



THE
POWER
TO KNOW.

SAS/ACCESS[®] 4.3

Interface to R/3

User's Guide

The correct bibliographic citation for this manual is as follows: SAS Institute Inc. 2009. *SAS/ACCESS® 4.3 Interface to R/3: User's Guide*. Cary, NC: SAS Institute Inc.

SAS/ACCESS® 4.3 Interface to R/3: User's Guide

Copyright © 2009, SAS Institute Inc., Cary, NC, USA

ISBN 978-1-59994-527-9

All rights reserved. Produced in the United States of America.

For a hard-copy book: 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.

For a Web download or e-book: Your use of this publication shall be governed by the terms established by the vendor at the time you acquire this publication.

U.S. Government Restricted Rights Notice. Use, duplication, or disclosure of this software and related documentation by the U.S. government is subject to the Agreement with SAS Institute and the restrictions 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 electronic book, February 2009

1st printing, March 2009

SAS® Publishing provides a complete selection of books and electronic products to help customers use SAS software to its fullest potential. For more information about our e-books, e-learning products, CDs, and hard-copy books, visit the SAS Publishing Web site at support.sas.com/publishing or call 1-800-727-3228.

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.

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

Contents

<i>What's New</i>	v
Overview	v
Authorization Checks for Business Information Warehouse Objects	v
Enhanced CALLRFC Procedure	vi
New AUTHDOMAIN= Option	vi
Secure Network Communication for Single Sign-on	vi
SAS RFC Server Changes	vi
Chapter 1 \triangle Getting Started with SAS/ACCESS Interface to R/3	1
Getting Started with SAS/ACCESS Interface to R/3	1
Searching R/3 Metadata	4
Exporting R/3 Metadata to SAS Warehouse Administrator Window and Field Help	7
Accessibility Features of the SAS/ACCESS Interface to R/3	8
Chapter 2 \triangle Accessing and Using R/3 Data	9
Accessing Data Using Direct Access and ODBC	9
Accessing Data Using the SAS/ACCESS Interface to R/3	9
Accessing Data Using Secure Network Communications	12
Authorization Checks for BW and BI Objects	13
Chapter 3 \triangle Using the SAS RFC Server	15
Overview of the SAS RFC Server	15
Using the SAS RFC Server on UNIX	15
Using the SAS RFC Server on Windows	17
Using the SAS RFC Server in Batch Mode	19
Chapter 4 \triangle Using the RFC Macros and Macro Variables	21
Using the RFC Macros and Macro Variables	21
Using the R/3 BAPI Connector: Logon Window	32
Performing Batch Operations	33
Chapter 5 \triangle Using the LIBNAME Statement for SAP	37
Overview of the LIBNAME Engine for SAP	37
LIBNAME Statement Syntax for SAP	37
SAP LIBNAME Engine Options	38
SAS LIBNAME Statement Options for the SAP Engine	50
SAS Data Set Options for the SAP Engine	54
Examples	58
Chapter 6 \triangle The CALLRFC Procedure	61
Overview: CALLRFC Procedure	61
Syntax: CALLRFC Procedure	61

Examples: CALLRFC Procedure 70

Appendix 1 \triangle Recommended Reading 73

Recommended Reading 73

Glossary 75

Index 77

What's New

Overview

The SAS/ACCESS Interface to R/3 provides a read-only engine for using SAS to access data in your SAP R/3 or SAP BW system. SAS/ACCESS Interface to R/3 has the following changes and enhancements:

- authorization checks for Business Information Warehouse Objects
- CALLRFC procedure DESCRIBE FUNCTION enhancement
- SAP LIBNAME engine AUTHDOMAIN= option
- secure network communication for single sign-on
- SAS RFC server changes

Note: The changes discussed here are from SAS/ACCESS 4.3 Interface to R/3 and later. Δ

Authorization Checks for Business Information Warehouse Objects

SAS provides a set of Business Add-Ins (BAI) that the SAP LIBNAME engine uses to perform authorization checks for BW and BI objects. The authorization checks are enabled or disabled by activating or deactivating the BAIs. The authorization check implementation depends on the SAP release:

- For SAP NetWeaver BI 7.0 and later, authorization checks are performed using the analysis authorization method. This provides an authorization check for the following:
 - the infoProvider (infoCube, infoObject, DSO)
 - column-level restrictions on master data attributes and key figures
 - row-level restrictions on attributes
- For SAP Systems older than SAP NetWeaver BI 7.0, the SAP LIBNAME engine supports checking user authorization with the reporting authorization method. User authorization is checked at the InfoCubes, InfoObject, and ODS levels.

Enhanced CALLRFC Procedure

The CALLRFC procedure DESCRIBE FUNCTION statement is enhanced to retrieve information about the expected SAS data type, length, and format of the data sets and parameters.

New AUTHDOMAIN= Option

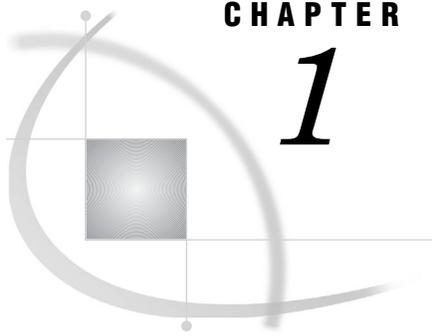
This new option enables retrieving user identity credentials from a SAS Metadata Server, when SAS/ACCESS Interface to R/3 is used in a metadata environment.

Secure Network Communication for Single Sign-on

Secure network communication (SNC) enables secure communication between the SAP LIBNAME engine and the SAP System. SNC can be configured to provide single sign-on.

SAS RFC Server Changes

A SAS RFC Server was needed in previous releases to enable communication between the SAP LIBNAME engine and the SAP System. The SAS RFC Server required administration to start and stop, as well as configuration. In SAS/ACCESS 4.3 Interface to R/3, the SAP LIBNAME engine communicates directly with the SAP System, removing the need for a SAS RFC Server. However, a SAS RFC Server is needed for z/OS.



CHAPTER

1**Getting Started with SAS/
ACCESS Interface to R/3**

<i>Getting Started with SAS/ACCESS Interface to R/3</i>	1
<i>Searching R/3 Metadata</i>	4
<i>Exporting R/3 Metadata to SAS Warehouse Administrator</i>	7
<i>Window and Field Help</i>	8
<i>Accessibility Features of the SAS/ACCESS Interface to R/3</i>	8

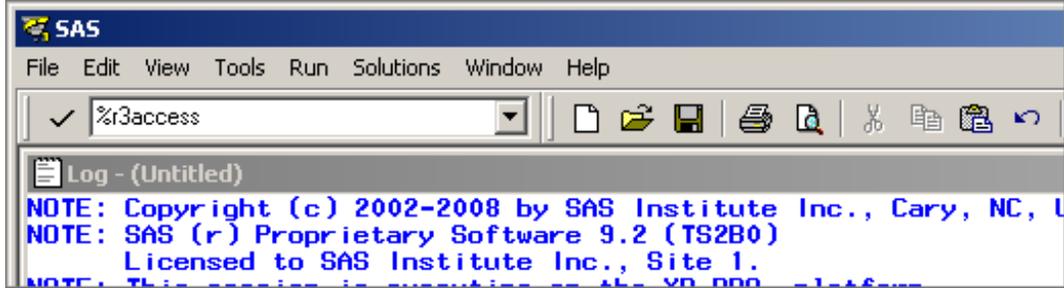
Getting Started with SAS/ACCESS Interface to R/3

This section provides a brief tutorial on how to use the SAS/ACCESS Interface to R/3. For more detailed information, consult the additional references that are listed in “Recommended Reading” on page 73 or contact your on-site SAS support personnel.

