CHAPTER

13

The OPERATE Procedure

Introduction **119**
Syntax **120**
Syntax Descriptions **120**
Library Management Commands **121**
  Allocating a Library to a Server That Is Already Running **121**
  Displaying Information about a Library **122**
  Freeing a Library **123**
  Quiescing a Library **124**
  Re-Starting a Library **124**
  Stopping a Library **125**
  Specifying a Host-Specific SAS Data Library **125**
Server Management Commands **126**
  Displaying Information about a Server **126**
  Quiescing a Server **126**
  Setting the Current Server **127**
  Re-Starting a Quiesced Server **128**
  Stopping a Server **128**
User Management Commands **128**
  Displaying Information about a User **128**
  Quiescing a User's Access to a Server **130**
  Re-Starting a Quiesced or a Stopped User **130**
  Terminating a User's Connection to a Server **130**
Specifying a Server **131**
Specifying a Server-Access Password **131**
Specifying a User **131**

Introduction

Use the OPERATE procedure to manage the server, its libraries, and its users. For example, using PROC OPERATE, you can

□ define a SAS data library to a server after the server has already started

□ display information about assigned libraries

□ release libraries from assignment

□ terminate access to a library

□ display userids of users who are connected to the current server

□ manage the server from a session other than the server session.

The OPERATE procedure can be used in any of the SAS methods of processing: noninteractive mode, interactive line mode, display manager mode, or batch mode.

The PROC OPERATE statement accepts commands that enable you to manage

- □ server libraries
- □ servers
- □ server users.

The OPERATE procedure is an interactive procedure; that is, its statements are executed as they are encountered. For this reason, statements used in the OPERATE procedure are called commands. The OPERATE procedure executes until it is terminated by a QUIT or a RUN command. The syntax for these commands is discussed in later subsections. The syntax for the PROC OPERATE statement follows.

Syntax

PROC OPERATE <*options*>;

Library Management Commands

Server Management Commands

User Management Commands

Syntax Descriptions

where *options* can be one of the following:

ID=*serverid*
SERVERID=*serverid*
SERVER=*serverid*

identifies the default server session to be managed.

If the SERVERID= option is not specified, you must identify the server in the SET SERVER command or in those PROC OPERATE commands that allow you to identify the server to be managed. For details, see “Specifying a Server” on page 131.

PRINTFILE=LOG|PRINT
PF= LOG|PRINT

directs the output from the OPERATE procedure. Specifying PRINTFILE=LOG directs the output to the SAS log. The default is LOG.

Specifying PRINTFILE=PRINT directs the output to the procedure output file or output window.

SAPW=*password*

specifies a server-access password, which must be in the form of a valid SAS name. This password is required to access a SAS server that is executing with the SERVER procedure OAPW= option in effect. The value that you specify for this option is replaced by Xs in the SAS log.

USER=*username*| _PROMPT_

specifies the userid of the accessing client on the server. The host on which the client runs can also affect username conventions. For details about username conventions imposed by the host, see *Communications Access Methods for SAS/CONNECT and SAS/SHARE Software.*

Nicknames for USER are USERNAME, USERID, and UID.

Valid values that can be assigned to USER are:

username
For details about a valid username, see “Username and Password Naming Conventions” on page 106.

PROMPT
a secure method, specifies that SAS prompt the user for a valid username.

PASSWORD=*password* | _PROMPT_
specifies the password of the accessing client on the server. The host on which the client runs can also affect password naming conventions. For details about password naming conventions imposed by the host, see *Communications Access Methods for SAS/CONNECT and SAS/SHARE Software.*

Nicknames for PASSWORD are PASSWD, PASS, PWD, and PW.

Valid values for PASSWORD are:

password
For details about a valid password, see “Username and Password Naming Conventions” on page 106.

PROMPT
a secure method, specifies that SAS prompt the user for a valid password.

