



CHAPTER

25

Options for Starting SAS/ CONNECT

Introduction 213

Dictionary 213

Introduction

System options are used by the SAS system and SAS/CONNECT software to control the environment and processing on a local and a remote host. The values assigned to system options are specified in the SAS command, in a configuration or autoexec file, or in the OPTIONS statement. Values for system options are effective throughout a SAS session (local or remote), not just for a step, data set, or remote submit. Some of the options used with SAS/CONNECT are specified in your local SAS session, while other options are specified when you invoke SAS on the remote host.

Dictionary

COMAMID=

Identifies communications access method.

Local and remote

Syntax

COMAMID=*access-method-id*

Syntax Description

access-method-id

specifies the name of the communications access method that is used by SAS/CONNECT when connecting to remote hosts.

Details

The COMAMID= system option specifies a communications access method that is used by SAS/CONNECT when connecting to remote hosts. To establish the link between two hosts that are running the SAS System, use the COMAMID= option on both hosts. Specify the COMAMID= option in an OPTIONS statement before you sign on to the link. On the remote host, include the COMAMID= option in the script TYPE statement that invokes the remote SAS session or in a remote SAS session configuration.

On the local host, you must specify both a COMAMID= option and a REMOTE= option. Use the following table to select the appropriate values for the COMAMID= and REMOTE= options for both hosts.

Note: You may need to customize some scripts for your site. The table notes (explained at the end of the Table) indicate modifications that are necessary, but your site may require more. △

Table 25.1 REMOTE= and COMAMID= System Options Local and Remote Hosts

If your local host is ...	and your remote host is ...	use this COMAMID= value		
		on the local host ... ¹	use this REMOTE= value on the local host ... ¹	and use this supplied script. ²
OS/390	OS/390	APPC	<i>luname</i>	none, in most cases ³
		TCP	<i>internet-address</i> ⁴	tcptso.scr or tcptso32.scr
	CMS	APPC	<i>AVS-gateway</i>	none
		TCP	<i>internet-address</i> ⁴	tcpcms.scr
	OpenVMS Alpha	TCP	<i>internet-address</i> ⁴	tcpvms.scr
		TCP	<i>internet-address</i> ⁴	tcpvms.scr
	UNIX	TCP	<i>internet-address</i> ⁴	tcpunix.scr
		TCP	<i>internet-address</i> ⁴	tcpunix.scr
	OS/2	APPC	<i>control-point-lu</i>	none
		TCP	<i>internet-address</i> ⁴	tcpo2.scr
	Windows NT	APPC	<i>luname</i>	none
		TCP	<i>internet-address</i> ⁴	tcpwin.scr
	Windows 95	APPC	<i>luname</i>	none
		TCP	<i>internet-address</i> ⁴	tcpwin.scr
CMS	OS/390	APPC	<i>luname</i>	none, in most cases ³
		TCP	<i>internet-address</i> ⁴	tcptso.scr or tcptso32.scr
	CMS	APPC	<i>AVS-gateway</i>	none
		TCP	<i>internet-address</i> ⁴	tcpcms.scr or tcpcms32.scr