To get started with the SAS/ACCESS Interface to R/3:

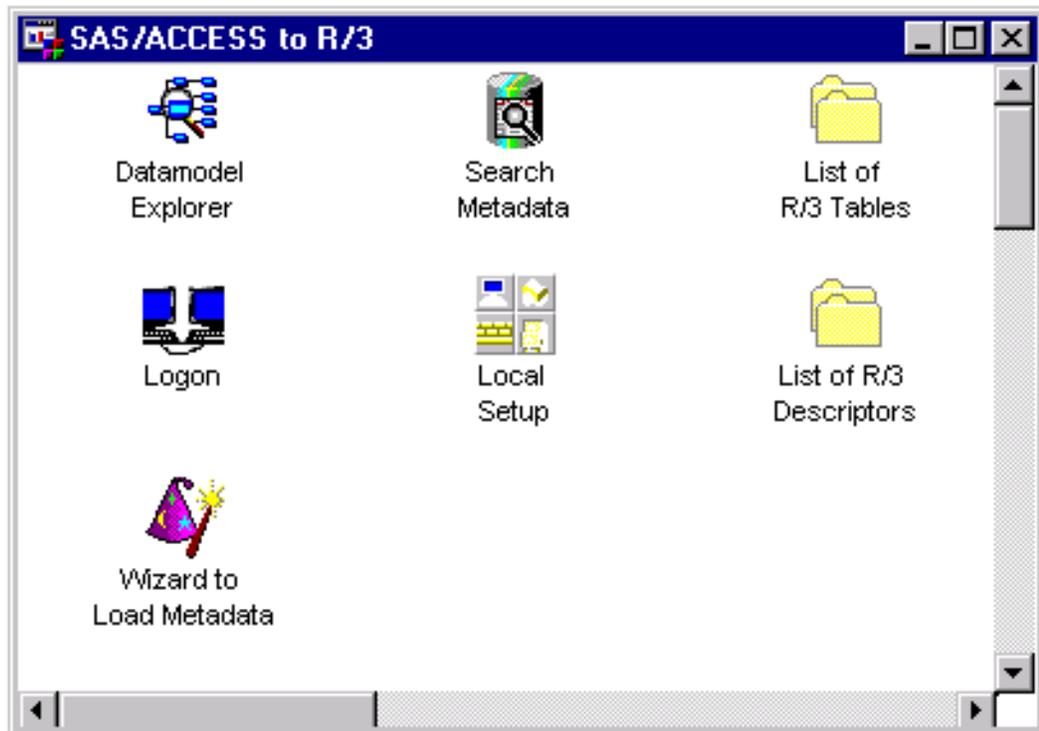
- 1 Open a SAS session.
- 2 Enter `%r3access` in the command line to start the SAS/ACCESS Interface to R/3:

Display 1.1 SAS Command Line



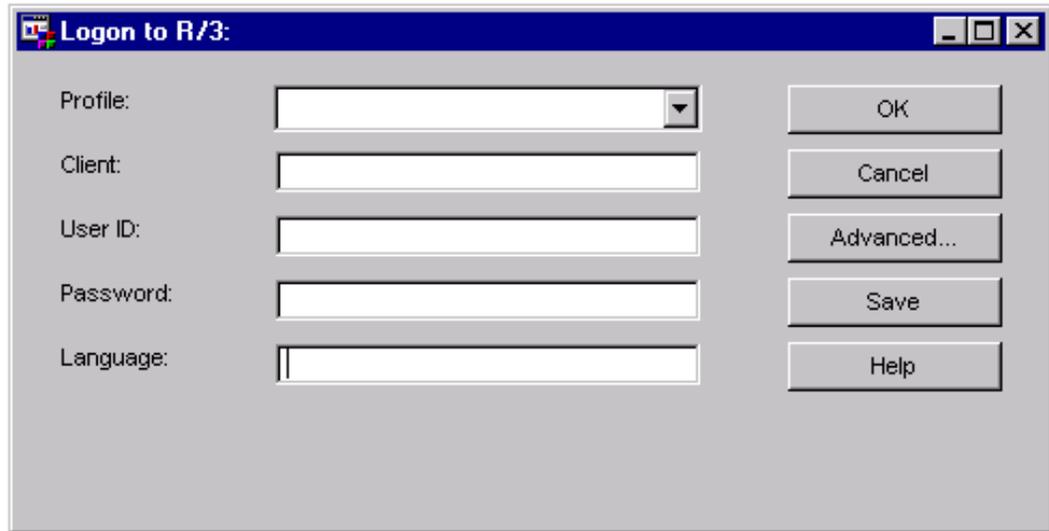
The SAS/ACCESS to R/3 desktop appears:

Display 1.2 SAS/ACCESS to R/3 Desktop



3 Double-click the Logon icon to open the Logon to R/3 window.

Display 1.3 Logon to R/3 Window



The screenshot shows a dialog box titled "Logon to R/3:". The dialog has a blue title bar with standard window controls (minimize, maximize, close). The main area is light gray and contains five input fields on the left and five buttons on the right. The input fields are labeled "Profile:", "Client:", "User ID:", "Password:", and "Language:". The "Profile:" field is a drop-down menu. The buttons are "OK", "Cancel", "Advanced...", "Save", and "Help".

Existing predefined profiles are available from the **Profile** field drop-down list.

To use a predefined profile:

- 1 Select a profile from the **Profile** field drop-down list.
- 2 Click **OK** to connect to the SAP System.

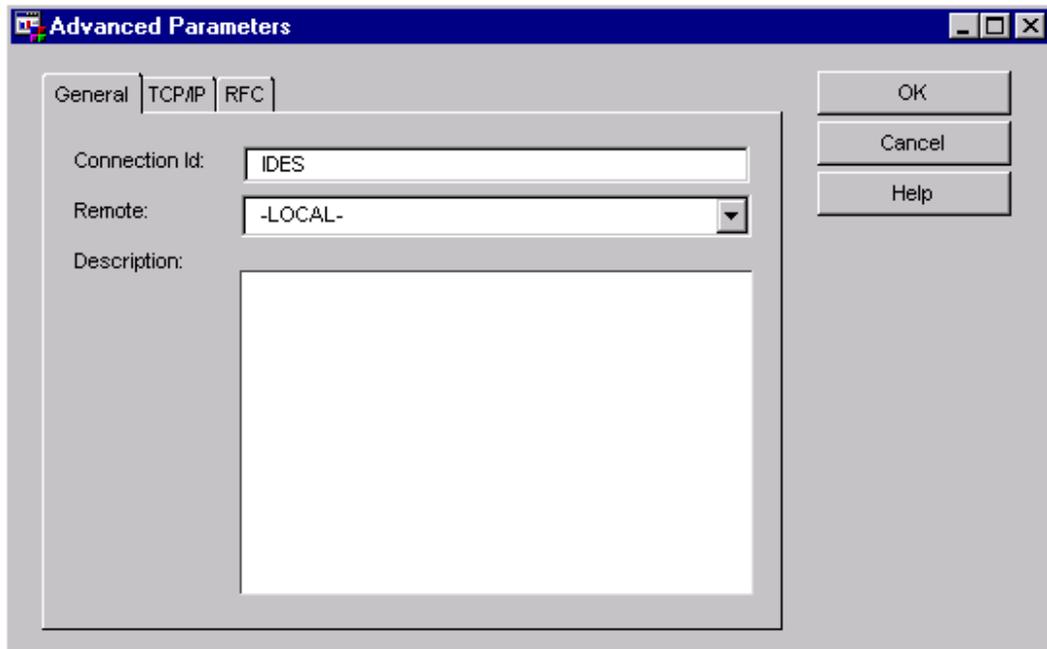
If there are no predefined profiles, you will need to create one.

To create a new profile:

- 1 Open the Logon to R/3 window as described earlier.
- 2 If you want to save the connection information for later use, enter a name for the **Profile**.
- 3 Enter a three-digit identification number for the **Client**.
- 4 Enter a valid **User ID**.
- 5 Enter the corresponding **Password** for the user ID.

- 6 Click **Advanced** in the Logon to R/3 window to display the Advanced Parameters window:

Display 1.4 Advanced Parameters Window



- 7 For z/OS operating environments, perform the following steps:
 - a Select the TCP/IP tab and enter the SAS RFC server name in the **Host** field. The default is **localhost**; a fully qualified domain name might not be required, depending on the DNS configuration for the network.
 - b Enter an appropriate port number in the **Port** field.
- 8 From the RFC tab, enter the SAP System host name in the **Host** field.
- 9 Click **OK** to close the Advanced Parameters window and return to the Logon to R/3 window.
- 10 If you want to use this logon information later, click **Save** to save the profile information.
- 11 Click **OK** to connect to the SAP System.

Searching R/3 Metadata

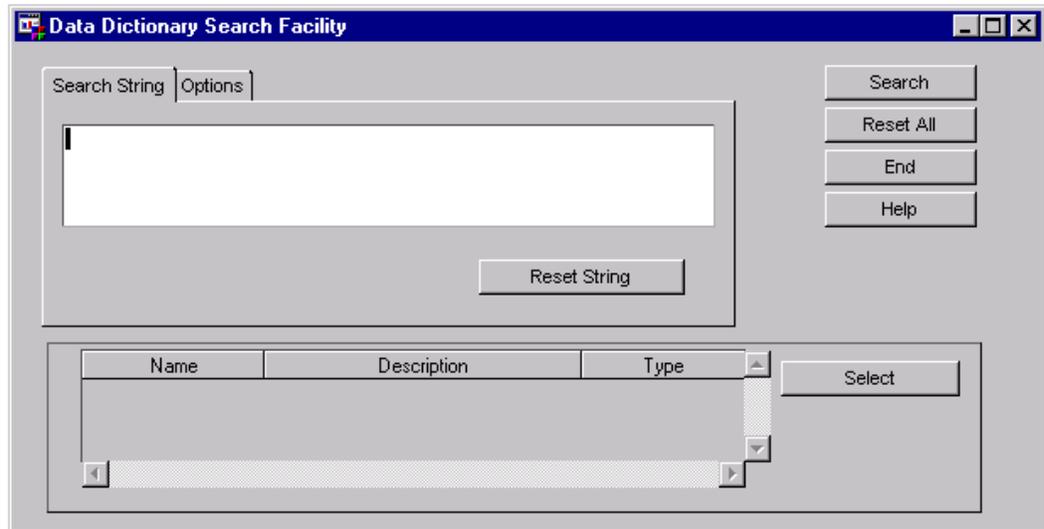
The SAS/ACCESS Interface to R/3 provides a robust search capability that you can use to search your metadata for specific information.