Library Management Commands

For Version 7 and later releases, multiple users can define different SAS data libraries with the same libref name. Version 7 and later release enhancements enable the server engine to differentiate among multiple user sessions that may use the same libref and to manage them appropriately.

Allocating a Library to a Server That Is Already Running

You define a SAS data library to a server that is already running by using the ALLOCATE LIBRARY command. The syntax for this command is

ALLOCATE LIBRARY *libref* <*'SAS-data-lib'*> <RENGINE=*engine-name*> <*engine/host options*>;

You may use the abbreviation ALLOC or AL in place of ALLOCATE.

You can define one SAS data library in each ALLOCATE LIBRARY command. The following arguments and options may be specified in the ALLOCATE LIBRARY command:

libref
is used by the server to refer to the SAS data library that is specified in the *SAS-data-lib* argument.

'SAS-data-lib'
is the physical name of the SAS data library that is being defined to the server. The name is host-specific and must be enclosed in either single or double quotes. See “Specifying a Host-Specific SAS Data Library” on page 125 for host-specific examples.

RENGINE=*engine-name*
specifies the engine to be used to process the SAS data library in the server's SAS execution. Specify this option only to override the SAS default for a library in the

server's execution or to reduce search time while the server determines which engine to use to access a particular SAS library. Ordinarily, this option should not be specified because the server determines which engine to use to process the data library.

engine/host-options

are options that apply to the SAS data library. You can specify any option that is valid in the LIBNAME statement for a particular operating system and engine. One or more options in the form *keyword=value* may be specified. You can specify as many options as you need. Use a blank space to separate options. This argument is not required.

These options are effective in the server SAS session, not in the user SAS session. See the SAS documentation for your operating environment for a complete list of the options that are available for your operating environment and engine.

Displaying Information about a Library

You display information about one or more libraries by using the DISPLAY LIBRARY command. The syntax for this command is

DISPLAY LIBRARY *libid-1*<. . . *libid-n*>;

DISPLAY LIBRARY _ALL_ ;

You may use the abbreviation DISP or D in place of DISPLAY.

This command displays information about one or more server libraries that are defined to the current server. The information includes the server libref, the physical name of the data library, its status, and the number of users that are accessing it.

The DISPLAY LIBRARY command reports summary information for a server library that is defined by a host-specific external method only after the library becomes active. A library becomes active after the user issues a LIBNAME statement to access a server library. For information about methods to define libraries to the server, see "Pre-Defining SAS Data Libraries to the Server" on page 30.

The *libid* argument is either a libref for a pre-defined server library or a physical name for a server library. If the _ALL_ argument is specified, the DISPLAY LIBRARY command provides summary information about each server library that is currently defined to the server.

For example, the following

```
proc operate serverid=share1;
   display library _all_;
```

produces a table such as this:

```
                                NUMBER
LIBREF          STATUS          OF USERS  LIBRARY NAME
------------------------------------------------------
DATALIB        QUIESCED            1       SAS-data-lib
POINT          ACTIVE              6       SAS-data-lib
POINTS         ACTIVE              4       SAS-data-lib
MAIN           STOPPED             0       SAS-data-lib
MAPS           INACTIVE            0       SAS-data-lib
```

If you specify one or more *libids* instead of _ALL_, the DISPLAY LIBRARY command displays summary information that is followed by detailed information for each libref that is active or quiesced. For example, the following

```
display library 'SAS-data-lib';
```

produces information such as this:

```
                        NUMBER
LIBREF      STATUS      OF USERS    LIBRARY NAME
------------------------------------------------------
POINTS      ACTIVE         4        SAS-data-lib
```

These users are accessing library ’*SAS-data-lib* ’:

```
USER              USER LIBREF
-----------------------------
JOHN(1)           SPACE
MARY(2)           ROW
IAN(5)            FEES
NORMAN(11)        JOBS
```