If your local host is ...	and your remote host is ...	use this COMAMID= value		
		on the local host ... ¹	use this REMOTE= value on the local host ... ¹	and use this supplied script. ²
	OpenVMS Alpha	TCP	<i>internet-address</i> ⁴	tcpvms.scr
	OpenVMS VAX	TCP	<i>internet-address</i> ⁴	tcpvms.scr
	UNIX	TCP	<i>internet-address</i> ⁴	tcpunix.scr
	OS/2	APPC	<i>control-point-lu</i>	none
		TCP	<i>internet-address</i> ⁴	tcpos2.scr
	Windows NT	APPC	<i>luname</i>	none
		TCP	<i>internet-address</i> ⁴	tcpwin.scr
	Windows 95	APPC	<i>luname</i>	none
		TCP	<i>internet-address</i> ⁴	tcpwin.scr
OpenVMS Alpha OpenVMS VAX	OS/390	TCP	<i>internet-address</i> ⁴	tcptso.scr or tcptso32.scr
		TELNET	<i>internet-address</i> ⁴	teltso.scr
	CMS	TCP	<i>internet-address</i> ⁴	tcpcms.scr or tcpcms32.scr
		TELNET	<i>internet-address</i> ⁴	telcms.scr
	OpenVMS Alpha	TCP	<i>internet-address</i> ⁴	tcpvms.scr
		TELNET	<i>internet-address</i> ⁴	telvms.scr
		DECNET	<i>host-id</i>	none
	OpenVMS VAX	TCP	<i>internet-address</i> ⁴	tcpvms.scr
		TELNET	<i>internet-address</i> ⁴	telvms.scr
		DECNET	<i>host-id</i>	none
	UNIX	TCP	<i>internet-address</i> ⁴	tcpunix.scr
	OS/2	TCP	<i>internet-address</i> ⁴	tcpos2.scr
	Windows NT	TCP	<i>internet-address</i> ⁴	tcpwin.scr
		DECNET	<i>host-id</i>	none
	Windows 95	TCP	<i>internet-address</i> ⁴	tcpwin.scr
		DECNET	<i>host-id</i>	none
UNIX	OS/390	APPC	<i>luname</i>	none
		TCP	<i>internet-address</i> ⁴	tcptso.scr or tcptso32.scr
		TELNET	<i>internet-address</i> ⁴	teltso.scr
	CMS	APPC	<i>AVS-gateway</i>	none
		TCP	<i>internet-address</i> ⁴	tcpcms.scr or tcpcms32.scr
		TELNET	<i>internet-address</i> ⁴	telcms.scr

If your local host is ...	and your remote host is ...	use this COMAMID= value		
		on the local host ... ¹	use this REMOTE= value on the local host ... ¹	and use this supplied script. ²
	OpenVMS Alpha	TCP	<i>internet-address</i> ⁴	tcpvms.scr
		TELNET	<i>internet-address</i> ⁴	telvms.scr
	OpenVMS VAX	TCP	<i>internet-address</i> ⁴	tcpvms.scr
		TELNET	<i>internet-address</i> ⁴	telvms.scr
	UNIX	TCP	<i>internet-address</i> ⁴	tcpunix.scr
		OS/2	APPC	<i>control-point-lu</i>
	OS/2	APPC	<i>control-point-lu</i>	none
		TCP	<i>internet-address</i> ⁴	tcpos2.scr
	Windows NT	APPC	<i>luname</i>	none
		TCP	<i>internet-address</i> ⁴	tcpwin.scr
	Windows 95	APPC	<i>luname</i>	none
		TCP	<i>internet-address</i> ⁴	tcpwin.scr
OS/2 (except with Release 6.06)	OS/390	APPC	<i>luname</i> ³	none, in most cases ³
		EHLAPI	<i>remote-session</i>	tso.scr or logtso.scr
		TCP	<i>internet-address</i> ⁴	tcptso.scr or tcptso32.scr
		TELNET	<i>internet-address</i> ⁴	teltso.scr
	CMS	APPC	<i>AVS-gateway</i>	none
		EHLAPI	<i>remote-session</i>	cms.scr or logcms.scr
		TCP	<i>internet-address</i> ⁴	tcpcms.scr or tcpcms32.scr
		TELNET	<i>internet-address</i> ⁴	telcms.scr
	OpenVMS Alpha	TCP	<i>internet-address</i> ⁴	tcpvms.scr
		TELNET	<i>internet-address</i> ⁴	telvms.scr
	OpenVMS VAX	TCP	<i>internet-address</i> ⁴	tcpvms.scr
		TELNET	<i>internet-address</i> ⁴	telvms.scr
	UNIX	TCP	<i>internet-address</i> ⁴	tcpunix.scr
		OS/2	APPC	<i>control-point-lu</i>
	OS/2	APPC	<i>control-point-lu</i> (for the remote PC)	none
		TCP	<i>internet-address</i> ⁴	tcpos2.scr
		NETBIOS	<i>network-name</i>	none
		APPC	<i>luname</i>	none
	Windows NT	APPC	<i>luname</i>	none
		TCP	<i>internet-address</i> ⁴	tcpwin.scr
		NETBIOS	<i>network-name</i>	none