For example, to search for all tables that contain the word “scrap” in their metadata:

- 1 From SAS, start the SAS/ACCESS Interface to R/3 to display the SAS/ACCESS to R/3 window.

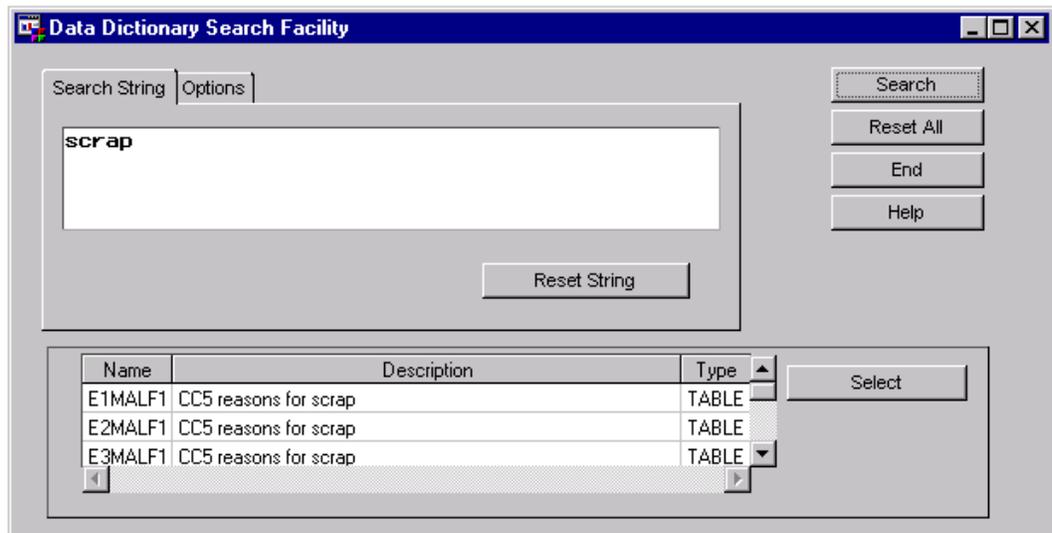
- 2 Double-click the Search Metadata icon to display the Data Dictionary Search Facility window:

Display 1.5 Data Dictionary Search Facility Window



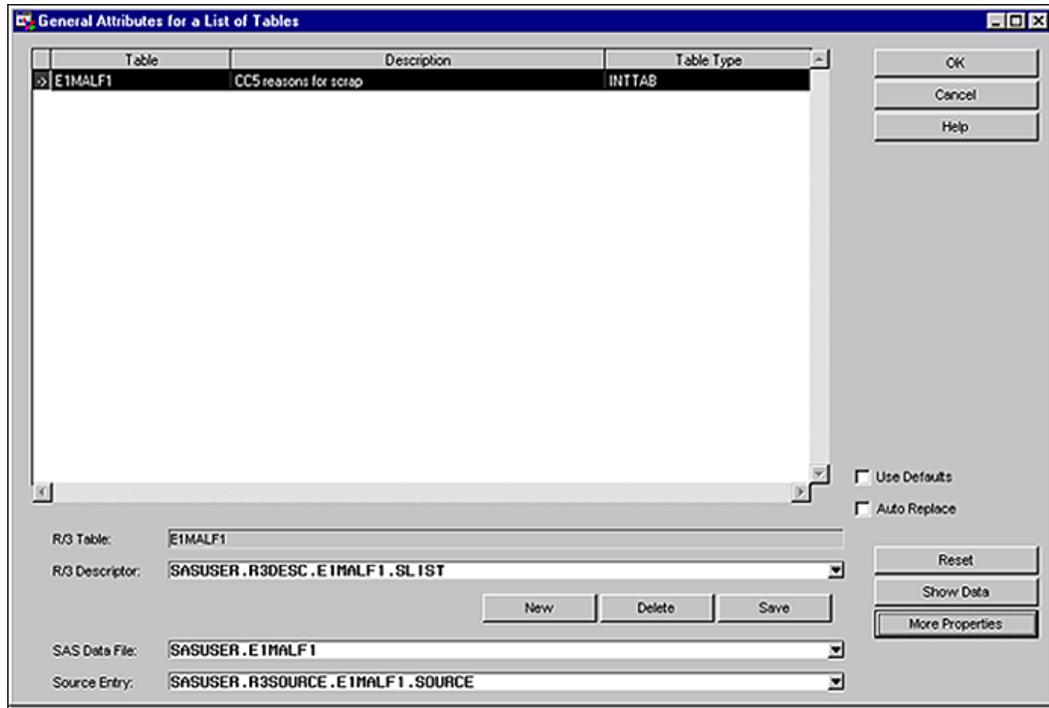
- 3 Click inside the text entry field on the Search String tab, and then enter text to search for, such as the word “scrap”. Click **Search**. A list of tables will display in the table in the bottom of the Search String tab as shown in the following figure:

Display 1.6 Data Dictionary Search Facility Window (Searching for 'scrap')



- 4 Double-click a table in the list. The General Attributes for a List of Tables window displays to give you more detailed information about the selected table as shown in the following figure:

Display 1.7 General Attributes for a List of Tables Window



- 5 Click **OK** to return to the Data Dictionary Search Facility window.

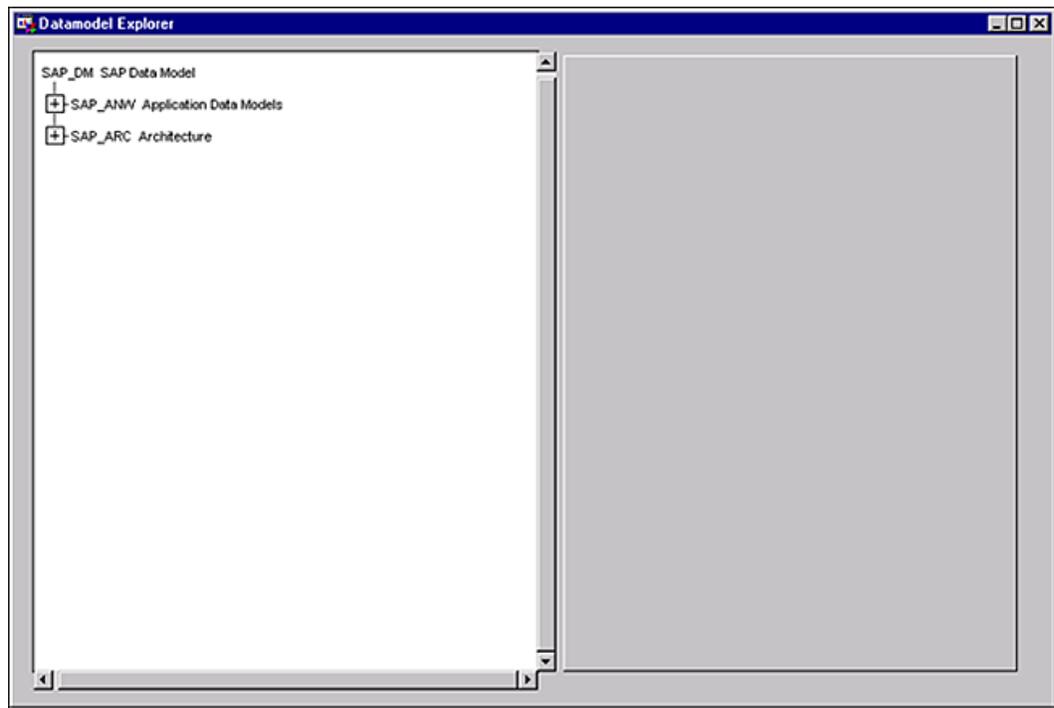
Exporting R/3 Metadata to SAS Warehouse Administrator

The SAS/ACCESS Interface to R/3 enables you to export metadata from SAP tables to a warehouse environment in SAS Warehouse Administrator.