These data sets in library ’*SAS-data-lib*’ are active:

```
MEMBER   TYPE     STATUS   USER    OPEN MODE   USER
                                             LIBREF
---------------------------------------------------
USAGE    DATA     ACTIVE   JOHN(1) UPDATE     SPACE
USAGE    DATA     ACTIVE   MARY(2) INPUT      ROW
PROD     DATA     ACTIVE   IAN(5)  INPUT      FEES
MODULE   CATALOG  ACTIVE   JOHN(1) OUTPUT     SPACE
```

The column LIBREF (in the first of the three screens) contains the server libref for a Version 7 or Version 8 server. The server libref is the name a server administrator assigns to the library by using one of the following methods:

- □ a LIBNAME statement specified before the PROC SERVER statement
- □ an ALLOCATE LIBNAME command in a PROC OPERATE statement
- □ a host-dependent external allocation.

A library that is not defined using one of these methods does not have a server libref. Therefore, adminiinistrative commands that subsequently refer to that library must use the host-specific library name instead of, for example, a UNIX pathname. However, a Version 6 server always reports a server-defined libref that is valid for references to libraries in subsequent commands.

The column USER LIBREF (in the final screen) contains the libref that is specified by the user in the LIBNAME statement. The user’s libref is provided only as an aid for communicating with the user, if necessary.

The member STATUS is always active. The column OPEN MODE indicates whether the user is currently executing a SAS program step to read, update, or create the member.

Freeing a Library

You free (or release) one or more server-defined libraries by using either:

- □ an ALLOCATE LIBRARY command
- □ a LIBNAME statement that is specified before the PROC SERVER statement in the SAS program that created the SAS server.

The syntax for the FREE LIBRARY command is

FREE LIBRARY *libid-1* <. . . *libid-n*>;

FREE LIBRARY _ALL_;

You may use the abbreviation FR in place of FREE.

The status of the library at the time that the FREE command is issued determines whether the library is freed immediately. When the FREE LIBRARY command is issued, a library that is not in use is freed immediately; a library that is in use is freed after it is no longer in use.

The *libid* argument is either a libref for a pre-defined server library or a physical name for a server library. The command FREE LIBRARY _ALL_ frees all SAS libraries that are defined to the SAS server by using an ALLOCATE LIBRARY command or the LIBNAME statement.

In contrast, the QUIESCE LIBRARY and STOP LIBRARY commands curtail the use of an active library. To bring a library to a stopped status gradually, issue the QUIESCE LIBRARY command. To bring the library to a stopped status immediately, issue the STOP LIBRARY command. See descriptions of these commands elsewhere in this section.

Quiescing a Library

You quiesce a library with the QUIESCE LIBRARY command to gradually terminate access to a library by denying new requests to use the library. That is, you use the command to move the library from an active status to a stopped status. The syntax for the QUIESCE LIBRARY command is

QUIESCE LIBRARY *libid-1* < . . . *libid-n*>;

QUIESCE LIBRARY _ALL_;

You may use the abbreviation QUI or Q in place of QUIESCE.

This command quiesces one or more of the libraries that are defined to the current server. It immediately stops libraries that do not currently have members open.

The *libid* argument is either a libref for a pre-defined server library or a physical name for a server library. The _ALL_ argument quiesces all the libraries that are defined to the server.

When a library is quiesced, users who are currently executing DATA or PROC steps are allowed to complete those steps. When a step finishes executing, the user cannot invoke another step to re-open the member that was closed in the previous step or to open another member in the quiesced library. When the user is closed out of all members, the server library is released.

If a user attempts to access a quiesced library by using a LIBNAME statement, the attempt fails.

After all users are released from a quiesced library, the library is stopped. If the library is user defined, after all users are released, the library is stopped and is no longer defined to the server.

Specifying a physical name for a library prevents the library from being accessed even though it is not defined to the server when the QUIESCE LIBRARY command executes.

Re-Starting a Library

You start a library by using the START LIBRARY command to re-start one or more server libraries that have been stopped or quiesced. Because server libraries are available by default, this command is necessary to undo the effect of a previous STOP LIBRARY or a QUIESCE LIBRARY command. The syntax for the START LIBRARY command is

START LIBRARY *libid-1* < . . . *libid-n*>;

START LIBRARY _ALL_;

You may use the abbreviation ST in place of START.