If your local host is ...	and your remote host is ...	use this COMAMID= value		
		on the local host ... ¹	use this REMOTE= value on the local host ... ¹	and use this supplied script. ²
Windows 95	Windows 95	APPC	<i>luname</i>	none
		TCP	<i>internet-address</i> ⁴	tcpwin.scr
		NETBIOS	<i>network-name</i>	none
Windows 95	OS/390	EHLAPI	<i>remote-session</i>	tso.scr or logtso.scr
		TCP	<i>internet-address</i> ⁴	tcptso.scr or tcptso32.scr
		TELNET	<i>internet-address</i> ⁴	teltso.scr
		APPC	<i>luname</i>	none
		EHLAPI	<i>remote-session</i>	cms.scr or logcms.scr
		TCP	<i>internet-address</i> ⁴	tcpcms.scr or tcpcms32.scr
	CMS	TELNET	<i>internet-address</i> ⁴	telcms.scr
		APPC	<i>AVS gateway</i>	none
		EHLAPI	<i>remote-session</i>	cms.scr or logcms.scr
		TCP	<i>internet-address</i> ⁴	tcpcms.scr or tcpcms32.scr
		TELNET	<i>internet-address</i> ⁴	telcms.scr
		APPC	<i>AVS gateway</i>	none
	OpenVMS Alpha	EHLAPI	<i>remote-session</i>	cms.scr or logcms.scr
		TCP	<i>internet-address</i> ⁴	tcpvms.scr
		TELNET	<i>internet-address</i> ⁴	telvms.scr
	OpenVMS VAX	DECNET	<i>host-id</i>	none
		TCP	<i>internet-address</i> ⁴	tcpvms.scr
		TELNET	<i>internet-address</i> ⁴	telvms.scr
		DECNET	<i>host-id</i>	none
		TCP	<i>internet-address</i> ⁴	tcpvms.scr
		TELNET	<i>internet-address</i> ⁴	telvms.scr
	UNIX	DECNET	<i>host-id</i>	none
		TCP	<i>internet-address</i> ⁴	tcpunix.scr
		TELNET	<i>internet-address</i> ⁴	telvms.scr
	OS/2	DECNET	<i>host-id</i>	none
		TCP	<i>internet-address</i> ⁴	tcpvms.scr
		TELNET	<i>internet-address</i> ⁴	telvms.scr
		DECNET	<i>host-id</i>	none
		TCP	<i>internet-address</i> ⁴	tcpvms.scr
		TELNET	<i>internet-address</i> ⁴	telvms.scr
	Windows NT	DECNET	<i>host-id</i>	none
		TCP	<i>internet-address</i> ⁴	tcpvms.scr
		TELNET	<i>internet-address</i> ⁴	telvms.scr
		DECNET	<i>host-id</i>	none
		TCP	<i>internet-address</i> ⁴	tcpvms.scr
		TELNET	<i>internet-address</i> ⁴	telvms.scr
	Windows 95	DECNET	<i>host-id</i>	none
		TCP	<i>internet-address</i> ⁴	tcpvms.scr
		TELNET	<i>internet-address</i> ⁴	telvms.scr
		DECNET	<i>host-id</i>	none
		TCP	<i>internet-address</i> ⁴	tcpvms.scr
		TELNET	<i>internet-address</i> ⁴	telvms.scr
Windows NT	OS/390	DECNET	<i>host-id</i>	none
		TCP	<i>internet-address</i> ⁴	tcpvms.scr
		TELNET	<i>internet-address</i> ⁴	telvms.scr