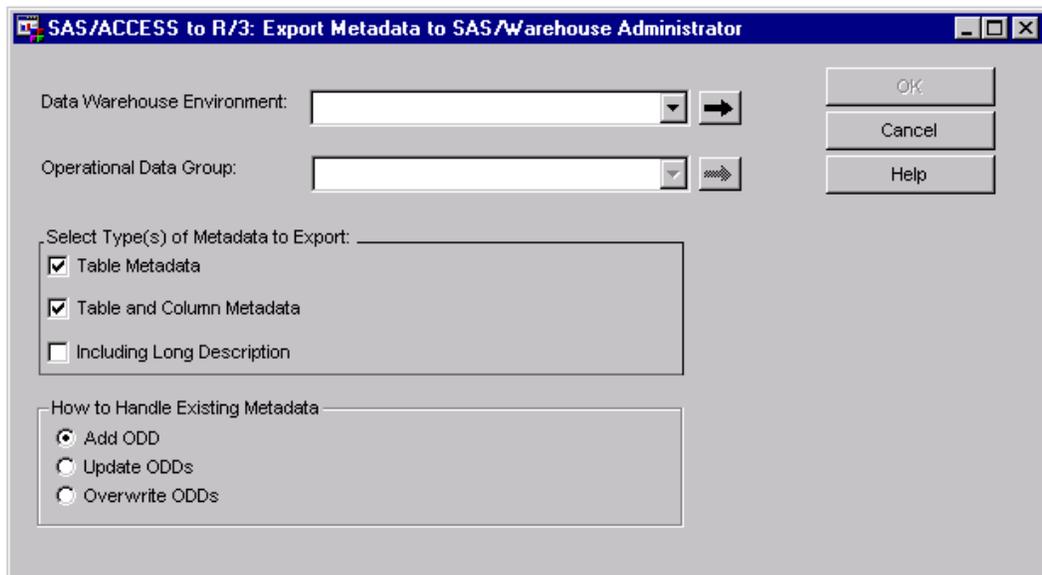
To export metadata:

- 1 From SAS, open the SAS/ACCESS Interface to R/3 to display the SAS/ACCESS to R/3 desktop.
- 2 Double-click the Datamodel Explorer icon to display the Datamodel Explorer window.

Display 1.8 Datamodel Explorer Window



- 3 For this example, click the plus sign (+) for the application data models topic in the Datamodel Explorer. Then, click the plus sign (+) beside the Industry Solutions and Industry Solutions Public Sector subtopics. Click ISPS_HR IS-PS Human Resources to select this subtopic.
- 4 From the main menu of the Datamodel Explorer window, select **File ► Export Metadata ► SAS Warehouse Administrator** to display the SAS/ACCESS to R/3: Export Metadata to SAS Warehouse Administrator window.

Display 1.9 SAS/ACCESS to R/3: Export Metadata to SAS Warehouse Administrator Window

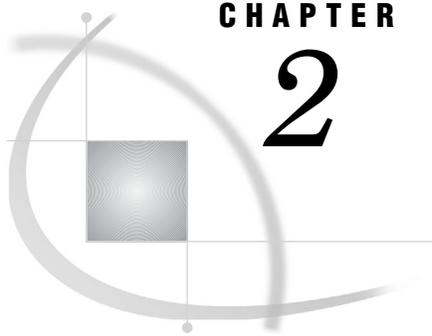
- 5 From the **Data Warehouse Environment** drop-down list, select a pre-defined environment from the SAS/ACCESS to R/3 folder.
- 6 From the **Operational Data Group** drop-down list, select an operational data group from the selected data warehouse environment. The **Operational Data Group** field is disabled until you select a data warehouse environment.
- 7 Click **OK** to export the metadata to the selected data warehouse environment in SAS Warehouse Administrator.

Window and Field Help

For descriptions of windows and buttons in the SAS/ACCESS to R/3 desktop, click **Help** on any window to access *SAS/ACCESS Interface to R/3: Help*.

Accessibility Features of the SAS/ACCESS Interface to R/3

The SAS/ACCESS Interface to R/3 includes accessibility and compatibility features that improve usability of the product for users with disabilities. These features are related to accessibility standards for electronic information technology adopted by the U.S. Government under Section 508 of the U.S. Rehabilitation Act of 1973, as amended.



CHAPTER

2

Accessing and Using R/3 Data

<i>Accessing Data Using Direct Access and ODBC</i>	9
<i>Accessing Data Using the SAS/ACCESS Interface to R/3</i>	9
<i>Remote Function Call (RFC) Method</i>	10
<i>Extract Management Method</i>	11
<i>Installation and Setup</i>	11
<i>Data Transfer and Use</i>	11
<i>Accessing Data Using Secure Network Communications</i>	12
<i>Using the saprfc.ini File</i>	12
<i>Using the RFC_STRING option</i>	12
<i>Authorization Checks for BW and BI Objects</i>	13

Accessing Data Using Direct Access and ODBC

There are two ways to access data that do not require the SAS/ACCESS Interface to R/3: direct access and ODBC.

Direct Access

This is a form of transparent access that does not require additional programming or intermediate operations. This form of access is possible because the SAP System is an Open Systems-compliant application.

Direct access uses an intermediate SAS/ACCESS product to the specific database.

ODBC (Open Database Connectivity)

This method allows direct data access in a PC environment by using the SAS/ACCESS Interface to the ODBC product.

Accessing Data Using the SAS/ACCESS Interface to R/3

You can use the SAS/ACCESS Interface to R/3 to access SAP data by using two different methods:

Remote Function Call

The Remote Function Call method is normally used. It offers online access to the SAP data.

Extract Management

The extract management method is the original method that has now largely been superseded. It offers off-line access to the SAP data that can be useful when

- there is a poor or slow communication link between SAS and SAP

- a backup file is required
- there is a lot of data

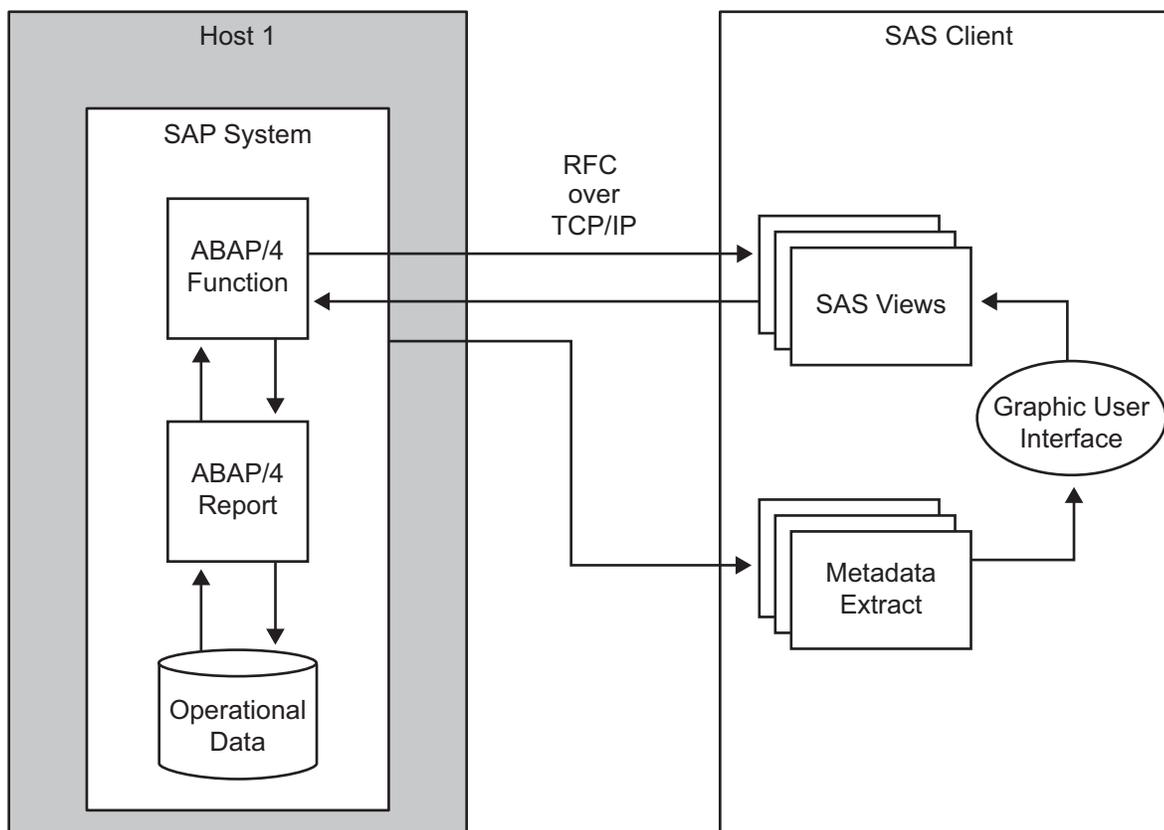
Note: Both methods generate a SAS DATA step. Δ

Remote Function Call (RFC) Method

The Remote Function Call (RFC) is an SAP implementation of the Remote Procedure Call. RFC allows access to R/3 from external applications, such as SAS or other SAP Systems.