The *libid* argument is either a libref for a pre-defined server library or a physical name for a server library. Specifying the _ALL_ argument re-starts all server libraries that are quiesced or stopped.

If a library that was defined by a user has been stopped and is then re-started with the START LIBRARY command, the library becomes unknown to the server.

Stopping a Library

You stop a library by using the STOP LIBRARY command to immediately terminate access to a library. The syntax for the STOP LIBRARY command is

STOP LIBRARY *libid-1* < . . . *libid-n*>;

STOP LIBRARY _ALL_;

This command immediately terminates all user access to one or more server libraries and brings the libraries to a stopped status. Subsequent attempts to access a stopped library by using a LIBNAME statement are denied.

The *libid* argument is either a libref for a pre-defined server library or a physical name for a server library. If the _ALL_ argument is specified, the server stops all libraries that are defined to the server.

If users are currently reading from and writing to members in the specified library, the server closes the members and releases each user's access to the library. If users are in the process of updating a data set, updates may be lost.

Specifying a physical name for a library prevents the library from being accessed, even though it is not defined to the server when the STOP LIBRARY command executes.

Specifying a Host-Specific SAS Data Library

Throughout this section are examples of library commands that contain the *SAS-data-library* argument, which is specified according to host type. In the library command examples, for each instance of the *SAS-data-library* argument, see the following list of host-specific examples:

CMS
’STATS AREA2’

OpenVMS
’DISK1:[AREA2.WEATHER.STATS]’

OS/2
’G:\AREA2\WEATHER\STATS’

OS/390
’AREA2.WEATHER.STATS’

UNIX
'/area2/weather/stats'

Windows
’G:\AREA2\WEATHER\STATS’

Server Management Commands

Displaying Information about a Server

You display information about a single server by using the DISPLAY SERVER command. The syntax for this command is

DISPLAY SERVER;

DISPLAY SERVER *serverid* <(SAPW=*password*)>;

DISPLAY SERVER *serverid* </ SAPW=*password*>;

You may use the abbreviation DISP or D in place of DISPLAY.

This command displays summary information about the server identified by the *serverid* argument. If you do not specify a serverid, this command displays summary information about the current server. For information about specifying a serverid, see "Specifying a Server" on page 131. For information about specifying the SAPW= option, see "Specifying a Server-Access Password" on page 131.

An example of using the DISPLAY SERVER command follows:

```
proc operate;
   display server share1;
```

This command displays information about the server SHARE1.

Alternatively, you can use the SERVERID= option in the PROC OPERATE statement to identify the default server. The default server is also the current server unless you use the SET SERVER command to identify a different current server. For example, the following statements display information about the default server SHARE1, which is also the current server:

```
proc operate serverid=share1;
   display server;
```

Quiescing a Server

You quiesce a server by using the QUIESCE SERVER command, which terminates the server process by gradually releasing libraries and users and by denying new requests to access libraries through the server. That is, it moves the server from an active status to a stopped status. The syntax for the QUIESCE SERVER command is

QUIESCE SERVER;

QUIESCE SERVER *serverid* <(SAPW=*password*)>;

QUIESCE SERVER *serverid* </ SAPW=*password*>;

You may use the abbreviation QUI or Q in place of QUIESCE.

This command gradually terminates the server that is identified by the *serverid* argument. If you do not specify a serverid, this command gradually terminates the current server.

A quiesced server accepts only PROC OPERATE commands. If a user attempts to access server libraries or open SAS files through a quiesced server by using a LIBNAME statement, the attempt fails.

When you quiesce a server, current DATA and PROC steps can continue to access files that are already open but cannot perform any further opens. Subsequent DATA and PROC steps cannot access any server files, and users cannot re-open members that are closed or open other members. After a user has closed all members in a server library, the server releases the library. If this is the only server library that the user has accessed, that user is disconnected from it.

