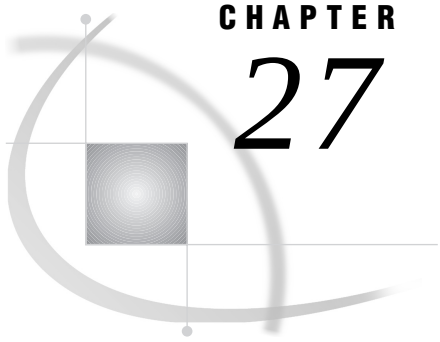

CHAPTER

27

Windows: NetBIOS Access Method

SAS Support for NetBIOS on Windows

You can use the NetBIOS communications access method with the Windows NT, Windows 95, Windows 98, and Windows 32s platforms.

Version 7 and later releases do not support the Windows 32s platform. However, information about Windows 32s is included here for Version 6 users.

Tasks That Are Common to SAS/CONNECT and SAS/SHARE

System Administrator, Network Administrator, Applications Programmer, or User

To use the NetBIOS access method with a Windows host for SAS/CONNECT and SAS/SHARE, perform these tasks:

1. Verify that you have met all your site and software requirements.
2. Verify that you know how to set options in SAS software.
3. Set the desired SAS/CONNECT and SAS/SHARE options.

System and Software Requirements for SAS/CONNECT and SAS/SHARE

Ensure that the following conditions have been met:

1. The NetBIOS application program interface (API) has been installed at both the local and remote hosts.
2. SAS software is installed on both the local and remote hosts.

Windows NT and Windows 95 Requirements

To use the NetBIOS access method with Windows NT and Windows 95, install the IBM compatible NetBIOS API that is included with Windows.

When configuring NetBIOS on a host, the system administrator binds the NetBIOS interface to a lower-level protocol, which is sometimes called the wire protocol. Examples of wire protocols are NETBEUI, IPX SPX, or TCP/IP. The wire protocol to which NetBIOS is bound is based on the type of network that your site uses. The IBM compatible NetBIOS interface is bound to a network that is running the NETBEUI wire protocol. The Novell compatible NetBIOS interface is bound to a network that is running the IPX/SPX wire protocol.

Windows 32s Requirements

To use the NetBIOS access method with Windows 32s, install the Windows interface to the IBM compatible NetBIOS that is loaded into DOS. Then, you should be able to use software from any vendor that supplies a NetBIOS product. The following two packages have been verified by SAS Institute:

- the IBM LAN Support Program
- the Novell Netware Requestor for DOS.

Setting SAS Options and Variables

You may need to set specific options to establish the connections that you want with SAS/CONNECT and SAS/SHARE when using the NetBIOS communications access method.

Consult with your network administrator to determine what options must be set and what values to assign to them.

You may specify an option in any of the following forms:

- in an OPTIONS statement in a SAS session or in an AUTOEXEC file:

  OPTIONS SET=*variable-name value*;

  Example:

  ```
  options set=vqmlinks 1;
  ```

- in a SAS configuration file or at SAS invocation:

  -SET *variable-name value*

  Example:

  ```
  -set vqmlinks 1
  ```

- as a SAS macro variable:

  %LET *variable-name=value*;

  Example:

  ```
  %let vqmlinks=1;
  ```

- as a DOS operating system environment variable:

  SET *variable-name=value*

  Example:

  ```
  set vqmlinks=1
  ```

Values for these options may contain up to eight characters, consisting of alphanumeric characters, the percent sign (%), the dollar sign ($), the pound sign (#), the at sign (@), and the underscore (_).

If you set multiple forms of the same option, here is the order of precedence that is followed:

SAS macro variable

OPTIONS statement

AUTOEXEC file

SAS invocation

SAS configuration file

DOS environment variable.

Setting Security for SAS/CONNECT and SAS/SHARE

Note: In the Windows environment, SAS/SHARE server security is supported on the Windows NT platform only. △

For SAS/CONNECT, you must supply identifying information to sign on without a script to a remote host running a spawner program. A SAS/SHARE server, running secured, requires identification from each connecting client. The next sections outline the alternatives for specifying security information for SAS/CONNECT and SAS/SHARE.