An overview of the Remote Function Call method for accessing data from SAP is shown in the following figure. Two modules (ABAP Function and ABAP Report, which are provided with the interface) attach to the R/3 system. Communication between the ABAP Function module and the interface itself uses RFC over a TCP/IP connection. Note that in a z/OS environment, in addition to the components in the previous figure, a SAS RFC Server is deployed between the SAP System and the SAS Server to provide access to the data in the SAP System.

Figure 2.1 Remote Function Call (RFC) Method of Accessing Data from the R/3 system



In general, the RFC method consists of generating a SAS DATA step to access the SAP data, and then it uses the DATA step to create either

- a data set that is a SAS copy of the SAP data
- a View that reads the SAP data each time it is required.

Extract Management Method

The extract management method creates an ABAP query that copies the data from SAP to an external file. It also creates a SAS DATA step that then reads the external file into SAS. After the SAP data has been exported to SAS, the data can be held either temporarily for discrete analysis or permanently in a SAS data warehouse. The interface generates all of the necessary programs for both ABAP and SAS (based on the SAP data dictionary tables).

The extract management method to access SAP data is discussed in the following sections.

Installation and Setup

- 1 Generate an ABAP report.

Use the descriptor file to create an ABAP report and write it to a text file. The ABAP report is used later to copy the SAP data to an external intermediate data file.

- 2 Generate a DATA step.

Generate a SAS DATA step to read the intermediate data file.

- 3 Install the ABAP report in R/3.

This step is independent of the SAS/ACCESS Interface to R/3 and SAS and can be performed simultaneously with Step 2.

- 4 Install the ABAP text file from Step 1 into SAP.

Data Transfer and Use

Note: After installation and setup, you can perform the following steps as many times as required. Δ

- 1 Run the ABAP report in R/3.

Run the installed ABAP report on SAP. This creates the external intermediate data file.

- 2 Use the DATA step.

Use the DATA step to read in or copy the external intermediate data file into SAS.

Note: Depending on your situation, it might be possible to simplify or streamline the outlined process. For example, in UNIX a named pipe or other external file can enable ABAP to write data to the external intermediate data file so that SAS can read it simultaneously. Δ

Accessing Data Using Secure Network Communications

The SAP System and the RFC protocol support secure network communications (SNC). SNC can be configured to enable single sign-on (SSO). SNC can also be used without single sign-on; in this case, user name and password credentials are still required. This section shows two examples of how to specify the SNC logon information with the SAP LIBNAME engine. The SAP System and the user accounts must be configured for SNC. For information about how to configure the SAP System and SNC, see the SAP document *Secure Network Communications User's Guide*.

Using the `saprfc.ini` File

This method requires creating an entry in the `saprfc.ini` file similar to the following example:

```
DEST=SI9
TYPE=A
ASHOST=sapr3srv.sup.com
SYSNR=03
SNC_MODE=1
SNC_PARTNERNAME=p:SAPServiceSI9@sup.com
SNC_LIB=C:\Public\win32sso\gssntlm.dll
```

Then, use the LIBNAME statement to use the connection:

```
libname a r3 dest=SI9;
```

Using the RFC_STRING option

Another alternative is to use the RFC_STRING option to apply additional settings:

```
libname a r3 ahost=sapr3srv.sup.com sysnr=03
  rfc_string="SNC_MODE=1 SNC_PARTNERNAME=p:SAPServiceSI9
  SNC_LIB=C:\Public\win32sso\gssntlm.dll";
```

Note: The SNC_LIB string must not contain blanks. \triangle

This document shows two examples of how to use SNC for single sign-on in a LIBNAME statement. For more information about SNC and SNC configuration, see SAP documentation and other third-party documentation.

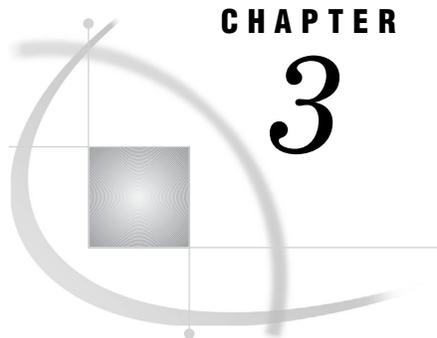
Authorization Checks for BW and BI Objects

The SAP System release level determines how authorization checks are performed for BW and BI objects. In releases before SAP BI 7.0, SAP uses the reporting authorization that uses the SAP standard authorization concept. For BI 7.0 and later, SAP uses an authorization concept for analysis authorization. SAS/ACCESS Interface to R/3 delivers one Business Add-In (BAi) that implements the reporting authorization, and one BAi that implements analysis authorization.

During installation, SAS is configured to use one of three implementations for table access authorization checks. The default implementation when no BAi is activated uses the SAP authorization object `S_TABU_DIS`. If you want to use one of the two BAi implementations you have to activate it during installation. A description of the implementations follows:

Table 2.1 Authorization Check Implementations

Authorization Check Implementation	Description
(Default, no BAi)	Authorization object <code>S_TABU_DIS</code>
Classic BAi / <code>SAS/AUTHBW01</code>	This BAi is compatible with BW and BI systems. User authorization checks are performed at the InfoCube, InfoObject, and ODS levels using the reporting authorization (SAP standard authorization concept).
Enhanced BAi / <code>SAS/IM_AUTHBI01</code>	This BAi is compatible with BI 7.0 and later systems. User authorization checks are performed using the analysis authorization. This not only provides an authorization check for the infoProvider (infoCube, infoObject, and DSO) but also column-level restrictions on master data attributes and key figures, and row-level restrictions on attributes.



CHAPTER

3

Using the SAS RFC Server

<i>Overview of the SAS RFC Server</i>	15
<i>Using the SAS RFC Server on UNIX</i>	15
<i>Using the SAS RFC Server on Windows</i>	17
<i>Using the SAS RFC Server in Batch Mode</i>	19

Overview of the SAS RFC Server

The SAS RFC server is part of the SAS/ACCESS Interface to R/3 in a z/OS operating environment. The SAS RFC server is a component of the SAP LIBNAME engine that provides access to the data in SAP Systems. The SAS RFC Server also supports calling ABAP RFC functions. The SAS RFC Server uses the TCP/IP protocol for communication with the SAP LIBNAME engine. For more information about the SAP LIBNAME engine, see “Overview of the LIBNAME Engine for SAP” on page 37.

The SAS RFC server:

- runs independently of SAS as a background process on
 - the SAS host
 - the R/3 host
 - another machine entirely
 - can be configured to listen on any TCP/IP port (default **6999**).
 - can have more than one concurrent instance on a given machine (but each instance must have a unique port number). Each instance can communicate with many SAP Systems.
 - can process requests from multiple SAP application servers.
 - can support Unicode and non-Unicode SAP Systems.
-

Using the SAS RFC Server on UNIX

In a UNIX environment, the SAS RFC server for SAP is installed in the **!SASROOT/saspgm/dbi/bin** directory. You must add this directory to the **PATH** environment variable and add the location of the shared libraries to the appropriate environment variable. For more detailed information about how to set up the environment variables, see *Installation Instructions— SAS/ACCESS Interface to R/3*.

To start the SAS RFC server in a UNIX environment, use the following command

```
sasrfc_server [-d 0|1|2|3|4|5|6] [-n name] [-p port] [-w workdir] [-N namespace]
              [-V variant] [-help] [-c dialog_compat_port]
```

```
[-b "-p port-number -V variant_name -R report_name"]
[-B number_of_bytes]
```

Optional parameters are enclosed in brackets [].

To stop the SAS RFC server in a UNIX environment, use the following command:

```
sasrfc_server [-k] [-p port]
```

Optional parameters are enclosed in brackets [].

The options for the SAS RFC server command for UNIX are as follows:

- d *n* specifies whether to run the server as a daemon process or as an application and specifies the diagnostic level.
 The server is run as a daemon process when you specify *-dn* where *n=0* or when you do not specify the *-dn* option. After successful initialization, all messages are routed to **syslog**.
 When you specify the *-dn* option where *n=* any number between **1** and **6**, the program runs as an application and messages are routed to standard error. Higher diagnostic levels route more detailed messages to standard error.
- k stops the service.
- n *name* specifies the name of the service. This option is used to distinguish multiple daemon processes. For example, a test service might be started as *-n test*.
- p *port* identifies the port number on which the service listens. The default port number is **6999**.
- w *workdir* specifies the working directory. For example, if your working directory is named **tmp**, specify the option as follows: **-w /tmp**.
- N *namespace* specifies the namespace for ABAP functions and programs that are used by SAS. The default value is **/SAS/**. If the ABAP programs are installed in the customer namespace rather than in the **/SAS/** namespace, then this parameter identifies where the ABAP programs are installed.
- V *variant* specifies the report variant that is assigned to this server. You must assign a report variant for each SAS RFC Server that accesses an SAP application server in batch mode. The variant parameter **G_DEST** specifies a unique destination to call back to the SAS RFC Server.
- help specifies that you want to display command line Help.
- c identifies the option that, when specified, runs the SAS RFC Server in V8 compatibility mode on the port indicated by the *dialog_compat_port* parameter. The port number must be different than the one used for the V9 requests. If the *-c* option is omitted, no dialog compatibility will be available.
- b enables batch compatibility so that you can process V8 ACCR3 requests while you are using a V9 SAS RFC server. You must use the double quotation marks in the batch compatibility option in order to separate the batch compatibility parameters from the V9 parameters that use the same syntax.
- "-p *port_number*"
- V
- variant_name*
- R *report_name*"

The port parameter is optional. If omitted, the port number defaults to **6999**.