use this COMAMID= value				
If your local host is ...	and your remote host is ...	on the local host ... ¹	use this REMOTE= value on the local host ... ¹	and use this supplied script. ²
		TCP	<i>internet-address</i> ⁴	tcptso.scr or tcptso32.scr
		EHLAPI	<i>remote-session</i>	tso.scr or logtso.scr
		TELNET	<i>internet-address</i> ⁴	teltso.scr
	CMS	APPC	<i>luname</i>	none
		TCP	<i>internet-address</i> ⁴	tcpcms.scr or tcpcms32.scr
		EHLAPI	<i>remote-session</i>	cms.scr or logcms.scr
		TELNET	<i>internet-address</i> ⁴	telcms.scr
	OpenVMS Alpha	TCP	<i>internet-address</i> ⁴	tcpvms.scr
		TELNET	<i>internet-address</i> ⁴	telvms.scr
		DECNET	<i>host-id</i>	none
	OpenVMS VAX	TCP	<i>internet-address</i> ⁴	tcpvms.scr
		TELNET	<i>internet-address</i> ⁴	telvms.scr
		DECNET	<i>host-id</i>	none
	UNIX	TCP	<i>internet-address</i> ⁴	tcpunix.scr
	OS/2	APPC	<i>luname</i>	none
		TCP	<i>internet-address</i> ⁴	tcp2.scr
		NETBIOS	<i>network-name</i>	none
	Windows NT	APPC	<i>luname</i>	none
		TCP	<i>internet-address</i> ⁴	tcpwnt.scr
		DECNET	<i>host-id</i>	none
		NETBIOS	<i>network-name</i>	none
	Windows 95	APPC	<i>luname</i>	none
		TCP	<i>internet-address</i> ⁴	tcpwin.scr
		DECNET	<i>host-id</i>	none
		NETBIOS	<i>network-name</i>	none

Table Notes

- 1 Submit these options in the OPTIONS statement on the local host.
- 2 Use a FILENAME statement to associate this script with the fileref RLINK or another fileref that you will use in the SIGNON and SIGNOFF commands.
- 3 Use a script with APPC only when you are running a version of OS/390/ESA that is earlier than Version 4, Release 2 on OS/390 as the remote host. In this case, use

the APPCTSO.SCR script with OS/2 only. In addition, if you are using a script, the value of the REMOTE= option on the local host is the *session id*, not the *luname-of-APPC/OS/390*.

- 4 The value of the REMOTE= option must be a valid SAS name. If you have an Internet address that exceeds eight characters or does not conform to SAS naming conventions in any other way, assign the address to a macro variable and specify that macro variable for the REMOTE= value, as illustrated here:

```
%let mynode=internet-address;
options remote=mynode;
```

Do not choose a macro variable that is also a valid host name on your network. The SAS System first attempts to reach a network host by using the value of the REMOTE= option (in this example, **mynode**).

Specifying the COMAMID= System Option on the Local Host

Usually, you set up the SAS autoexec file on the local host so that the COMAMID= option executes automatically. To accomplish this, include an OPTIONS statement in the autoexec file that sets the value of the COMAMID= option for the local host each time the local SAS System is invoked.

If you do not include an OPTIONS statement that specifies the COMAMID= option in your SAS autoexec file, you can use one of these methods to specify the system option on the local host:

- Type and submit the OPTIONS statement before executing a SIGNON command.
- If you are using pull-down menus in the SAS windowing environment, complete the **Communications access method ID:** field in the dialog box that appears after you select the SIGNON item from the pull-down menu.
- Include the COMAMID= option in the SAS command that starts the SAS session on the local host.
- Include the COMAMID= option in the SAS configuration file that is processed when you start the SAS session on the local host. See the SAS documentation for your local host for more information about the configuration file.

Specifying the COMAMID= System Option on the Remote Host

For the remote host, you can specify the COMAMID= system option in the TYPE statement that contains the SAS command in the script that you use to sign on. Alternatively, you can specify the COMAMID= option in the configuration file that is processed when you invoke the SAS System on the remote host. If you fail to specify the COMAMID= option in one of these places, an error message is issued and the link terminates.

