



CHAPTER

27

SAS Component Language (SCL) Functions and Options

Introduction 239

Dictionary 239

Locating and Storing Sample Script Files with SCL 244

Introduction

The SAS Component Language (SCL) of SAS/AF and SAS/FSP provides four functions that can be used with SAS/CONNECT:

- COMAMID
- RLINK
- RSESSION
- RSTITLE.

In addition, two system options can be used by either user-written SCL applications or SAS/ASSIST software.

- SASSCRIPT
- SASFRSCR.

Dictionary

COMAMID SCL Function

Returns a string that contains all of the COMAMIDs that are valid for the operating system on which the SCL code executes.

Local and remote

Syntax

```
cval=COMAMID();
```

Syntax Description

cval

a string that contains all of the communications access methods that are valid for the specific operating system.

Details

The COMAMID function returns a string that contains all of the communications access methods (COMAMIDs) that are valid for the operating system on which the SCL code executes. Each value is separated by a blank. This function is useful for providing a list of COMAMIDs for users. The list is displayed as determined by the developer. The function merely returns a string of values.

Example

The following program fragment gets the string of COMAMIDs that are valid for the operating system on which this SCL program is executed. After that string is returned, one way to display the values would be in a listbox. Although this example does not include it, you would specify that the listbox be filled with the text string ***cval***.

```
comlist=makelist();
str=comamid();
do i=1 to 10;
  com=scan(str,i,' ');
  if com^=' ' then
    comlist=insertc(comlist,com,i);
end;
```

RLINK SCL Function

Verifies whether a link was established between a local and a remote SAS session.

Local and remote

Syntax

```
rc=RLINK(remote-session-id);
```

Syntax Description**rc**

is the return code.

remote-session-id

is the name of the remote session (specified by REMOTE=*remote-session-id*) that is being tested.

Details

The RLINK function verifies whether a link was established between the local SAS session and a remote SAS session.

Example

The following statements use the RLINK function and the identifier REMSESS.

```
rc=rlink('REMSESS');
if (rc=0) then
  _msg_='No link exists.';
else
  _msg_='A link exists.';
```

RSESSION SCL Function

Returns the name, description, and SAS System version of a remote session.

Local and remote

Syntax

cval=RSESSION(*n*);

Syntax Description

cval

is the character string that contains the following information:

characters 1 through 8

the session identifier (REMOTE=*remote-session-id*)

characters 9 through 40

the description

n

is the number of the remote session to get session information for. If no remote link exists, the returned value is blank. If a link exists but no description was specified, characters 9 through 40 in the returned value are blanks.

Details

The RSESSION function returns the session identifier and corresponding description for a remote session. You must have previously defined the description by using the RSTITLE function.

Example

This example loops through four sessions and gets the remote session and description, which is returned by using the RSESSION function. The program puts the descriptions in separate arrays for later use (for example, to display a choice of remote sessions to upload to).

```
do i=1 to 4;
  word=rsession(i);
```

```

    if word ^= ' ' then do;
        remote=substr(word,1,8);
        desc=(substr(word,9,20));
        if rlink(remote) then do;
            if desc=' ' then desc = remote;
            cnt=cnt + 1;
            entrys{cnt}=remote;
            comam{cnt}=desc;
        end;
    end;
end;
end;

```

RSTITLE SCL Function

Defines a description for an existing connection to a remote session.

Local and remote

Syntax

sysrc=**RSTITLE**(*session-id*, *description*);

Syntax Description

sysrc

is zero if the description was saved or non-zero if the operation failed.

session-id

is the name of the remote session (specified by **CONNECTREMOTE=remote-session-id**). The string can be up to eight characters long.

description

is a description to associate with the remote session. The string can be up to 40 characters long.

Details

The RSTITLE function saves the session identifier and description for an existing connection to a remote session. This information can be retrieved by using the RSESSION function to build a list of connections. The list can then be used to select a connection when submitting statements to a remote host.

Example

The following statements define the description **OS/390 Payroll Data** for the remote session by using the identifier **A**:

```

session='A';
descrip='OS/390 Payroll Data';

```

```
rc=rstitle(session,descrip);
```

SASSCRIPT Option

Specifies one or more storage locations of SAS/CONNECT script files

Syntax

SASSCRIPT=(*'location-1' 'location-2' ... 'location-n'*)

Syntax Description

'location'

specifies the name of an aggregate storage location, such as PDS for OS/390, a maclib for CMS, or a directory for directory-based systems.

Details

The system option SASSCRIPT= uses the value of *location* to generate an internal fileref that is stored as the value of the SASFRSCR system option.

To specify one or more aggregate storage locations, enclose each location in quotation marks, use blanks to separate multiple locations, and place the entire specification in parenthesis.

After you specify the location of the script files with SASSCRIPT, you can do the following:

- In an SCL program, you can use the fileref that is stored as the value of SASFRSCR to build a list of available scripts for establishing a link with SAS/CONNECT software.
- In SAS/ASSIST software, when you use the REMOTE CONNECT item to edit a remote configuration, enter a question mark (?) in the SCRIPT field of the CREATE NEW CONFIGURATION submenu to display a list of script file names.
- You can specify the script file with the CSCRIPT= option, and if the script file is not found in the current directory, then it will be looked for in the SASSCRIPT directory.

SASFRSCR Option

Contains the fileref that is generated by the SASSCRIPT system option

Syntax

SASFRSCR

Details

The option SASFRSCR contains a fileref that is automatically generated when the SASSCRIPT= system option changes. SASFRSRC cannot be updated by the user, it is a read-only option. SASFRSCR is commonly used in SCL applications to prompt a user for the correct SAS/CONNECT sign-on script.

Locating and Storing Sample Script Files with SCL

The system option SASSCRIPT= defines the location of the SAS/CONNECT script files. The value of the SASSCRIPT= SAS system option is a logical name or one or more aggregate storage locations (such as, directories or partitioned data sets). When you set the SASSCRIPT= system option, it generates another SAS system option, SASFRSCR, which is set to the value of a fileref that is used to build a list of scripts for SCL applications. When you establish a link while using SAS/ASSIST, this product uses the information provided by the SASFRSCR option to provide a list of available scripts. You can also build a similar menu of script files for user-written applications by accessing the SASFRSCR system option from an SCL program.

The following SCL program obtains the value of the SASFRSCR system option and uses it to create a list of scripts. Refer to *SAS Component Language: Reference* for information about the SCL functions used in this example.

```

INIT;
return;

MAIN:
    /* Get internally-assigned fileref.          */
    fileref=optgetc('sasfrscr');

    /* Open the directory (aggregate storage */
    /* location).                            */
    dirid=dopen(fileref);

    /* Get the number of files.              */
    numfiles=dnum(dirid);

    /* Define a custom selection list the */
    /* length of the number of files and */
    /* allowing users to make one choice. */
    call setrow(numfiles,1);
return;

TERM:
    /* Close the directory.                  */
    rc=dclose(dirid);
return;

GETROW:
    /* Display the list of file names.      */
    filename=dread(dirid,_currow_);
return;

```

```
PUTROW:
    /* Get directory path name.          */
    fullname=pathname(fileref);

    /* Concatenate filename user selects */
    /* with directory path name.        */
    name=fullname || '/' || filename;
    /* Other SCL statements to use complete */
    /* filename stored in name.          */
return;
```


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.