USER= and PASSWORD= Options in Selected Statements

In Version 8, you provide client identification to a SAS/CONNECT remote host or a SAS/SHARE server using the USER= and PASSWORD= options. These options are valid in the following statements:

SIGNON

RSUBMIT

LIBNAME

PROC SQL
Connect to Remote

PROC OPERATE
(in the PROC statement)
set server
stop server
quiesce server
start server
display server

Specifying client identification in the applicable security option (SASUSER= and SASPASS= for SAS/CONNECT and SASSECUR= for SAS/SHARE) is still accepted but is not recommended in Version 8. The USER= and PASSWORD= options take precedence over the client security option when both are specified. For example, a SAS/SHARE client's execution of a LIBNAME statement with values assigned to the USER= and PASSWORD= options would override a SASSECUR= setting in the same client SAS session.

CAUTION:
In order to make a SAS/SHARE server secured, the SASSECUR= option must be set at a SAS/SHARE server that can run on any host. △

Here is the syntax and definitions for these options:

USER | **USERNAME** | **USERID** | **UID**=*username* | _PROMPT_

PASS | **PASSWORD** | **PASSWD** | **PWD** | **PW**=*password* | _PROMPT_

Specifying these options allows a user on the local host whose username and password have been verified to access the remote host.

username
is a valid userid for the remote host and is thus host-dependent in form. If the value contains blanks or special characters, it must be enclosed in quotes. On Windows NT only, the username can also include the domain name, which locates the specified username in a domain.

password
is the password, if any, required for authentication of the supplied username. This value will not be echoed in the SAS log. If the value contains blanks or special characters, it must be enclosed in quotes.

PROMPT
specifies that the SAS System prompts the client for *username* and *password*.

Note: The values provided when prompted must NOT be quoted. △
Specifying USER=_PROMPT_ and omitting the PASSWORD= specification will cause SAS to prompt you for both userid and password.
This is especially useful for allowing the SAS statements containing the USER= and PASSWORD= options to be copied and otherwise effectively reused by others.

For SAS/SHARE, the values supplied for the USER= and PASSWORD= options are valid for the duration of the remote host connection. Additional accesses of the remote host while the connection to that host is still in-effect do not require re-supply of the

USER= and PASSWORD= options. For example while the first, connecting library assign to a SAS/SHARE server may require specification of the options, subsequent assigns to the same server will not heed specification of these options as long as the original connection is in-effect. A subsequent re-connect to the same server or connect to a different server would require re-supply of the USER= and PASSWORD= options.

Here is a Version 8 example for SAS/SHARE:

```
libname test 'prog2 a' user=joeblue password="2muchfun" server=share1;
```

For SAS/CONNECT, these values are valid until SIGNOFF.

Here is a Version 8 example for SAS/CONNECT:

```
signon rmthost user=joeblack password=born2run;
```

As a security precaution, PASSWORD= field entries echoed in the log are replaced with Xs. If _PROMPT_ was specified for entering the password, the entry would not be displayed on the screen as it is typed.

SAS/CONNECT SASUSER and SASPASS Options

SASUSER *userid*

SASPASS *password*

On the local host, either assign a valid userid and password to the SASUSER and SASPASS options or supply them to SAS, when prompted.

Consult with the system administrator of the remote host at which the spawner is invoked for a valid userid and password.

The SASUSER and SASPASS options store the userid and the password of the remote host that, when passed to the remote host, allow a local host connection.

Example:

```
options set=sasuser bass;
options set=saspass time2go;
```

See “Setting SAS Options and Variables” on page 388 for examples of the forms that you can use to specify the SASUSER and SASPASS options.

Also see Chapter 35, “PC Spawner Program,” on page 471 for information about the -SECURITY option in the PC spawner program, which controls the security of the remote host.

SAS/SHARE SASSECUR Option

You must set the SASSECUR option in order to pass a remote host userid and a password to a SAS/SHARE server for verification. After the userid and the password have been verified, the connection to the SAS/SHARE server can proceed.