CONNECTREMOTE=

Identifies which remote session to connect to.

Local

Syntax

CONNECTREMOTE=*remote-session-id*

Syntax Description

remote-session-id

specifies the remote session to connect to when using SAS/CONNECT.

Details

Use the CONNECTREMOTE= option to identify which remote session or machine to use with SAS/CONNECT. There is no default value for the CONNECTREMOTE= option; you must supply one. Aliases for this option are CREMOTE= and REMOTE=.

When you are signing on, the CONNECTREMOTE= option is used in combination with the COMAMID= option. Valid combinations of values for these options are shown in the table with the COMAMID= option.

You can also use this system option to identify the session to which statements should be remote-submitted when you have multiple links established.

Note: This option is available in Version 7 and later. △

SIGNON Example

This example shows the OPTIONS statement that you use on the local host and the script TYPE statement that invokes the SAS System on the remote host. (For additional examples, refer to the chapters for each access method in *Communications Access Methods for SAS/CONNECT and SAS/SHARE Software*.)

Suppose you use Communications Manager to communicate with a CMS remote host and you have named this session MYCMS. Specify the COMAMID= and CONNECTREMOTE= options in the local SAS session OPTIONS statement as follows:

```
options comamid=ehllapi connectremote=mycms;
```

The TYPE statement in the script that invokes the SAS session on the CMS host looks like this:

```
type
"sas (dmr comamid=pclink noterminal no$syntaxcheck)"
enter;
```

Note: You can omit the CONNECTREMOTE= option and specify the remote-session id in the SIGNON command if you have assigned the fileref RLINK to the script file that is used to sign on. △

Examples for Remote-Submitting

Suppose that you signed on to an OpenVMS Alpha host by means of TCP/IP and you use REMOTE=*internet-address*, then you signed on to an OS/390 host by means of EHLLAPI and you use REMOTE=A. You can remote-submit statements to either of these hosts by specifying the remote-session id when you issue the RSUBMIT statement or command. This example submits statements to both hosts:

```
options remote=a;
rsubmit;
    statements for OS/390 remote host
endrsubmit;
```



```
options cremote=tcp-host-name;
rsubmit;
    statements for OpenVMS Alpha remote host
endrsubmit;
```

You can also omit the REMOTE= option and simply include the remote-session id in the RSUBMIT statement as follows:

```
rsubmit a;
    statements for OS/390 remote host
endrsubmit;

rsubmit tcp-host-name;
    statements for OpenVMS Alpha remote host
endrsubmit;
```

CONNECTSTATUS=

Specifies the default setting for the display of the status window.

Local

Syntax

CONNECTSTATUS=YES|NO

Syntax Description

YES|Y

status window will be displayed during file transfers

NO|N

status window will not be displayed during file transfers

Details

The CONNECTSTATUS= system option is used to set whether a status window displays by default during a PROC UPLOAD or a PROC DOWNLOAD. This default setting can be overridden by specifying the CONNECTSTATUS= option in subsequent RSUBMIT, PROC UPLOAD or PROC DOWNLOAD statements for specific transfers. CSTATUS and STATUS are aliases for this option.

Note: This option is available in Version 7 and later. △

CONNECTWAIT=

Specifies the default whether remote submits are to be executed synchronously or asynchronously.

Remote

Syntax

CONNECTWAIT=YES|NO

Syntax Description

YES|Y

indicates to execute RSUBMITs synchronously.

NO|N

indicates to execute RSUBMITs asynchronously.

Details

The CONNECTWAIT= system option is used to set whether remote submits are to be executed synchronously or asynchronously by default during a SAS session. This default setting can be overridden by specifying the CONNECTWAIT= option in subsequent RSUBMIT statements for a particular remote submit. CWAIT is the alias for this option.