The report variant parameter is required and must be a valid variant of the report.

The report parameter is optional and would normally be omitted. If the report parameter is omitted, then *report_name* defaults to / SAS/ZSASV8COMPAT.

The values in *port_number*, *variant_name*, and *report_name* must be different from the values used by V9 and different from the values used by dialog compatibility.

-B sets the minimum buffer size for data transfers in batch and dialog modes. The *number_of_bytes* should be greater than 10000 and no more than 8 digits, or it will be ignored. If the **-B** option is omitted, the default is 10000 bytes.

In batch mode, if you specify a value for the G_BUFMAX parameter that is less than the value of **-B**, then the **-B** value is used.

For example, to start the SAS RFC server as a daemon process with the name of **test** on port number **6991**, use the following command:

```
sasrfc_server -n test -p 6991
```

The following example command shows how to display command line Help:

```
sasrfc_server
```

To start the SAS RFC server with debug level **4** on port number **6991**, use the following command:

```
sasrfc_server -d 4 -p 6991
```

To stop a SAS RFC server that is running on port number **6991**, use the following command:

```
sasrfc_server -k -p 6991
```

Using the SAS RFC Server on Windows

In a Windows environment, the SAS RFC server for SAP is installed in the **!SASROOT\access\sasexe** directory.

To start or stop the SAS RFC server using a command line in the Windows operating environment, use the following command:

```
sasrfc_server [-i n] [-r] [-s] [-k] [-t n] [-d n] [-n name[,desc]] [-p port]
              [-N namespace] [-V variant] [-help] [-c dialog_compat_port]
              [-b "-p port-number -V variant_name -R report_name"]
              [-B number_of_bytes]
```

The options for the SAS RFC server command for Windows are as follows:

-i n installs the program as a Windows service and specifies the startup option for a new service. Startup options are

	<i>n</i> =2	starts the service automatically
	<i>n</i> =3	starts the service on demand
	<i>n</i> =4	disables the service
-r		removes the program from the Service Manager.
-s		starts the service.
-k		stops the service
-t <i>n</i>		sets the startup option for an existing service. Startup options are
	<i>n</i> =2	starts the service automatically
	<i>n</i> =3	starts the service on demand
	<i>n</i> =4	disables the service.
-d <i>n</i>		specifies the diagnostic level. When you specify the -d <i>n</i> option where <i>n</i> = any number between 1 and 6, the program runs as an application and messages are routed to standard error. Higher diagnostic levels route more detailed messages to standard error.
-n <i>name,desc</i>		specifies the name and description of the service.
-p <i>port</i>		specifies the port number on which the server listens. The default port number is 6999.
-N <i>namespace</i>		specifies the namespace for ABAP functions and programs that are used by SAS. The default value is /SAS/. If the ABAP programs are installed in the customer namespace rather than in the /SAS/ namespace, then this parameter identifies where the ABAP programs are installed.
-V <i>variant</i>		specifies the R/3 report variant that is assigned to this server. You must assign a report variant for each SAS RFC Server that accesses an R/3 application server in batch mode. The variant parameter G_DEST specifies a unique destination to call back to the SAS RFC Server.
-help		specifies that you want to display command line Help.
-c <i>dialog_compat</i> <i>_port</i>		identifies the option that, when specified, runs the SAS RFC Server in V8 compatibility mode on the port indicated by the <i>dialog_compat_port</i> parameter. The port number for a SAS 9.1 request must be different than the one used for the SAS 9 requests. If the -c option is omitted, no dialog compatibility will be available.
-b “-p <i>port_number</i> -V <i>variant_name</i> -R <i>report_name</i> ”		enables batch compatibility so that you can process V8 ACCR3 requests while you are using a V9 SAS RFC Server. You must use the double quotation marks in the batch compatibility option in order to separate the batch compatibility parameters from the V9 parameters that use the same syntax.

The port parameter is optional. If omitted, the port number defaults to 6999.

The report variant parameter is required and must be a valid variant of the report.

The report parameter is optional and would normally be omitted. If the report parameter is omitted, then *report_name* defaults to /SAS/ZSASV8COMPAT.

The values in *port_number*, *variant_name*, and *report_name* must be different from the values used by V9 and different from the values used by dialog compatibility.

-B *number_of_bytes* sets the minimum buffer size for data transfers in batch and dialog modes. The *number_of_bytes* should be greater than 10000 and no more than 8 digits, or it will be ignored. If the **-B** option is omitted, the default is 10000 bytes.

In batch mode, if you specify a value for the `G_BUFMAX` parameter that is less than the value of **-B**, then the **-B** value is used.

For example, if you want to start the SAS RFC Server with a debug level of 1 on port number 6991, use the following command:

```
sasrfc_server -d 1 -p 6991
```

To install the SAS RFC Server as a service with automatic startup on port 6991 and use the name **test** and a description of '**Test RFC server**', use the following command:

```
sasrfc_server -i 2 -n 'test,Test RFC server' -p 6991
```

To uninstall the server that was started in the previous example, use the following command:

```
sasrfc_server -r -n 'test'
```

Using the SAS RFC Server in Batch Mode

The SAS RFC Server can perform batch mode processing of RFC calls to the SAP system. Users who submit interactive RFC calls to the SAP system might encounter time-out limitations. This occurs when interactive SAP jobs exceed the default processing time-out limit of 10 minutes.

By allowing batch mode processing of RFC calls to the SAP system, time-out limitations for interactive SAP dialog processes are resolved. Batch mode processing is often used to reduce the use of dialog processes and to reduce the impact of extractions (a SAS program reading the tables and views of the SAP system by means of the SAP LIBNAME engine) on the SAP system. Batch processes usually run at a lower priority. Batch mode processes of RFC calls can also help control the number of simultaneous extractions.

The ABAP program that is normally generated (by another ABAP program), executes a requested read and join operation on the database table(s). When operating in batch mode, the SAS RFC Server creates and releases an SAP job to execute this ABAP program. The SAS RFC Server then waits for the job that is running the ABAP program to start and to recall the SAS RFC Server. Finally, the resulting data stream (that the SAS RFC Server receives from the SAP system) is passed on to the SAP LIBNAME engine.

LIBNAME or data set option: BATCH | BATCH_MODE | BATCHMODE= 0|1|Y|N

Indicates whether the SAS RFC Server should use SAP batch jobs for the data extracts.

Y RFC uses batch jobs to extract R/3 data

N RFC uses dialog processes to extract R/3 data

The default value for this option is N.

To submit batch requests to the SAS RFC Server, an RFC destination and a variant for ABAP program /SAS/Z_SAS_READ must be set up for each SAS RFC Server instance. Before the SAS RFC Server can execute requests as a batch job the following steps must be performed:

- 1 *Create a SAS RFC Server destination.* For each SAS RFC Server that accesses an SAP System, a destination must be set up on that SAP System. The program ID and the gateway information defined in the destination are used to register the SAS RFC Server on an SAP gateway. If two SAS RFC Servers are used to read data from one SAP application server, two RFC destinations must be set up on that SAP application server.

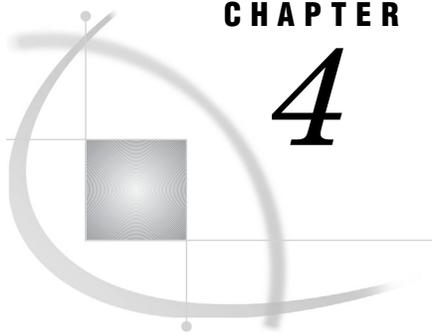
If gateway information is omitted from the SM59 destination, the batch job will be constrained to run on the R/3 server mentioned in the SAS LIBNAME statement for the request.

If explicit gateway information is entered in the SM59 destination information, the batch job will be run on any eligible R/3 server as determined by the SAP site batch configuration.

Omitting the gateway information is typically used in small SAP configurations such as test, QA, and single server systems. In production environments, it is preferable to have the jobs run on a server allocated by the SAP batch system.