SASSECUR=_NONE_ | _PROMPT_ | *userid.password* | _SECURE_

CAUTION:

Windows NT only SAS/SHARE server security is supported on the Windows NT platform only. △

NONE

must be set at the SAS/SHARE client.

Setting this value does not establish secure sessions for connecting SAS/SHARE clients.

This is the default.

PROMPT
must be set at the SAS/SHARE client.

PROMPT specifies that SAS prompt the user for userid and password information. When prompted for a password, the input field is not displayed. Choosing to prompt for a userid and a password provides more security than assigning the userid and the password to the system option.

userid.password
must be set at the SAS/SHARE client.

This value specifies both the userid and password. Assigning the userid and password directly to the SASSECUR option at the SAS/SHARE client may inadvertently publicize this information and compromise the security of the SAS/SHARE server. Assigning the value to the option in a file allows anyone to read it.

SECURE
must be set at the SAS/SHARE server on a Windows NT host only.

The _SECURE_ value for the SASSECUR option requires a SAS/SHARE client to supply a valid userid and password to the remote host or to the remote host on which the server is running in order to allow client access to the server.

Specify the SASSECUR option before you create a server.

Examples:

```
options set=sassecur _none_;
options set=sassecur _prompt_;
options set=sassecur bass.time2go;
options set=sassecur "apex\bass.time2go";
options set=sassecur _secure_;
```

See “Setting SAS Options and Variables” on page 388 for examples of the forms that you can use to specify SASSECUR.

SAS/CONNECT and SAS/SHARE Options

VQMLINKS *number-of-links*
specifies the number of links that can be active simultaneously. The default is 0.

For SAS/CONNECT, each time you sign on to a remote host, you initiate one link. If you want to sign on to more than one remote host during a single SAS session, set VQMLINKS to the number of links that will be active at the same time. There is no limit to the number of links that you can specify, but use the smallest number possible to conserve NetBIOS session resources. The number that you specify for this option must be the same as or less than the maximum number of sessions that are configured for NetBIOS when it is installed. If you specify 0, VQMLINKS defaults to the number of sessions that are configured for a single NetBIOS user.

At the SAS/CONNECT remote host, set both VQMLINKS and VQMCONVS to 1. Specify a higher value if you are accessing a SAS/SHARE server from your SAS/CONNECT remote session. Details about the VQMCONVS option are given later in this section.

At a SAS/SHARE server, set VQMLINKS to a value that represents the maximum number of clients that can be connected simultaneously. Specifying 0 implies that no limit is to be enforced and that the maximum is constrained only by system memory.

The server administrator should specify this value if you want to set a limit.

VQADAPTR *adapter-number*
for SAS/CONNECT and SAS/SHARE, specifies which network adapter, and, therefore, which network to use when establishing the link. You do not need this option if you are connected to only one network. The default is 0.

Note that if both the SAS/CONNECT local and remote hosts or a SAS/SHARE server and clients are connected to multiple networks, both hosts must specify the same network in order to establish a connection. For example, if your node has network connections for a Token Ring network and an Ethernet network and you want to connect to another node on the Ethernet network, you must set VQADAPTR to the correct adapter number for that network. This doesn't necessarily mean that the value of VQADAPTR is the same on both hosts. One host may have adapter 0 set to Ethernet while the other host has adapter 1 set to Ethernet. In this case, VQADAPTR must be 0 for one host and 1 for the other host.

Ask your PC installation staff or a SAS Software Representative for help to determine which adapter to use for each network.

For Windows NT hosts using SAS/SHARE, VQADAPTR specifies the logical adapter number as configured in the Main -> Control -> Panel -> Network -> Netbios Interface menu that matches the desired network route (driver and adapter combination).

Ask the system administrator for help to determine this value.

VQCAMLEN *access-method-buffer-and-packet-length*
specifies the access method buffer and packet length. This option determines the maximum number of characters that can be transmitted in a single packet. The value can range from 55 to 65535 characters. The default value is 4096.

Ask your system administrator for help to determine this value.

