CHAPTER

16

OS/2: SPX Access Method

SAS Support for SPX on OS/2 **233**

Tasks That Are Common to SAS/CONNECT and SAS/SHARE **233**

- System and Software Requirements for SAS/CONNECT and SAS/SHARE **234**
- Setting SAS Options and Variables **234**
- SAS/CONNECT and SAS/SHARE SPXMSGSIZE Option **235**

SAS/CONNECT **235**

- Local Host Tasks **235**
- Setting the Remote Host Userid and Password **236**
- Specifying the SPX Communications Access Method **236**
- Specifying the Remote Host Name **236**
- Signing On to the Remote Host **236**
- Local Host Example **237**
- Remote Host Tasks **237**
- Starting the PC Spawner Program **237**
- Setting Options at the Remote Host **238**
- Remote Host Example **238**

SAS/SHARE **239**

- Client Tasks **239**
- Specifying the SPX Communications Access Method **239**
- Specifying a Server Name **239**
- Client Example **240**
- Server Tasks **240**
- Specifying the SPX Access Method at the Server **240**
- Specifying the Server Name **241**
- Server Example **241**

SAS Support for SPX on OS/2

Note: Beginning with Version 7, the SPX access method is not supported. Information about SPX is included here for Version 6 users. △

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

System Administrator or User

To use the SPX access method with an OS/2 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 SAS/CONNECT and SAS/SHARE options that you want.

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

Ensure that the following conditions have been met:

1. SPX has been installed at both the local and remote hosts at your site.
2. SAS software has been installed on both the local and remote hosts.
3. Release 3.0 or a subsequent release of OS/2 has been installed.
4. Version 2.10 or a subsequent version of the Novell NetWare Requestor for OS/2, Version 2.10 or a subsequent release, has been installed.

Setting SAS Options and Variables

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

Consult with your network administrator for advice about these settings.

You may specify an option in several forms, as follows:

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

OPTIONS SET=*variable-name value*;

Example:

```
options set=spxmsgsize 1514;
```

□ in a SAS configuration file or at SAS invocation:

-set *variable-name value*

Example:

```
-set spxmsgsize 1514
```

□ as a DOS operating system environment variable:

SET *variable-name*=*value*

Example:

```
SET spxmsgsize=1514
```

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:

OPTIONS statement

AUTOEXEC file

SAS invocation

SAS configuration file

DOS environment variable.

SAS/CONNECT and SAS/SHARE SPXMSGSIZE Option

SPXMSGSIZE *size-of-input/output-buffer*

specifies the size of the SAS program's input and output buffer. The range of acceptable values for this option is from 298 bytes to the maximum packet size that your network allows; for example, 1514 bytes on an Ethernet network and 4202 bytes on a Token Ring network.

Be aware that your selection should not violate the buffer size limitations of the Novell Requestor. The OS/2 requestor defaults to a buffer size of 1514 bytes, and the SPX access method also defaults to a buffer size of 1514 bytes. Because the size of the buffer is limited to 576 bytes when using a bridge, if you use SPX over a bridge, you must explicitly define a 576 buffer size for SPX and the Novell Requestor. You set the buffer size for SAS software and the Novell Requestor as follows:

SAS software

set the SPXMSGSIZE option to 576.

Novell Requestor

add a BUFFERS statement under the link support attribute in the NET.CFG file, as follows:

```
link support
buffers 30 576
```

The number 30 refers to the number of buffers; the number 576 refers to the size of each buffer, in bytes. The product of these numbers cannot exceed 64K bytes (1,024 bytes = 1K byte).

The maximum buffer size for Ethernet is 1514 bytes, and the maximum buffer size for Token Ring is 4202 bytes. If you are using a Token Ring network, you may want to change the BUFFERS statement in your NET.CFG file to a 4202-byte buffer size and set the SPXMSGSIZE option to 4202.

Ask your network administrator for advice about setting this option.

SAS/CONNECT

Local Host Tasks

User or Applications Programmer

To connect an OS/2 local host to a remote host, perform these tasks:

1. If connecting to a spawner that is running secured, set a userid and a password to ensure security at the remote host, as required.
2. Specify the communications access method.
3. Specify a remote network name.
4. Sign on to the remote host.

Setting the Remote Host Userid and Password

If you are connecting to an OS/2 remote host by means of a PC spawner program with the -PROTECTION option, you must either assign a valid remote host userid and a password to the SASUSER and SASPASS options or supply them to SAS, when prompted. Set the userid and the password in the following form:

```
OPTIONS SET=SASUSER userid;
OPTIONS SET=SASPASS password;
```

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

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

Specifying the SPX Communications Access Method

You must specify the SPX 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. SPX (an abbreviation for the Novell Sequenced Packet Exchange protocol) is an example of an *access-method-id*.

Example:

```
options comamid=spx;
```

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 an OS/2 local host to a remote host, use the following syntax:

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

where *network-name* is the NETNAME argument to the PC spawner program that you start on the remote host. The values for the REMOTE= option and the NETNAME argument to the PC spawner must be identical. See “Starting the PC Spawner Program” on page 237 for details about specifying the NETNAME argument when starting the PC spawner.

