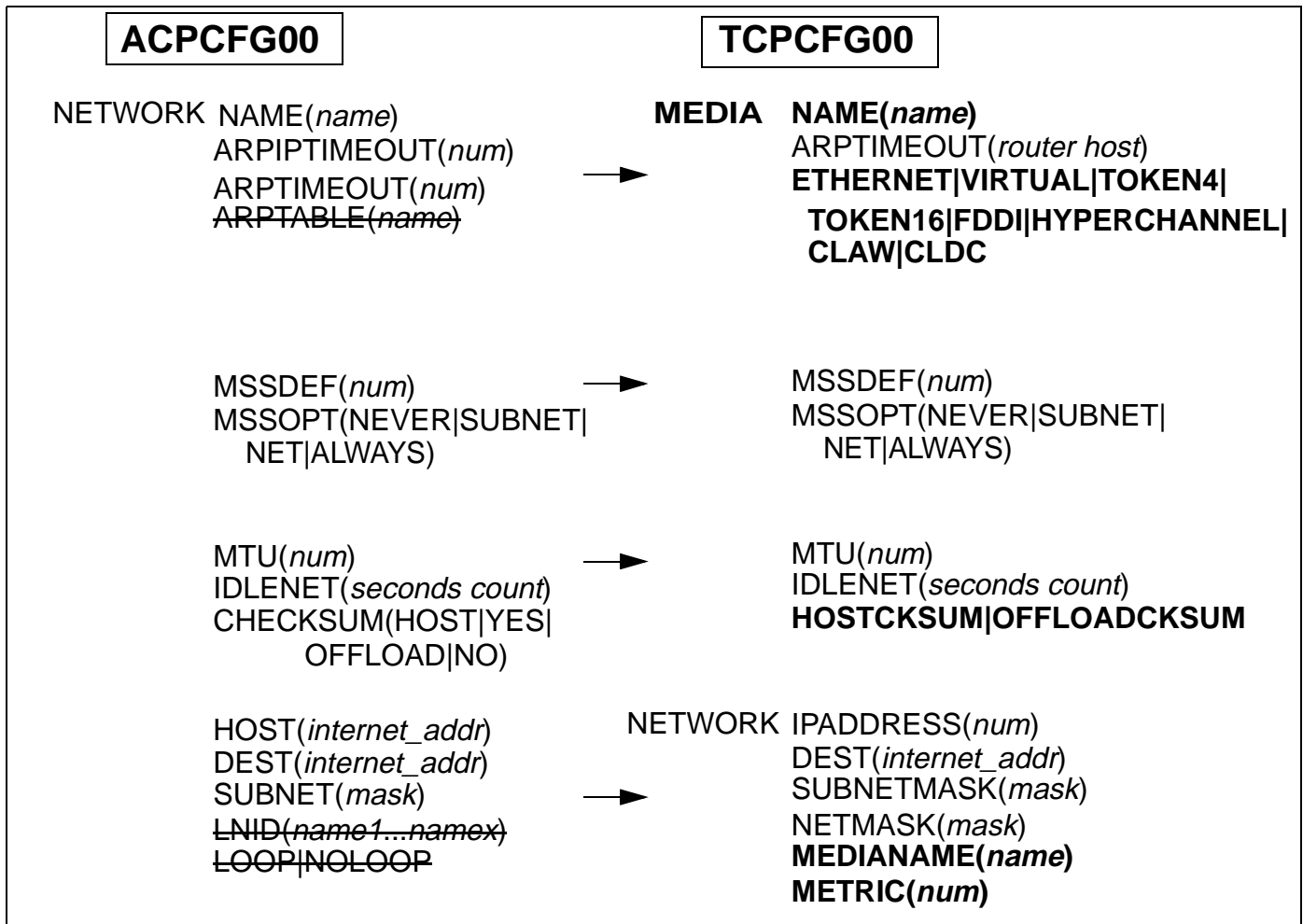
Parameters renamed for 5.3:

- CHECKSUM -> HOSTCKSUM

New parameters for 5.3.