VQMCONVS *number-of-conversations*
specifies the number of conversations that can occur simultaneously. Each time that you sign on to a remote host, access a server, or access a new library, you initiate one conversation; therefore, set this value to at least the same number as VQMLINKS. There is no limit to the number of conversations that you can specify, but use the smallest number possible to conserve NetBIOS command resources. The number that you specify for this option must be the same or less than the number of commands that are configured for NetBIOS. If you specify 0, VQMCONVS defaults to the number of commands that are configured for a single NetBIOS user.

SAS/SHARE Only Option

VQPNAME *symbolic-user-name*
specifies the symbolic user name to be used by a SAS/SHARE server when referring to a user session in its SAS log and in output from the OPERATE procedure. This name can be any valid SAS name. See *SAS Language Reference: Dictionary* for information about SAS naming rules. The default name is the alphabetic character U followed by the last seven characters of the network hardware address.

SAS/CONNECT

Local Host Tasks

User or Applications Programmer

To connect a Windows local host to a remote host, perform these tasks at the local host:

1. Set a userid and a password, as necessary.
2. Specify the communications access method.
3. Specify a remote host to connect to.
4. Sign on to the remote host.

Setting the Remote Host Userid and Password

If the PC spawner program is running in secure mode, you must also set the remote host's userid and password at the local host. Set the -SECURITY option in the PC spawner invocation command to secure the server.

Set security at the local host using either of the methods explained in "Setting Security for SAS/CONNECT and SAS/SHARE" on page 389. For Version 8 security behavior, specify the USER= and PASSWORD= options in the SIGNON statement. For details, see "USER= and PASSWORD= Options in Selected Statements" on page 389.

For Version 7 security behavior, if you set the SASPASS and SASUSER options at the local host, either specify a userid and a password that are valid on the remote host or specify _PROMPT_ to supply the userid and password when connecting to a remote host. For information about setting the SASUSER and SASPASS option, see "SAS/CONNECT SASUSER and SASPASS Options" on page 391.

See Chapter 35, "PC Spawner Program," on page 471 for information about starting the spawner on the remote host.

Specifying the NetBIOS Communications Access Method

You must specify the NETBIOS communications access method to make a remote host connection. Use the following syntax:

```
OPTIONS COMAMID=access-method-id;
```

where COMAMID is an acronym for Communications Access Method Identification. *access-method-id* identifies the method used by the local host to communicate with the remote host. NetBIOS (an acronym for Network Basic Input/Output System) is an example of an *access-method-id*.

Example:

```
options comamid=netbios;
```

Alternatively, you may set this option at a SAS invocation or in a SAS configuration file.

Specifying the Remote Host Name

To make a connection from a Windows local host to a remote host, use the following syntax:

```
OPTIONS REMOTE=network-name;
```

where *network-name* is the -NETNAME option to the PC spawner program that was started on the remote host.

Example:

```
options remote=sasrem;
```

Alternatively, you may set this option at a SAS invocation or in a SAS configuration file.

See Chapter 35, "PC Spawner Program," on page 471 for more information about the -NETNAME option.

Signing On to the Remote Host

To complete your signon to the remote host, enter the SIGNON statement, as follows:

```
signon;
```

Note: Sign-on script files are not needed with the NetBIOS access method because the PC spawner program directly invokes the remote SAS session and replaces the need for a script file. △

Although no errors are produced if you specify a script file, you do waste processing time. If you defined the RLINK fileref before establishing a connection, when you sign on, SAS/CONNECT processes and loads the script file that is identified by the fileref, but the NetBIOS access method will ignore the script.

If you do not want to omit the RLINK fileref but you want to avoid wasting processing time, use the NOSCRIPT option in the SIGNON and SIGNOFF statements, as shown here:

```
options comamid=netbios  remote=sasrem;
signon noscript;
.
.
.
signoff noscript;
```

Local Host Example

The following example illustrates the statements that you specify in a Windows local host SAS session to connect to a remote host with the NetBIOS access method:

```
options set=vqmlinks 3 set=vqmconvs 3;
options comamid=netbios remote=sasrem;
signon user=_prompt_;
```

This example assumes a connection to a PC spawner that is running in secure mode. The NetBIOS communications access method is declared with a connection to the remote host SASREM. SASREM is the name that is specified in the -NETNAME option that the PC spawner uses to communicate with the local host. The USER= option in the SIGNON statement specifies that the connecting local host be prompted for a userid and a password that are valid on the remote host.