Synchronous processing means that you will wait for the remote processing to complete before regaining control in the local SAS session. This is the default processing technique for RSUBMIT if the CONNECTWAIT option is not specified. In the asynchronous case, after the RSUBMIT statement begins to execute in the background of the remote host, you will regain control of your local SAS session to continue local processing or to continue to RSUBMIT to other remote sessions.

If CONNECTWAIT=NO is specified, it will also be useful to specify the CMACVAR= option in the RSUBMIT statement. This will allow you to test the status of the current asynchronous RSUBMIT by determining whether it has completed or is still in progress.

When %SYSRPUT executes within a synchronous (CWAIT=YES) remote submit, the macro variable is defined to the local SAS session as soon as it executes.

When %SYSRPUT executes within an asynchronous (CWAIT=NO) remote submit, the macro variable will not be set in the local session until a synchronization point. See “%SYSRPUT Statement” on page 29 for more details about synchronization points.

DMR

Invokes a remote version of the windowing environment for use with SAS/CONNECT.

Remote

Syntax

DMR

Details

The DMR system option must be specified either in the remote CONFIG.SAS file or on the SAS command line to invoke a remote SAS windowing environment session on the remote host.

The remote windowing environment receives input from the local SAS session and sends log and output lines to the local session's LOG and OUTPUT windows or files.

SASCMD=

Specifies the command to be used for invoking the remote SAS session for MP CONNECT.

Local

Syntax

SASCMD=“*SAS-command*”

Syntax Description

“*SAS-command*”

is the command that is used to invoke the remote SAS session for MP CONNECT, which is a remote session on a local host processor.

Details

SASCMD= can also be specified as an option in the SIGNON and the RSUBMIT statements and commands, whose values take precedence over those set in the SASCMD system option.

Note: If you need to execute additional host commands prior to the SAS invocation, it is recommended that you write a host-specific script that contains your host commands and the SAS invocation, and specify this script as the SASCMD value. △

The TCP/IP access method automatically appends the DMR= option to the SASCMD value. For details about DMR, see “DMR” on page 222.

TBUFSIZE=

Specifies the value of the default buffer size used for transmitting data.

Local and Remote

Syntax

TBUFSIZE=*value*

Syntax Description

value

specifies the default buffer size that SAS/CONNECT should use for transmitting data.

Note: The *value* specified must be a multiple of 1024. △

Details

The TBUFSIZE= system option specifies the default buffer size that SAS/CONNECT should use for transmitting data. This default setting can be overridden by setting the option TBUFSIZE=.

TBUFSIZE= should only be specified when using the program-to-program communications access methods, such as APPC, DECnet, NetBIOS and TCP/IP. The default buffer size is 32K. To get a buffer size of 16K, specify:

```
options tbufsize=16384;
signon;
```

The packet sizes of terminal-based access methods are limited by the screen size and cannot be overridden. Because the default value is the maximum buffer size allowed, the TBUFSIZE= option is ignored if it is specified by using terminal-based access methods, such as EHLLAPI and TELNET.

Note: This option is available in Version 7 and later. △

TRANTAB=

Specifies the translation tables that are used by various parts of the SAS System.

Local

Syntax

TRANTAB=(*translation-tables*)

Syntax Description

translation-tables

specifies the translation tables that are used by various parts of the SAS System.

Details

The TRANTAB= system option specifies a translation table for your SAS session or job and includes all file transfers. The process in which data is translated between a local SAS session and a remote SAS session has changed in Version 7 and later versions. However for compatibility, the translation process that was used in Version 6 has been retained.

- In Version 6, translation occurs twice for every transmission. The data is translated from local to transport format, and then the receiving side translates

from transport format to local format. Two translations occur for all data that is transferred.

- In Version 7 and later versions, when both the local and remote sides run Version 7 or later, the translation rules have changed. First, no translation occurs if both the remote and local sides have the same machine representations. This prevents any unnecessary translation from occurring. If translation is needed, the receiving side translates the data directly into its native representation. So when both the local and remote sides run Version 7 or later, the data is only translated once when translation is necessary, and is not translated at all when both hosts have the same machine architecture.