- MEDIA Statement
- NETWORK MEDIANAME
- NETWORK METRIC

Figure 3-6 NETWORK Parameters



ROUTE Parameters

The following diagram shows the routing statements from TCPAccess 4.1 and their usage in TCPAccess 5.3. Most important to notice is that the old ROUTE NET parameter is now a ROUTE MEDIANAME parameter.

Parameters removed from 4.1:

- NET
- ARPTABLE NAME, TYPE
- ARP FLAGS, MTU, TRUNK, TYPE

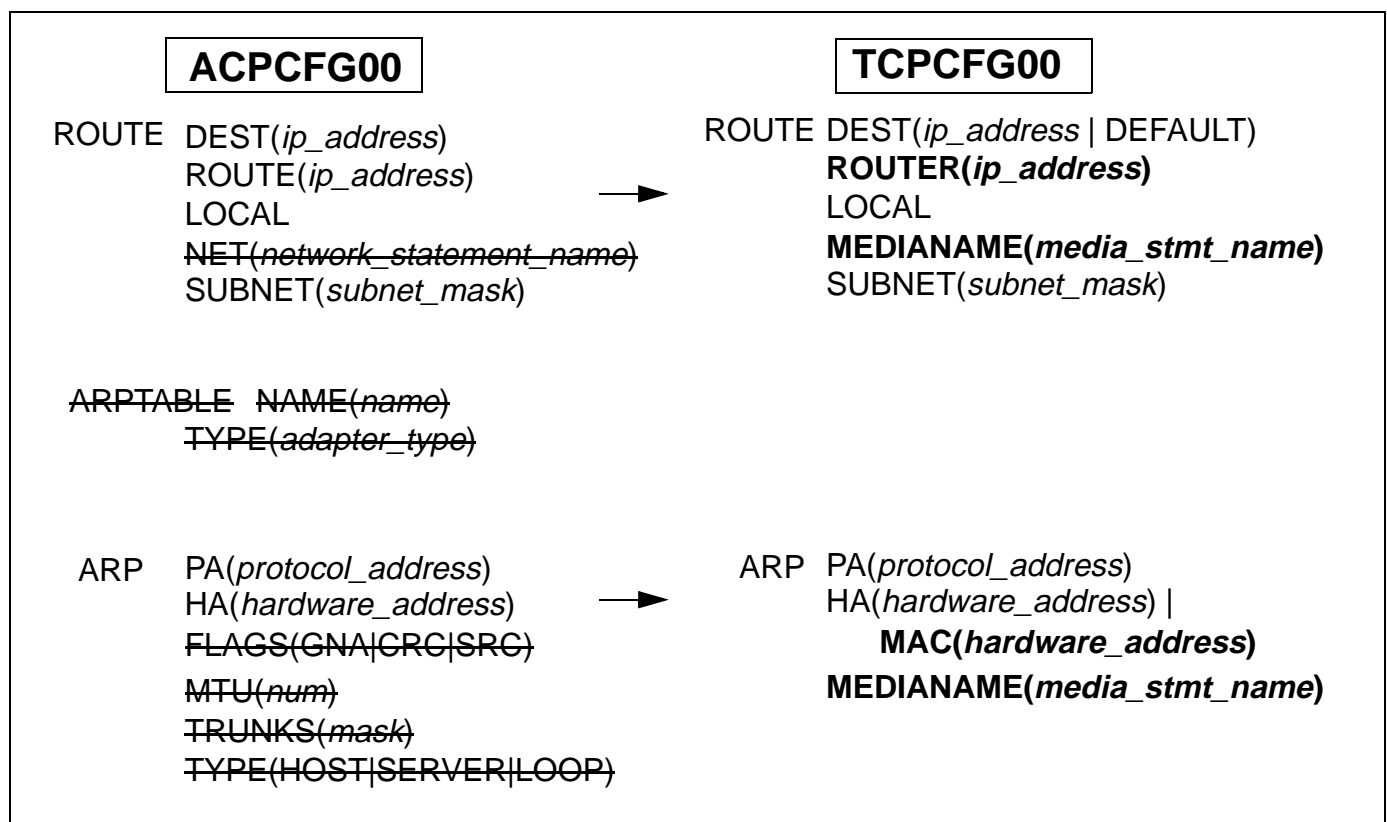
Parameters renamed for 5.3:

- ROUTE -> ROUTER

New parameters for 5.3.