Remote Host Tasks

System Administrator

To allow a local host to make a remote host connection, perform these tasks at the remote host:

1. Start the PC spawner program.
2. Set several remote host options, as needed.

Starting the PC Spawner Program

You must invoke the PC spawner program on the Windows NT, Windows 95, or Windows 98 remote host to enable local hosts to connect to it. The spawner program resides on a remote host and listens for SAS/CONNECT client requests for connection to the remote host. After the spawner program receives a request, it invokes the remote SAS session.

For Windows NT only, setting the -SECURITY option in the PC spawner invocation command secures the spawner.

The spawner will then verify the userid and the password that are specified by means of the USER= and PASSWORD= options in the SIGNON statement.

See Chapter 35, “PC Spawner Program,” on page 471 for information about starting the spawner on the remote host.

Setting Options at the Remote Host

Although sign-on script files are not used for the NetBIOS access method, you may set remote host options at the remote host. It is recommended that you set these options:

NO$SYNTAXCHECK
: allows the continuation of statement processing at the remote host regardless of syntax error conditions.

  This option is valid as part of a configuration file, at a SAS invocation, or in an OPTIONS statement.

NOTERMINAL
: specifies whether a terminal is attached at SAS invocation. If NOTERMINAL is specified, requestor windows are not displayed.

  Setting NOTERMINAL at the remote host is advisable so that no terminal is associated with the remote session. This option prevents SAS from displaying error messages and dialog boxes on the remote host, which requires user intervention.

  This option is valid as part of a configuration file or a SAS invocation.

  See *SAS Language Reference: Dictionary* for details about this option.

NOXWAIT
: *Note:* applies only to OS/2 or Windows remote hosts. △

  specifies whether you have to type EXIT at the DOS prompt before the DOS shell closes. Setting NOXWAIT at the remote host is recommended to prevent SAS from displaying a dialog box on the remote host. Such a display requires that you explicitly type EXIT at the remote host and gives the appearance that the REMOTE SUBMIT command is hung.

  This option is valid as part of a configuration file, at a SAS invocation, or in an OPTIONS statement.

  See *SAS Companion for the Microsoft Windows Environment* for details about this option.

Remote Host Example

The following example illustrates the statements that you specify in a Windows NT or a Windows 95 remote host's configuration file to prepare for a connection from a supported local host with the NetBIOS access method:

```
-dmr
-comamid netbios
-no$syntaxcheck
-noterminal
-noxwait
```

An example follows of how the PC spawner is invoked on a Windows NT or a Windows 95 remote host:

```
c:\sas\connect\sasexe\spawner -comamid netbios -netname sasrem
                              -file mysas.cmd
```

The spawner is invoked and the NetBIOS access method is specified. The -NETNAME option specifies the name of the network (SASREM) that the PC spawner program uses to communicate with the local host. The -FILE option executes the MYSAS.CMD file, which invokes a SAS session.

See Chapter 35, "PC Spawner Program," on page 471 for information about the contents of a command file and executing the PC spawner. Options that are set through the spawner may override options that are set in a remote host configuration file.

SAS/SHARE

Client Tasks

CAUTION:

Windows NT only Server security is supported on the Windows NT platform only. △

System Administrator, User, or Network Administrator

To prepare for accessing a SAS/SHARE server, perform the following tasks:

1. For a Windows NT client only, assign the appropriate rights to each connecting client.
2. For a Windows NT client only, set security for connecting clients.
3. Specify the NetBIOS access method.
4. Specify a server name.

Assigning the Appropriate Rights for Connecting Clients

CAUTION:

Windows NT only Server security is supported on the Windows NT platform only. △

The account in which a connecting client runs must have the appropriate rights. To assign these rights

1. Click on the Administrative Tools icon.

2 Click on the User Manager icon.
3 From the Policies pull-down menu, select "User Rights."
4 Click the "Show Advanced User Rights" box.
5 Assign "Log on as a batch job" rights to the appropriate users.