The QUIESCE SERVER command does not quiesce server administrators. When a QUIESCE SERVER command is issued, server administrators who are executing the OPERATE procedure remain connected, and the server continues to accept connections from them. Because the server terminates only after all users are disconnected, a server administrator can keep the server quiesced indefinitely by remaining connected to it. While a server is quiesced, an administrator can issue the START SERVER command to change the status of the server back to active.

For information about specifying a serverid, see "Specifying a Server" on page 131. For information about specifying the SAPW= option, see "Specifying a Server-Access Password" on page 131.

Setting the Current Server

You set the current server by using the SET SERVER command, which is referred to by a subsequent PROC OPERATE command. The syntax for the SET SERVER command is

SET SERVER;

SET SERVER *serverid* <(SAPW=*password*)>;

SET SERVER *serverid* </ SAPW=*password*>;

The server specified in the SET SERVER command overrides the server that is specified in a previous SET SERVER command. It also overrides the default server that is specified in the SERVERID= option in the PROC OPERATE statement.

If no server is specified, the SET SERVER command re-sets the current server to the default server that is specified in the SERVERID= option in the PROC OPERATE statement. However, if the SERVERID= option was not specified in the PROC OPERATE statement, SET SERVER retains the current server value.

The effect of a SET SERVER command is limited to the current execution of PROC OPERATE.

For information about specifying a serverid, see "Specifying a Server" on page 131. For information about specifying the SAPW= option, see "Specifying a Server-Access Password" on page 131.

In the following example, you use the SET SERVER command to establish the current server. After a server is set as current, you can display information about that current server without specifying the serverid.

```
proc operate;
   set server share1;
   display server;
```

In the following example, the OPERATE procedure first displays information about the two server libraries identified by LIB1 and LIB2, which are defined to the current server MYSHR. Next, the current server is re-set to SHARE, and information is displayed about library LIBALPHA, which is defined to the server SHARE.

```
proc operate serverid=share;
   set server myshr;
   display library lib1 lib2;
   set server;
   display library libalpha;
```

If you fail to identify the server before you issue a command that acts on the current server, PROC OPERATE issues the following message:

```
ERROR: PROC OPERATE is not currently set to any
       server, so this command will be ignored.  Use
       the 'SET SERVER serverid;' command to establish
       communication with a server.
```

Re-Starting a Quiesced Server

You re-start a server that has been quiesced by using the START SERVER command. The syntax for this command is

START SERVER;

START SERVER *serverid* <(SAPW=*password*)>;

START SERVER *serverid* </ SAPW=*password*>;

You may use the abbreviation ST in place of START.

This command re-starts a SAS server from a quiesced state only. If the *serverid* argument is not specified, this command starts the current server.

For information about specifying a serverid, see "Specifying a Server" on page 131. For information about specifying the SAPW= option, see "Specifying a Server-Access Password" on page 131.

You cannot use the START SERVER command to re-start a stopped server; instead, you must execute the SERVER procedure.

Stopping a Server

You terminate a server immediately by using the STOP SERVER command. The syntax for this command is

STOP SERVER;

STOP SERVER *serverid* <(SAPW=*password*)>;

STOP SERVER *serverid* </ SAPW=*password*>;

This command immediately terminates the server that is identified by the *serverid* argument or the current server if no serverid is specified.

If users are currently reading from or writing to members in the server library, the server closes the members. If users are in the process of updating data sets, updates may be lost. The server releases the libraries held by each user and disconnects each user.

For information about specifying a serverid, see "Specifying a Server" on page 131. For information about specifying the SAPW= option, see "Specifying a Server-Access Password" on page 131.

User Management Commands

Displaying Information about a User

You display information about one or more users by using the DISPLAY USER command. The syntax for this command is

DISPLAY USER *userid-1 <. . . userid-n>*;

DISPLAY USER _ALL_;

You may use the abbreviation DISP or D in place of DISPLAY.