- ARP MAC, MEDIANAME

Figure 3-7 ROUTE Parameters



Multiple Default ROUTE Statements

If multiple default ROUTE statements are present in the start-up configuration files, the first statement becomes the selected default route. The order of appearance in the TCPCGFxx file also corresponds to the sequence of a NETSTAT ROUTE display.

Driver Configuration

Driver configuration has been moved to TCPCFG00.

CETI Driver

CETI driver commands have been moved to TCPCFG00.

Parameters removed from 4.1:

- LOGOUT
- OPTIONS
- SIDB
- SIDM
- TRACE
- NAME
- ERROR

Parameters changed:

- LNICETI -> CETI
- RETRY has been changed to RESTART

New parameters for 5.3.

- AUTOSTART
- MEDIANAME

Figure 3-8 CETI Driver Parameters

ACPCFG00	→	TCPCFG00
LNICETI		CETI
CUTYPE(<i>cutype</i>)		CUTYPE(3752 3762))
DEVADDR(<i>ccuu</i>)		DEVADDR(<i>ccuu</i>)
IBUF(<i>number</i>)		IBUF(<i>number</i>)
IPARM(<i>buf_thresh gat_time</i>)		IPARM(<i>buf_thresh gat_time</i>)
OBUF(<i>number</i>)		OBUF(<i>number</i>)
OPARM(<i>buf_thresh gat_time</i>)		OPARM(<i>buf_thresh gat_time</i>)
RETRY(<i>sec</i>)		RESTART(<i>restart_time</i>)
WTIME(<i>number</i>)		WTIME(<i>number</i>)
LOGOUT(<i>num</i> {SYNC NOSYNC})		AUTOSTART NOAUTOSTART
OPTIONS((ASYNC SYNC)		MEDIANAME(<i>media_stmt_name</i>)
— {DSM ESM} {GETMAIN PCORE}		
— {LOOP NOLOOP} {DOCP EOCP})		
SIDB(<i>sense_id</i>)		
SIDM(<i>sense_mask</i>)		
TRACE NOTRACE		
NAME(<i>lniname</i>)		
ERROR(ABEND)		

LNICLAW

ACPCFG00 LNICLAW parameters have been moved to the TCP CFG00 CLAW statement.

Parameters removed from 4.1:

- TRACE
- NAME
- ERROR

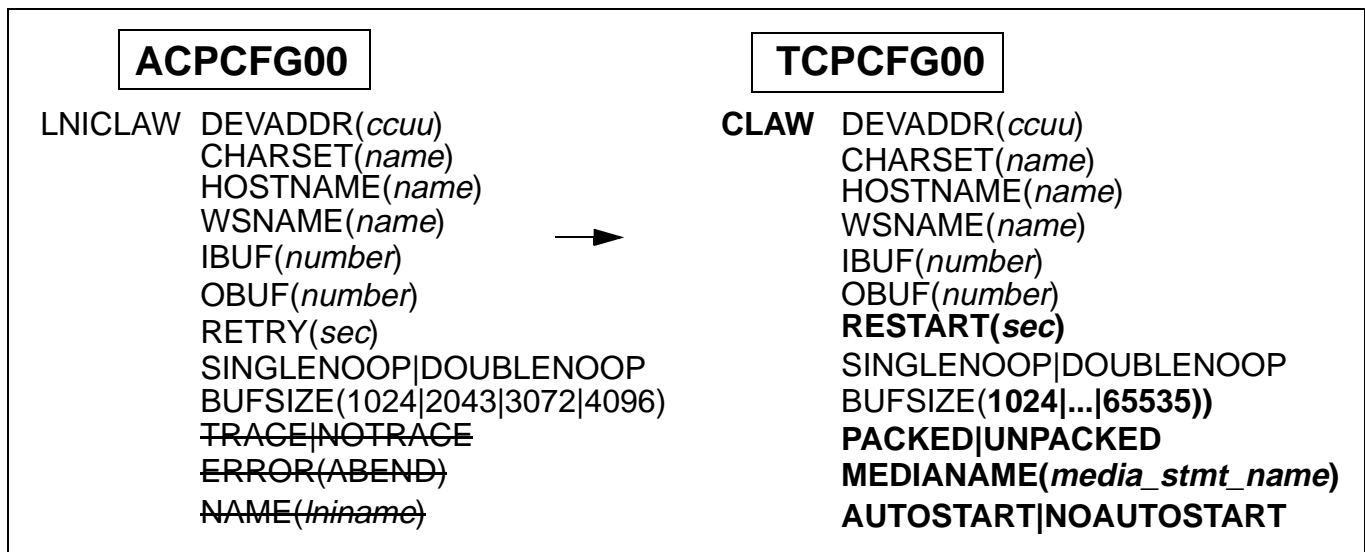
Parameters changed:

- LNICLAW -> CLAW
- RETRY has been changed to RESTART

New parameters for 5.3.

- PACKED
- MEDIANAME
- New BUFSIZE options have been added
- AUTOSTART

Figure 3-9 CLAW Driver Parameters



LNICDLC

ACPCFG00 LNICDLC parameters have been moved to the TCPCFG00 CDLC statement.

Parameters removed from 4.1:

- TRACE
- NAME
- ERROR

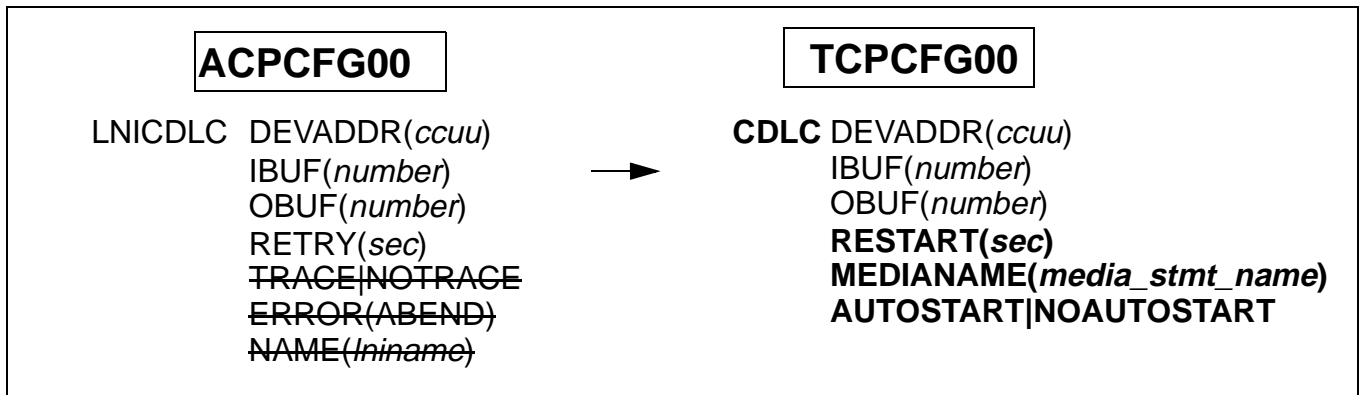
Parameters changed:

- LNICDLC -> CDLC
- RETRY has been changed to RESTART

New parameters for 5.3.

- MEDIANAME
- AUTOSTART

Figure 3-10 CDLC Driver Parameters



LNIHYPR

ACPCFG00 LNIHYPR parameters have been moved to the TCPCFG00 HYPER statement.