Setting Secure Userids and Passwords for Connecting Clients

CAUTION:
In the Windows environment, server security is supported on the Windows NT platform only. △

Set security at the client using either of the methods explained in “Setting Security for SAS/CONNECT and SAS/SHARE” on page 389. For Version 8 security behavior, specify the USER= and PASSWORD= options in the appropriate statement. For details, see “USER= and PASSWORD= Options in Selected Statements” on page 389.

For Version 7 security behavior, if you set the SASSECUR option at the client, either specify a userid and a password that are valid on the server or specify _PROMPT_ to supply the userid and password when connecting to a server. For information about setting the SASSECUR option, see “SAS/SHARE SASSECUR Option” on page 391.

For Windows NT only that runs Version 8, you may qualify *username* in the form *Windows-NT-domain-name\username*. Here is an example of how you might specify this information in the LIBNAME statement in SAS/SHARE :

```
libname test 'prog2 a' user="apex\bass.time2go" server=share1;
```

Domain name **apex** identifies the location of the username and password database. Username **bass** and password **time2go** will be verified against those in the identified domain's username and password database.

To set up a secure server, you must assign a userid and a password that are valid on the server's host. Once set up, connecting clients must use valid userids and passwords to connect to the server. See “SAS/SHARE SASSECUR Option” on page 391 for information about setting the SASSECUR option.

Specifying the NetBIOS Access Method

You must specify the NetBIOS access method at each connecting client before you can access a server. Use the following syntax:

```
OPTIONS COMAMID=access-method-id;
```

where COMAMID is an acronym for Communications Access Method Identification. *access-method-id* identifies the method used by the client to communicate with the server. NetBIOS (an acronym for Network Basic Input/Output System) is an example of an *access-method-id*.

Example:

```
options comamid=netbios;
```

The server is accessed using the NETBIOS access method.

You may specify the COMAMID option in an OPTIONS statement, at a SAS invocation, or in a SAS configuration file.

Additionally, you may use the COMAUX1 and COMAUX2 options to designate auxiliary communications access methods. See Table 1.2 on page 9 for the supported access methods by host.

If the first method fails to access a server, the second method is attempted, and so on. You can specify up to two auxiliary access methods, depending on the number of methods that are supported between client and server hosts.

COMAUX options can be specified only at SAS invocation or in a SAS configuration file. The syntax for the COMAUX options follows:

```
-COMAUX1 alternate-method
-COMAUX2 alternate-method
```

An example of SAS configuration file entries for an OS/2 client that is connecting to a Windows NT server follows:

```
-comamid netbios
-comaux1 tcp
-comaux2 appc
```

If the server cannot be reached using the NetBIOS access method, a second attempt is made with the TCP/IP access method, and then with the APPC method.

Specifying a Server Name

You must specify the server's identifier on the LIBNAME and PROC OPERATE statements as follows:

```
SERVER=identifier
```

Follow standard SAS naming rules when defining a server name. See *SAS Language Reference: Dictionary* for details about SAS naming rules. See *SAS/SHARE User's Guide* for details about the LIBNAME and PROC OPERATE statements.

Example:

```
server=share1;
```

Client Example

The following example illustrates the statements that you specify in a Windows NT client configuration file that are used to access a server with the NetBIOS access method:

```
-set vqmlinks 1
-set vqadaptr 0
```

See “Setting SAS Options and Variables” on page 388 for details about these options.

The following example shows the statements that are specified in a Windows NT client session:

```
options comamid=netbios;
libname sasdata 'c:\edc\prog2\sasdata' user=_prompt_ server=share1;
```

The NetBIOS access method is declared. The LIBNAME statement specifies the data library that is accessed through the server SHARE1 by means of a prompt for a username and a password that are valid on the server.