Due to this change in the translation process, the functionality of TRANTAB has also been changed. In Version 7 and later of SAS/CONNECT, you have the ability to provide a translation table for three possible translation points.

- On the input file's host when translating from the format of the file into the native format of the input host. You can specify a translation table on the first translation point by either specifying TRANTAB= as a data set option on the input file or by specifying the TRANIN option in the PROC UPLOAD or the PROC DOWNLOAD statement.
- On the output file's host when translating from the format of the received observations into the native format of the output host. The only way to specify the translation table for the second translation point is by specifying TRANNET= in the PROC UPLOAD or the PROC DOWNLOAD statement.
- On the output file's host when translating the observations from native format to some other specified format. This third point of translation only occurs when writing to a format other than the native host, that is, OUTREP must be specified. You can specify a translation table on the third translation point by either specifying TRANTAB= as a data set option in the output file or by specifying the TRANOUT= option in the PROC UPLOAD or the PROC DOWNLOAD statement.

For Version 6 style of translation, translation tables are specified in a parenthetical list that has eight positions. The position in which a table appears in the list determines the type of translation table that is being specified. Individual entries in the following list are separated by commas. See the following list of positions and types:

Position	Type of Translation Table
first	local-to-transport-format
second	transport-to-local-format
third	lowercase-to-uppercase
fourth	uppercase-to-lowercase
fifth	character classification
sixth	scanner translation
seventh	delta characters
eighth	scanner character classification

Note: The Version 6 style applies only when translating from Version 7 or Version 8 to an earlier version, or when the V6TRANSPORT option is used between a local and a remote session that uses a Version 7 or later version. Δ

The SAS System tries to locate the catalog entries that contain the translation tables first in SASUSER.PROFILE and then in SASUSER.HOST.

CAUTION:

Changing Translation Tables. Do not change translation tables unless you are familiar with their purpose. Translation tables are used internally by the SAS supervisor to implement National Language Support (NLS). If you are unfamiliar with the purpose of translation tables, do not change the specifications without technical advice. △

To change one table, specify null entries for the other tables. For example, to change the lowercase-to-uppercase table, which is third in the list, specify the following:

```
options trantab = ( , , new-uppercase-table);
```

The other tables remain unchanged. The output from PROC OPTIONS reflects the last specification for the TRANTAB option and not the composite specification. For example:

```
options trantab = ( , , new-uppercase-table);
options trantab = ( , , , new-lowercase-table);
```

PROC OPTIONS shows that the value for TRANTAB is (, , , *new-lowercase-table*), but both the *new-uppercase* and *new-lowercase* tables are in effect.

For more information about the TRANTAB procedure, see the *SAS Procedures Guide*.

The correct bibliographic citation for this manual is as follows: SAS Institute Inc., *SAS/CONNECT User's Guide, Version 8*, Cary, NC: SAS Institute Inc., 1999. pp. 537.

SAS/CONNECT User's Guide, Version 8

Copyright © 1999 by SAS Institute Inc., Cary, NC, USA.

ISBN 1-58025-477-2

All rights reserved. Printed in the United States of America. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, or otherwise, without the prior written permission of the publisher, SAS Institute Inc.

U.S. Government Restricted Rights Notice. Use, duplication, or disclosure of the software by the government is subject to restrictions as set forth in FAR 52.227-19 Commercial Computer Software-Restricted Rights (June 1987).

SAS Institute Inc., SAS Campus Drive, Cary, North Carolina 27513.

1st printing, September 1999

SAS® and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries.® indicates USA registration.

IBM®, AIX®, DB2®, OS/2®, OS/390®, RS/6000®, System/370™, and System/390® are registered trademarks or trademarks of International Business Machines Corporation. ORACLE® is a registered trademark or trademark of Oracle Corporation. ® indicates USA registration.

Other brand and product names are registered trademarks or trademarks of their respective companies.

The Institute is a private company devoted to the support and further development of its software and related services.