- 2 *Create a variant of ABAP program /SAS/Z_SAS_READ.* After defining the destination for the SAS RFC Server, you will need to make the RFC destination known to the ABAP program. This is done by creating a variant for the ABAP program /SAS/Z_SAS_READ. The concept of variants provides a way of parameterizing ABAP programs. For each defined destination, a variant needs to be created that references the destination. Variants are client dependent.
- 3 *Start the SAS RFC Server with the variant.* For more information about using the SAS RFC Server and particularly the sasrfc_server option, see “Using the SAS RFC Server on UNIX” on page 15 and “Using the SAS RFC Server on Windows” on page 17.

For further information about the setup of SAS RFC Server destinations and variants, see the *Installation Instructions— SAS/ACCESS Interface to R3*.



CHAPTER

4

Using the RFC Macros and Macro Variables

<i>Using the RFC Macros and Macro Variables</i>	21
<i>Using the R/3 BAPI Connector: Logon Window</i>	32
<i>Performing Batch Operations</i>	33
<i>Source Parameter</i>	34
<i>Passwords</i>	34

Using the RFC Macros and Macro Variables

The SAS/ACCESS Interface to R/3 includes several SAS macros and SAS macro variables that enable you to communicate with the SAS RFC Server. The following SAS macros and SAS macro variables are available:

%CALLRFC

enables you to call a specified function. For more information about using the %CALLRFC macro, see “%CALLRFC” on page 22.

%R3CONN

enables you to use a profile to log on to the SAP System for batch processing. For more information about using the %R3CONN macro, see “%R3CONN” on page 26.

%R3CONNC

enables you to log on to the SAP System when performing batch processing. For more information about using the %R3CONNC macro, see “%R3CONNC” on page 27.

%R3CONNE

enables you to log off of the SAP System when performing batch processing. For more information about using the %R3CONNE macro, see “%R3CONNE” on page 29.

RFC_LOGON_INFO

enables you to specify the logon information and connection parameters that are used in the %CALLRFC macro. For more information about using the RFC_LOGON_INFO macro variable, see “RFC_LOGON_INFO” on page 30.

%CALLRFC

Calls a specified RFC-enabled function module

Type: autocall macro

See also:

“Using the R/3 BAPI Connector: Logon Window” on page 32

“RFC_LOGON_INFO” on page 30

Syntax

```
%CALLRFC(<RFC-enabled function module name> EXPORTING <parameter string>
          IMPORTING <parameter string> INTABLES <parameter string> OUTTABLES
          <parameter string> USING <parameter string>)
```

Details

%CALLRFC is a SAS macro that allows you to call any RFC-enabled function module that you specify. Function modules that are called using this macro must

- be RFC-enabled
- be synchronous
- have no user interaction

The parameters for %CALLRFC are as follows:

EXPORTING	<p>passes field values or structures to the specified function module. The EXPORTING parameters are declared as import parameters in the function interface and are defined as follows:</p> <ul style="list-style-type: none"> □ <i>p1=value1 ... pn=valuen</i> These parameters are passed as field values to the specified function module. □ <i>p1.n1=value p1.n2=value2 ... pn.vn=valuen</i> These parameters are passed as elements of a structured parameter (a complex parameter that contains fields).
IMPORTING	<p>passes field values or structures from the specified function module back to SAS. The IMPORTING parameters are defined as follows:</p> <ul style="list-style-type: none"> □ <i>p1=var1 ... pn=varn</i> The <i>p1 ... pn</i> parameters are export parameters declared in the function interface. The <i>var1 ... varn</i> parameters are SAS macro variables that are used to pass the values from the SAP System back to SAS. The macro variables must be defined before the function call is submitted. □ <i>p1=dataset1 pn=datasetn</i> Parameters that are written into a data set require a two-level data set name. Writing parameters into data sets can be useful for structured parameters.

- INTABLES** passes references to input SAS data sets. The INTABLES parameters are defined as follows:
- \square *p1=dataset1 ... pn=datasetn*
- The SAS data sets are converted into internal tables and are passed to the specified function module.
- Note:* The variable names in the SAS data set must match the field names of the internal table as they are defined in the function interface. Δ
- OUTTABLES** specifies references to output SAS data sets. The OUTTABLES parameters are defined as follows:
- \square *p1=dataset1 ... pn=datasetn*
- Internal tables passed back to SAS from the specified function module are converted into SAS data sets.
- USING** specifies alternate connection parameters when you log on to the SAP System. By default, the %CALLRFC macro uses the connection parameters that are defined in the RFC_LOGON_INFO variable. However, if you want to use an alternate set of connection parameters, create a variable that contains those parameters and reference that variable in the USING parameter.
- Note:* The alternate logon variable must be defined before the function call is submitted. Δ

Note: If an error occurs while the %CALLRFC macro is executing, the macro variables &SYSRC and/or &SYSERR contain a nonzero value. (See “Example 4” on page 25.) The ABAP function module might also contain parameters to return error conditions to the calling program. These return codes can be written into a SAS macro variable and checked in the SAS program. (See “Example 5” on page 25.) Δ

Using

The following macro variables are used with the %CALLRFC macro:

- \square RFC_LOGON_INFO

defines the R/3 BAPI connection parameters that are required to access the SAP System. You can specify these connection parameters by using the RFC_LOGON_INFO macro string or by using the R/3 BAPI Connector window, which is automatically displayed if you do not use the RFC_LOGON_INFO macro string when you submit your %CALLRFC macro statement. The following connection parameters are required:

- \square CLIENT
- \square USER
- \square PASSWD (or PASSWDX)
- \square LANG

Note: If any of the required values are missing, the user is prompted to provide the missing information. Δ

The RFC_LOGON_INFO macro variable also requires some connection information. The connection parameters should be defined as follows:

- \square If you use the saprfc.ini, specify the DEST value.

- If you use a specific application server, specify the ASHOST and SYSNR values.
- If you use Load Balancing, specify the MSHOST, R3NAME, and GROUP values.

Note: Load Balancing is available only if the CALLRFC executable has been linked using Release 4.0 or higher of the RFCSDK. △

For more information about defining the RFC_LOGON_INFO variable, see “RFC_LOGON_INFO” on page 30. For more information about using the R/3 BAPI Connector window, see “Using the R/3 BAPI Connector: Logon Window” on page 32.

Example 1

The following example shows how to use the %CALLRFC macro to read data from an SAP R/3 system into a SAS data set:

```
%callrfc(RFC_SYSTEM_INFO
         IMPORTING RFCSI_EXPORT=WORK.RFCSI_EXPORT);
```

In this example, the %CALLRFC macro calls the RFC-enabled function module named RFC_SYSTEM_INFO and writes the system information output to the SAS data set WORK.RFCSI_EXPORT.

Example 2

The following example shows how to specify the logon parameters for the %CALLRFC macro using the RFC_LOGON_INFO:

```
%let RFC_LOGON_INFO CLIENT=010 USER=TestUser
                    PASSWD=TestPwd LANG=E
                    ASHOST=HostName SYSNR=02;
%callrfc(RFC_SYSTEM_INFO
         IMPORTING RFCSI_EXPORT=WORK.RFCSI_EXPORT);
```

In this example, TestUser is connecting to a specific application server to read data from the R/3 system into a temporary SAS data set named WORK.RFCSI_EXPORT.

Example 3

The following example demonstrates how to use SAS dates in the function call:

```
%let MRETURN=;
%callrfc(BAPI_COSTCENTER_GETLIST
         EXPORTING CONTROLLINGAREA=1000
         DATE=%sysfunc(date (), yymmddn8.)
         IMPORTING RETURN=MRETURN
         OUTTABLES COSTCENTER_LIST=WORK.COSTCENTER_LIST);
```

In this example, the %CALLRFC macro is used to call the function module named BAPI_COSTCENTER_GETLIST. The function call reads the list of cost centers for controlling area 1000 for the current date and writes the output list into a temporary SAS data set named WORK.COSTCENTER_LIST. The return code of the function call is written to the SAS macro variable MRETURN. Note that the macro variable is defined before the %CALLRFC macro is used.

Example 4

The following example illustrates how to use the &SYSRC and &SYSERR macro variables to check for errors that occur while the %CALLRFC macro is executing:

```
%macro example4;
%callrfc(INVALID_FUNCTION_CALL);
    %if %eval(&sysrc) ne 0 or %eval(&syserr) ne 0 %then %do;
        %put An error occurred while calling the function.
        %put sysrc=&sysrc syserr=&syserr;
    %end;
%mend;
%example4;
```

In this example, a new macro named %EXAMPLE4 is created. %EXAMPLE4 uses the SAS macro variables &SYSRC and &SYSERR to check return codes from the %CALLRFC macro. The new macro also generates an error message if errors occurred.