Server Tasks

Server Administrator

Note: Server security is supported on the Windows NT platform only. △

To set up a secure server and to make it accessible to a client, perform the following tasks:

1. Assign the appropriate rights for a secure server for Windows NT only.
2. Require only validated clients to access a secure server for Windows NT only.
3. Set NetBIOS access method security for Windows NT only.
4. Specify the NetBIOS access method.
5. Specify the server name.

Assigning the Appropriate Rights for a Secure Server

CAUTION:

Windows NT only This process is supported on the Windows NT platform only. △

The account in which a secure server runs must have the appropriate rights. To assign these rights

1. Click on the Administrative Tools icon.
2. Click on the User Manager icon.
3. From the Policies pull-down menu, select "User Rights."
4. Click on the "Show Advanced User Rights" box.
5. Assign "Act as part of the operating system" rights to the appropriate users.

Setting NetBIOS Access Method Security

CAUTION:

Windows NT only This procedure is supported on the Windows NT platform only. △

Before you can create a secure SAS/SHARE server, you must make the access method secure by assigning the _SECURE_ value to the SASSECUR option. See "SAS/SHARE SASSECUR Option" on page 391 for information about setting the SASSECUR option.

Specifying the NetBIOS Access Method at the Server

You must specify the NetBIOS communications access method at the server before you can create and access a SAS/SHARE server. Use the following syntax:

```
OPTIONS COMAMID=access-method-id;
```

where COMAMID is an acronym for Communications Access Method Identification. *access-method-id* identifies the method used by the server to communicate with the client. NetBIOS (an acronym for Network Basic Input/Output System) is an example of an *access-method-id*.

For a server that is running on a host on which only one communications access method is available, use only the COMAMID option.

Example:

```
options comamid=netbios;
```

The server will be available only to SAS/SHARE sessions that use the NetBIOS access method.

You may specify the COMAMID option in an OPTIONS statement, at a SAS invocation, or in a SAS configuration file.

However, if the host on which a server is running supports multiple access methods, you may specify up to two auxiliary access methods by which clients may access the server by using the COMAUX1 and COMAUX2 options. See Table 1.1 on page 8 for the supported access methods by host.

All of the access methods initialize when the server initializes. The activation of multiple access methods makes a server available to several groups of clients, each using a different communications access method simultaneously.

COMAUX options can be specified only at a SAS invocation or in a SAS configuration file. The syntax for the COMAUX options follows:

```
-COMAUX1 alternate-method
-COMAUX2 alternate-method
```

An example of configuration file entries for a server that is running on a Windows NT host follows:

```
-comamid netbios
-comaux1 tcp
-comaux2 appc
```

When the server starts, all of the communications access methods are initialized. The server is simultaneously available to client sessions that use the NetBIOS access method as well as to clients that use the TCP/IP and APPC access methods.

Specifying a Server Name

Specify the server identifier in the PROC SERVER statement as follows:

```
SERVER=server-id
```

Example:

```
server=share1;
```

Follow standard SAS naming rules when defining a server name. See *SAS Language Reference: Dictionary* for details about SAS naming rules. See *SAS/SHARE User's Guide* for details about the LIBNAME and PROC OPERATE statements.

Server Example

The following example illustrates the statements that you specify in a configuration file on the Windows NT host at which you start a server:

```
-set vqmlinks 1
-set vqadaptr 0
```

See “Setting SAS Options and Variables” on page 388 for details about these options.

Specify the following statements in a SAS session on the Windows NT remote host to start a server:

```
%let sassecur=_secure_;
options comamid=netbios;
proc server id=share1 authenticate=req;
```

```
run;
```

The first line uses the SAS macro variable SASSECUR to prompt clients for a userid and a password that are valid on the server. The NetBIOS access method is declared for the *server* SHARE1 that is started on a Windows NT remote host.The additional options in the PROC SERVER statement allow only validated clients to access the server.

The correct bibliographic citation for this manual is as follows: SAS Institute Inc., *Communications Access Methods for SAS/CONNECT and SAS/SHARE Software, Version 8*, Cary, NC: SAS Institute Inc., 1999. pp. 643.

Communications Access Methods for SAS/CONNECT and SAS/SHARE Software, Version 8

ISBN 1–58025–479–9

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

1st printing, September 1999