Parameters removed from 4.1:

- TRACE
- NAME
- ERROR

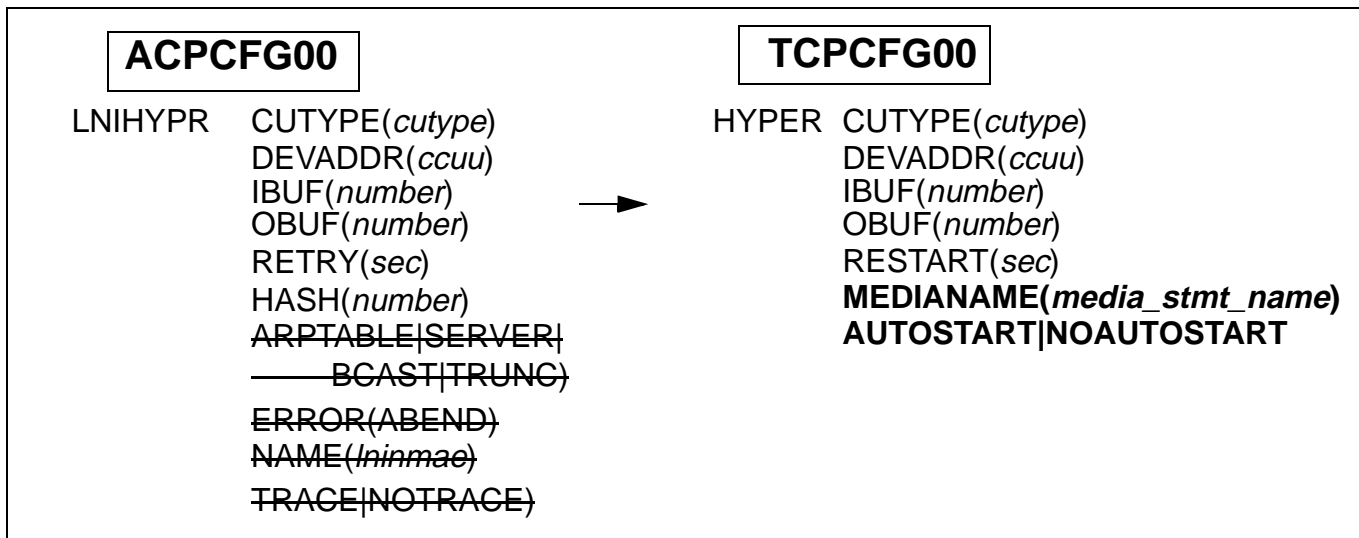
Parameters changed:

- LNIHYPR -> HYPR
- RETRY has been changed to RESTART

New parameters for 5.3.

- MEDIANAME
- AUTOSTART

Figure 3-11 HYPR Driver Parameters



LNILCS and LNILINK

ACPCFG00 LNILCS and LNILINK parameters have been moved to TCPCFG00 LCS and LINK statements.

Parameters removed from 4.1:

- TRACE
- NAME
- ERROR

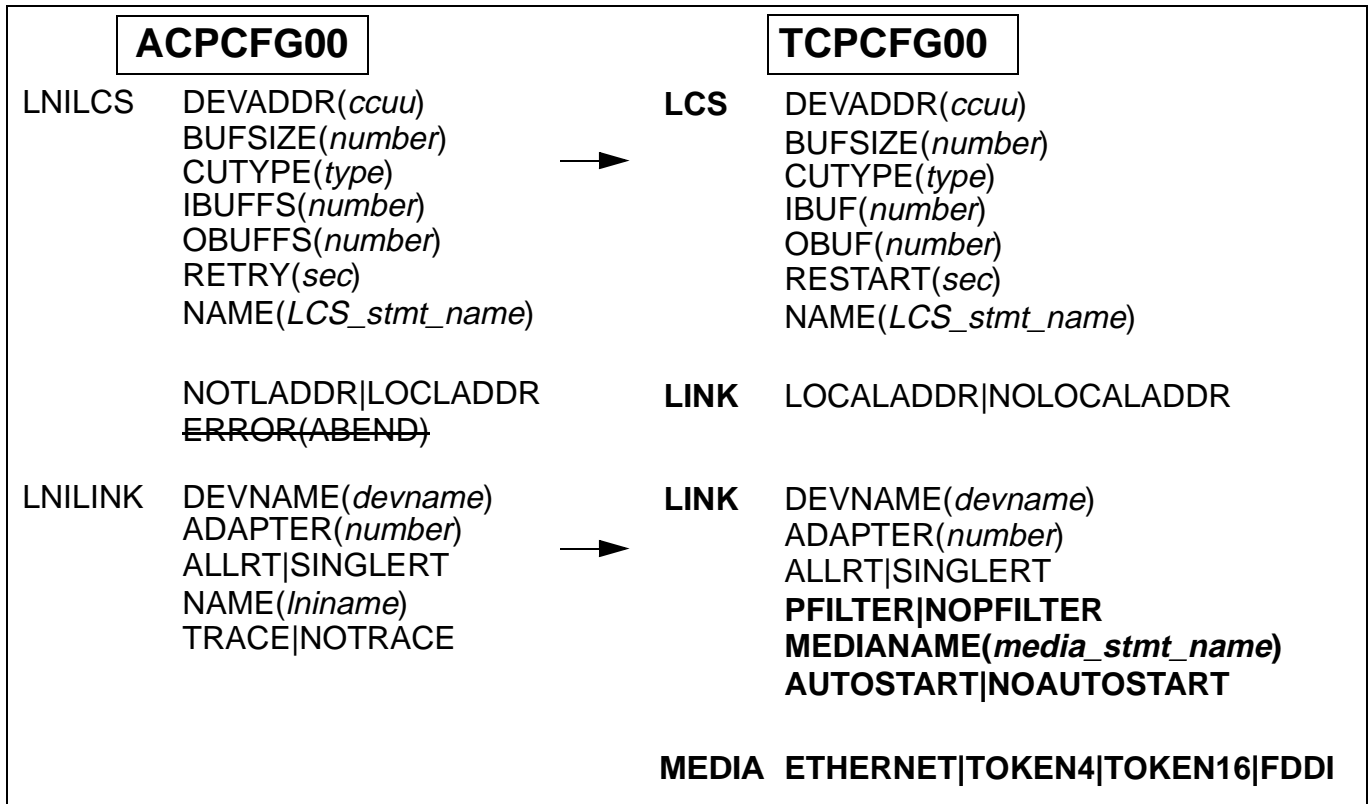
Parameters changed:

- LNILCS -> LCS
- LNILINK -> LINK
- LOCLADDR -> LOCALADDR, NOTLADDR -> NOLOCALADDR
- RETRY has been changed to RESTART

New parameters for 5.3.

- MEDIANAME
- AUTOSTART
- PFILTER
- MEDIA statement parameters ETHERNET|TOKEN4|TOKEN16|FDDI

Figure 3-12 LCS and LINK Parameters



IJTCFG00 Changes

There have been minimal changes to the IJTCFG00 IFSPARM and AUTH statements. The LOGGING statement has been added to IJTCFG00.

Parameters removed from 4.1:

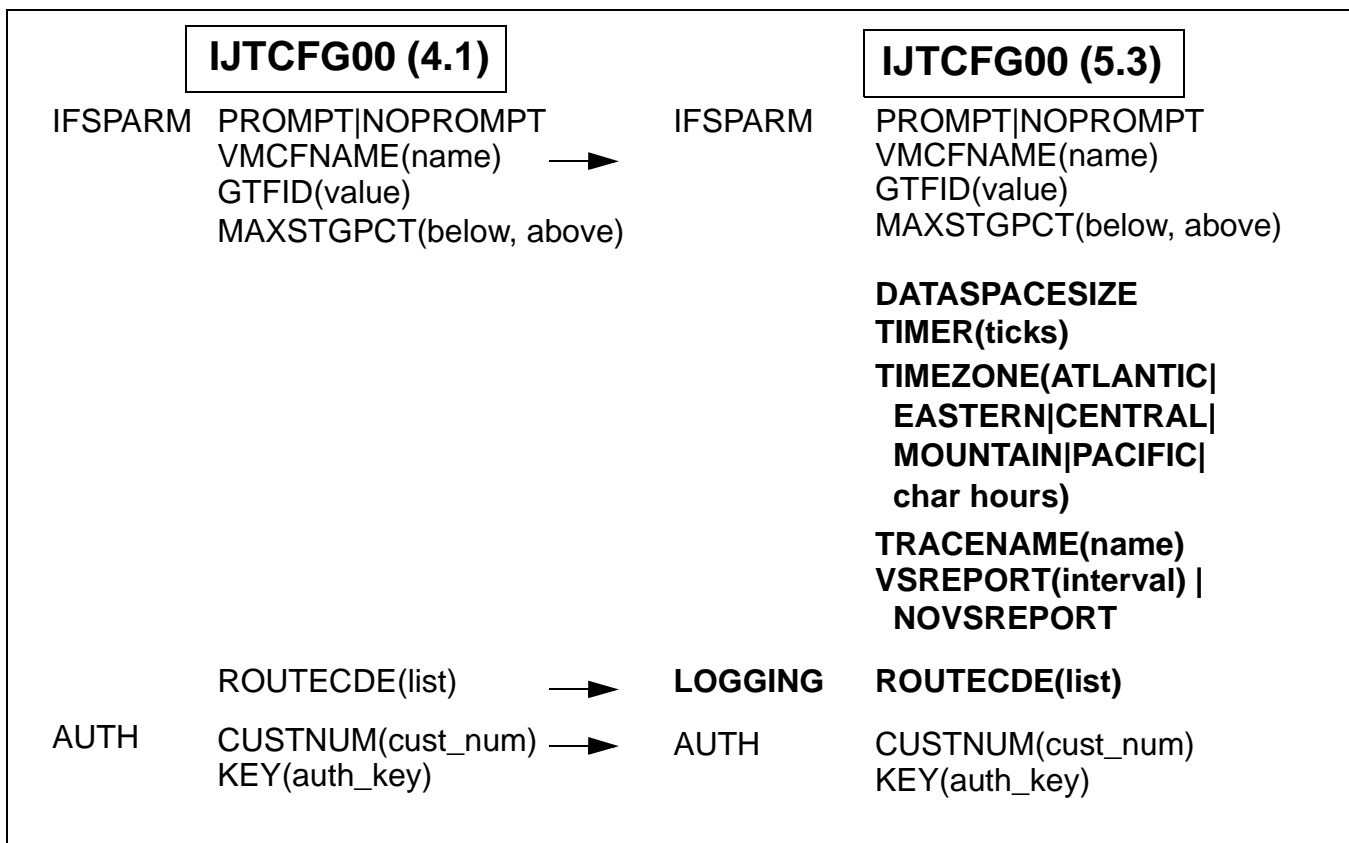
- (none)

Parameters changed:

New parameters for 5.3.

- LOGGING
- DATASPACE SIZE
- TIMER
- TIMEZONE
- TRACENAME
- VSREPORT

Figure 3-13 IFSPARM and AUTH Statements



LOGGING Statement

One of the design goals of TCPAccess 5.3 was to reduce the volume of message output printed. SMF recording was enhanced to allow many of the events previously recorded only by a message to be written to the SMF data sets.

Messages were renumbered so that each message now has a unique prefix, which includes both a severity, and a component code to identify the area of the product which issued the message. The LOGGING statement was added in IJTCFGxx to give you control over which messages will be written to the log or to the console. Message routing can be controlled by severity and component, using the LOGGING statement.

By default, many events previously reported by messages will go unreported in TCPAccess 5.3. Some customers may prefer to continue seeing the same level of message output as in previous releases of the product. The following LOGGING statement parameters are designed to give a level of event recording reasonably close to what was provided by previous releases of TCPAccess:

```
LOGGING PRINT(ALL,(DN,RFEW),(IF,RFEWIS))  
WTO((LL,RFEWIS),(CF,RFEWI))
```

You may find that there are other defaults you will wish to change. The LOGGING parameters can be changed dynamically by means of the new LOGGING command, making it easy to experiment with different configurations.

For complete information on the LOGGING statement, and the LOGGING operator command, refer to the TCPAccess documentation.