This command displays information about one or more users who are accessing the current server.

If one or more userids are specified, the DISPLAY USER command displays summary information that is followed by detailed information for each user who has one or more libraries assigned.

If the _ALL_ argument is specified, summary information is displayed for other administrators and for all users who have connected and who have been explicitly stopped.

This section contains examples of display command output that contains host-specific *SAS-data-lib* values. See "Specifying a Host-Specific SAS Data Library" on page 125 for a list of SAS data library host-specific examples. For information about specifying a userid, see "Specifying a User" on page 131.

The following example contains two types of librefs: a user libref and a server libref. The *user libref* is the name by which the user refers to the library. It is provided only as an aid for communicating with the user, if necessary.

The *server libref* is the name that a server administrator assigns to the library by using one of these methods:

- □ a LIBNAME statement specified before the PROC SERVER statement
- □ a PROC OPERATE ALLOCATE LIBNAME command
- □ a host-dependent external allocation.

A library that is not defined by using these methods does not have a server libref. Therefore, admininistrative commands that subsequently refer to that library must use the host-specific library name instead of, for example, a UNIX pathname. However, a Version 6 server always reports a server-defined libref that is valid for references to libraries in subsequent commands.

```
DISPLAY USER 15;

         USER ID       STATUS      LIBRARIES
         ---------------------------------
         MIKE(15)      ACTIVE         2
```

User MIKE(15) is accessing these libraries:

```
USER LIBREF   SERVER LIBREF    LIBRARY NAME
-------------------------------------------------
USAGE         USAGE            SAS-data-lib
MEM                            SAS-data-lib
```

User MIKE(15) is accessing these data sets:

```
USER   SERVER    MEMBER     TYPE       OPEN MODE
LIBREF LIBREF
---------------------------------------------------
USAGE  USAGE     USAGE      CATALOG    UPDATE
USAGE  USAGE     MODULE     DATA       INPUT
MEM              MEMOBY     DATA       INPUT
```

In this example, MIKE(15) currently has files open in both of the libraries to which he currently has access. If user MIKE(15) had no files opened when the DISPLAY USER command was issued, only the first two parts of the output would be displayed.

The libname that a user refers to as MEM is not server-defined.

Quiescing a User's Access to a Server

You quiesce (gradually terminate) a user's access to a SAS server by using the QUIESCE USER command to deny new user requests for resources. This command moves the user from an active status to a stopped status. The syntax for the QUIESCE USER command is

QUIESCE USER *userid-1* <. . . *userid-n*>;

QUIESCE USER _ALL_;

You may use the abbreviation QUI or Q in place of QUIESCE.

This command quiesces one or more users who are connected to the current server. It quiesces users who currently have files open, that is, those who are currently reading from or writing to files at the time the command is issued. When a quiesced user closes all files in a server library, the server releases that user's access to the library. If the user has no open files in an accessed server library, the server releases that user's access to the library immediately. When the user has released all server libraries, the userid is assigned a stopped status and is disconnected from the server. While a user is quiesced, you can issue the START USER command to change the user's status back to active.

If you issue the QUIESCE USER command that specifies the userid for a user who is not currently connected to the server, the server assigns a stopped status to that userid. A stopped user cannot connect to the server until you issue the START USER command for that userid.

For information about specifying a userid, see "Specifying a User" on page 131.

The _ALL_ argument quiesces all users who are connected to the current server. When _ALL_ is specified, the QUIESCE USER command does not affect the administrator who issues the command. You cannot quiesce yourself even if you specify your userid in the QUIESCE USER command. However, be aware that you can quiesce other server administrators by name. When PROC OPERATE terminates and its server session is quiesced, that administrator is assigned a stopped status.

Re-Starting a Quiesced or a Stopped User

You re-start a user who has been stopped or quiesced by using the START USER command. The syntax for the START USER command is

START USER *userid-1* <. . . *userid-n*>;

START USER _ALL_;

You may use the abbreviation ST in place of START.