Example:

```
options remote=mynet;
```

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

Signing On to the Remote Host

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

```
signon;
```

Note: Sign-on script files are not needed on an OS/2 local host because the SPX access method connects to the spawner program that runs on the remote host. The spawner program then invokes the remote SAS session. This process 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 identified by the fileref but the SPX access method will ignore the script.

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

```
signon noscript;
.
.
.
signoff noscript;
```

Local Host Example

The following example illustrates the statements that you specify in an OS/2 local host SAS session to connect to a remote host with the SPX access method.

```
options set=sasuser userid set=saspass password;
options set=spxmsgsize 4202;
options comamid=spx remote=sasrem;
signon;
```

This example assumes a connection to a PC spawner that is running in secure mode. The SASUSER and SASPASS options allow the userid and the password to be passed to the remote PC spawner, which permits a connection. The SPXMSGSIZE option is set. See “SAS/CONNECT and SAS/SHARE SPXMSGSIZE Option” on page 235 for more information. The SPX access method is declared with a connection to the remote host SASREM, which is the name of the network that is specified in the -SPXNAME option at the PC spawner invocation. The SIGNON statement performs the sign-on process.

Remote Host Tasks

System Administrator

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

1 Start the PC spawner.

2 Set several remote host options.

Starting the PC Spawner Program

You must invoke the PC spawner program on the OS/2 remote host to enable local hosts to connect to it. The spawner program resides on a remote host, listening for SAS/CONNECT client requests for connection to the remote host. After the spawner program receives a request, it invokes the remote SAS session. See “Starting the PC Spawner Program” on page 471 for more information.

Setting Options at the Remote Host

Although sign-on script files are not used for the SPX 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.

NO$SYNTAXCHECK 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.

NOTERMINAL is valid as part of a configuration file or at a SAS invocation.

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

NOXWAIT
: 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.

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

See the *SAS Companion for the OS/2 Environment* for details about this option.

Remote Host Example

The following example illustrates the statements that you specify in an OS/2 remote host's configuration file to prepare for a connection from a supported local host with the SPX access method:

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

An example for invoking the PC spawner on an OS/2 remote host follows:

```
c:\sas\connect\sasexe\spawner -comamid spx
  -protection -spxname sasrem -file mysas.cmd
```

The spawner is invoked and the SPX access method is specified. The -PROTECTION option verifies the userids and the passwords of connecting clients. The -SPXNAME option specifies the name of the network 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 “Starting the PC Spawner Program” on page 471 for information about the contents of a command file and about executing the PC spawner. Options that are set through the spawner override options that are set in a remote host configuration file.

SAS/SHARE

Client Tasks

User or Applications Programmer

To prepare for creating and accessing a SAS/SHARE server, you must know how to:

1. Specify the SPX access method.
2. Specify a server name.

Specifying the SPX Communications Access Method

Specify the SPX communications access method at each connecting client before you 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* is an abbreviation for the method used by the client to communicate with the server. SPX (an acronym for Sequenced Packet Exchange) is an example of an *access-method-id*.

Example:

```
options comamid=spx;
```

The server is accessed using the SPX 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 1.3 on page 10 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 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 an OS/2 client connecting to an OS/2 server follows:

```
-comamid spx
-comaux1 tcp
-comaux2 netbios
```

If the server cannot be reached using the SPX method, a second attempt is made with the TCP/IP access method. Finally, if necessary, the NetBIOS method is used.

Specifying a Server Name

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

```
SERVER=identifier
```

Select the name that you want for a server, following standard SAS naming rules. See *SAS Language Reference: Dictionary* for details about SAS naming rules. See *SAS/SHARE User's Guide* for details about the PROC OPERATE and LIBNAME statements.

Example:

```
server=share1
```

Client Example

The following example illustrates the statements that you specify in an OS/2 client session that are used to access a server with the SPX access method:

```
options comamid=spx;
libname sasdata 'c:edc\prog2\sasdata' server=share1;
```

The SPX access method is declared. The LIBNAME statement specifies the data library that is accessed through the server SHARE1.

Server Tasks

Server Administrator

To make a server accessible to a client, perform the following tasks:

1 Specify the SPX access method as the server.

2 Specify the server name.

Specifying the SPX Access Method at the Server

Specify the SPX access method at the server before you create 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* is an abbreviation for the method used by the server to communicate with the client. SPX (an abbreviation for the Novell Sequenced Packet Exchange) 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.

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. See Table 1.3 on page 10 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 option follows:

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

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

```
-comamid spx
-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 SPX access method as well as to clients that use the TCP/IP and APPC access methods.

Specifying the Server Name

You must specify the server identifier in the PROC SERVER statement as follows:

```
SERVER=identifier
```

Select the name that you want for a server, following standard SAS naming rules. See *SAS Language Reference: Dictionary* for details about SAS naming rules. See the *SAS/SHARE User's Guide* for details about the PROC SERVER statement.

Example:

```
server=share1
```

Server Example

The following example illustrates the statements that you specify in a configuration file on the OS/2 host at which you start a server:

```
-set spxmsgsize 4202
```

See “SAS/CONNECT and SAS/SHARE SPXMSGSIZE Option” on page 235 for details about this option. The following statements in a SAS session on the OS/2 remote host illustrate how to start a server:

```
options comamid=spx;
proc server id=share1;
run;
```

The SPX access method is declared, and the server SHARE1 is started on the OS/2 remote host.

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

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