This command allows one or more SAS users to access the SAS server. Because users are allowed access to a SAS server by default, this command is necessary only to undo the effect of a previous STOP USER or a QUIESCE USER command.

If the _ALL_ argument is specified, the command re-starts all users who are quiesced or are stopped. For information about specifying a userid, see "Specifying a User" on page 131

When a stopped userid is re-started, that userid becomes unknown to the server.

Terminating a User's Connection to a Server

You immediately terminate a user's connection to a server by using the STOP USER command. The syntax for the STOP USER command is

STOP USER *userid-1* <. . . *userid-n*>;

STOP USER _ALL_;

This command immediately stops one or more users from using the server and disconnects their communication path to the server. If a user who has a stopped status issues a LIBNAME statement to re-access a server library, the request fails.

If you specify the userid of a user who is currently active or quiesced, the server closes library members that the user has open, terminates the user's access to libraries that are accessed through the server, and terminates the user's communication path to the server. If the user is updating a data set when the command is issued, updates may be lost.

For information about specifying a userid, see "Specifying a User" on page 131.

Because users are allowed access to a SAS/SHARE server by default, this command can be useful as a security tool. To stop users who are not currently using the server, specify their userids in the STOP USER command.

If the _ALL_ argument is specified, users who are currently active or quiesced are stopped; however, the administrator who issues the command is not affected. You cannot stop yourself even if you specify your userid in the STOP USER command. However, be aware that you can stop other server administrators.

Specifying a Server

You should specify an appropriate serverid for the server that you want to manage. Assuming that you started the server, you should already know its name.

A *serverid* specifies a one- or two-level name for the server. The server name must meet the criteria for a valid SAS name, although it can also include the following special characters: dollar sign ($), at sign (@) and pound sign (#).

Server naming is also constrained by the host type and the access method that you specify for communication between a server session and a user session. For complete server-naming details by host, see *Communications Access Methods for SAS/CONNECT and SAS/SHARE Software*.

Specifying a Server-Access Password

A server-access password (SAPW) is required if you are not already connected to the server, and the OAPW= option to the SERVER procedure is in effect. You can specify the server-access password in two ways: as a resource ID option (SAPW=*password*) or as a PROC OPERATE command option (/SAPW=*password*).

Specifying a User

A *userid* identifies a specific user or a particular connection to a server. A userid is specified as either a number, an identifying connection, or a name that identifies the user. A userid name must meet the criteria for a valid SAS name, although it can also include the following special characters: dollar sign ($), at sign (@), and pound sign (#). The host on which the client runs can impose particular usernaming constraints. For details, see *Communications Access Methods for SAS/CONNECT and SAS/SHARE Software*. Examples of userids are:

maria

3

Each time a user accesses a SAS/SHARE server, that new connection is assigned a number. A user is referred to by a combination of that number and the applicable userid in the server SAS log and in the OPERATE procedure output. A user ID is represented in the form: *userid*(*nnnn*).

A USER command in which you specify a user connection number is restricted to only that user's specific connection. For example, if Maria accesses the same server for a third time, she is identified by the server as MARIA(3). To display information about that single connection, you can issue the following command:

```
display user 3;
```

A USER command in which you specify a userid operates on all current connections of that user. Additionally, the QUIESCE, STOP, and START commands act on that user's future connections. For example, if Maria connects as described in the preceding example and also accesses the server a fifth time, the following command provides information about both MARIA(3) and MARIA(5):

```
display user maria;
```

The following command terminates the MARIA(3) and MARIA(5) connections to the server and prevents Maria from reconnecting to the server.

```
stop user maria;
```

The correct bibliographic citation for this manual is as follows: SAS Institute Inc., *SAS/SHARE User's Guide, Version 8*, Cary, NC: SAS Institute Inc., 1999. pp. 247.

SAS/SHARE User's Guide, Version 8

ISBN 1–58025–478–0

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.