Example 5

The following example shows how to use the parameters in the function module to check for errors that occurred during the execution of the %CALLRFC macro:

```
%macro example5;

%global bapi_return;

    /* create the input data set */
data WORK.IDRANGE;
sign='I';
option='BT';
low='0000000000';
high='9999999999';
output;
run;

%callrfc(BAPI_CUSTOMER_GETLIST
IMPORTING RETURN=BAPI_RETURN
INTABLES IDRANGE=WORK.IDRANGE
OUTTABLES ADDRESSDATA=WORK.ADDRESSES);
%if %substr(&bapi_return,1,1)=E or
    %substr(&bapi_return,1,1)=W %then %do;
    %put An error occurred while calling the BAPI_CUSTOMER_GETLIST function.;
    %put bapi_return=&bapi_return;
%end;

%else %do;
    proc print data=WORK.ADDRESSES;
        run;
    %end;
%mend;

%example5;
```

In this example, a new macro named %EXAMPLE5 is created to retrieve a list of customers from the R/3 system. The range of customer numbers to be retrieved is specified in the SAS data set WORK.IDRANGE. This data set is then used as the input

table in the function call. The information for customers between customer number 0000000000 and 9999999999 will be read into the SAS data set WORK.ADDRESSES. The function parameters contain return codes and messages from the BAPI_CUSTOMER_GETLIST function. The structure of the return parameter is defined in the function interface. In this example, the first character in the return string contains the message type. E indicates an error message and W indicates a warning message.

%R3CONN B

Connects to or disconnects from an SAP System using a profile

Type: batch macro

Syntax

```
%R3CONN B(PROFILE=value, LIBREF=value, FUNCTION=open | close, ID=value,
SOURCE=value)
```

Details

The %R3CONN B macro allows you to log on to and log off of the SAP System using parameters stored in the data set *libref.R3CONN* under the profile name. Profiles that connect to and disconnect from the SAP System can be used for batch operation, such as overnight processing, and to simplify the logon process for end users.

The parameters for %R3CONN B are as follows:

PROFILE	identifies the name of the profile that contains the information used in the Logon to R/3 window.
LIBREF	identifies the SAS library containing the LIBREF.R3CONN data set. This data set contains the logon parameters defined for the profile.
FUNCTION	specifies whether you want to open or close the connection to the R/3 system. Valid values are <ul style="list-style-type: none"> <input type="checkbox"/> OPEN <input type="checkbox"/> CLOSE
ID	specifies the connection ID that is associated with the profile. <p><i>Note:</i> This parameter was defined for previous versions of this macro and is supported in this release for backward compatibility. However, it is recommended that this parameter not be used. Δ</p>
SOURCE	specifies the logon parameters that are defined in SOURCE type catalog entries. Although the SOURCE parameter is supported in this version, logon parameters should now be defined in profiles that are stored in a SAS data set. <p><i>Note:</i> This parameter was defined for previous versions of this macro and is supported in this release for backward compatibility. However, it is recommended that this parameter not be used. Δ</p>

Example 1

The following example shows how to use the %R3CONN macro to connect to the R/3 system using a profile:

```
%r3connb(PROFILE=BatchUser, LIBREF=batch, FUNCTION=open);
```

In this example, the profile BatchUser is used to connect to the R/3 system. The BatchUser profile uses the connection parameters stored in the BATCH.R3CONN data set.

Example 2

The following example shows how to use the %R3CONN macro to disconnect from the R/3 system by using a profile:

```
%r3connb(PROFILE=BatchUser, LIBREF=batch, FUNCTION=close);
```

The BatchUser profile was used to connect to the R/3 system. This example demonstrates how to use the %R3CONN macro to close the BatchUser profile's connection to the R/3 system.

%R3CONN

Connects to an SAP System

Type: batch macro

See also: “%R3CONN” on page 29

Syntax

```
%R3CONN(CCONN=value, HOST=value, PORT=value, USR=value, PWD=value,
        CLI=value, LNG=value, HST=value, DST=value, SNA=value, R3=value,
        SYS=value, GWS=value, GWH=value, FUNC=value, CPICUSR=value,
        CPICPWD=value, CPICCLI=value, CPICLNG=value, CPICDST=value,
        CPICFRM=value, DEBUG=value, REMSESS=value)
```

Details

The %R3CONN macro allows you to connect to an SAP System. This macro can be used to log on to the SAP System during batch operation.

The parameters for %R3CONN are as follows:

CCONN	specifies the connection identifier to be used when connecting to the SAP System.
HOST	specifies the name of the TCP/IP host to be used when connecting to the SAS RFC Server. This parameter is used in z/OS operating environments only.
PORT	specifies the TCP/IP port to be used when connecting to the SAS RFC Server. This parameter is used in z/OS operating environments only.
USR	specifies the user ID to be used when connecting to the SAP System.

PWD	specifies the encoded password to be used when connecting to the SAP System.
CLI	specifies the three digit client ID to be used with the SAP System.
LNG	specifies the logon language to be used with the SAP System.
HST	specifies the SAP server host name to be used when connecting to the SAP System.
DST	specifies the destination name to be used when using the SIDEINFO file to specify connection parameters.
SNA	specifies whether SAS connects to an R/2 system. Valid values are <ul style="list-style-type: none"> <input type="checkbox"/> Y—SAS connects to an R/2 system. <input type="checkbox"/> blank—SAS is <i>not</i> connecting to an R/2 system.
R3	specifies whether SAS is connecting to an R/3 system. Valid values are <ul style="list-style-type: none"> <input type="checkbox"/> Y—SAS is connecting to an R/3 system. <input type="checkbox"/> blank—SAS is <i>not</i> connecting to an R/3 system.
SYS	specifies the SA P System number. This parameter can be specified here or in the SIDEINFO file.
GWS	specifies the TCP service of the SAP gateway. This parameter can be specified here or in the SIDEINFO file.
GWH	specifies the name of the host on which the SAP gateway is running. This parameter can be specified here or in the SIDEINFO file.
FUNC	specifies the function module to be used.
CPICUSR	specifies the user identifier for CPIC.
CPICPWD	specifies the encoded password for CPIC.
CPICCLI	specifies the client for CPIC.
CPICLNG	specifies the logon language for CPIC.
CPICDST	specifies the destination for CPIC communication (as defined in the TXCOM table in the SAP System).
CPICFRM	specifies the form for CPIC.
DEBUG	specifies whether you want to use the trace option. Valid values are <ul style="list-style-type: none"> <input type="checkbox"/> Y—Use the trace option. <input type="checkbox"/> blank—Do <i>not</i> use the trace option.
REMSESS	specifies the SAS/CONNECT remote session ID to be used when connecting to the R/3 system.

Note: The SIDEINFO file provides the connection parameters required for SAS to communicate with the SAP System. The SIDEINFO file is not needed if you specify all connection parameters in the %RCONNC macro. Δ

Example 1

The following example shows how to use the %R3CONNC macro to connect to an R/3 system when you specify a specific application server:

```
%r3conn(CCONN=IDES, USR=USER1, PWD=D27A927AD9E768,
        CLI=800, LNG=EN, HST=HostName1, SYS=02,
        CPICUSR=CPICUSER, CPICPWD=6FD3E4BAC4);
```

In this example, USER1 is connecting to an R/3 system on an application server named HostName1 with a system number of 02.

Example 2

The following example shows how to use the %R3CONN macro to connect to an R/3 system and an SAS RFC Server that runs on a specified application server:

```
%r3conn(CCONN=IDES, USR=USER1, PWD=D27A927AD9E768, CLI=800, LNG=EN,
        HOST=HostName2, PORT=6998,
        HST=HostName1, SYS=02,
        CPICUSR=CPICUSER, CPICPWD=6FD3E4BAC4);
```

In this example, SAS connects to an R/3 application server (HostName1) with the system number of 02 using an SAS RFC Server running on a separate host (HOSTNAME2) on port 6998.

Example 3

The following example shows how to use the %R3CONN macro and a SIDEINFO file to connect to an R/3 system identified by a logical destination:

```
%r3conn(CCONN=IDES, USR=USER1, PWD=D27A927AD9E768,
        CLI=800, LNG=EN, DST=SYSTEM1,
        CPICUSR=CPICUSER, CPICPWD=6FD3E4BAC4);
```

In this example, USER1 is connecting to an R/3 system. The R/3 system is identified by a logical destination (SYSTEM1), and the connection information is defined in the SIDEINFO file.

%R3CONN

Disconnects from an SAP System

Type: batch macro

See also: “%R3CONN” on page 27

Syntax

```
%R3CONN(CCONN=value, REMSESS=value)
```

Details

The %R3CONN macro allows you to disconnect from the SAP System. This macro can be used to log out of the SAP System during batch operation.

The parameters for %R3CONN are as follows:

CCONN	specifies the connection identifier that is associated with the current session.
-------	--

REMSESS specifies the SAS/CONNECT remote session identifier that is associated with the current remote session.

Example

The following example shows how to use the %R3CONNE macro to disconnect from an R/3 system:

```
%r3conne(CCONN=IDES);
```

In this example, the %R3CONNE macro is used to terminate the R/3 connection with an ID of IDES.

RFC_LOGON_INFO

Defines the connection parameters that are required to access the SAP System

Type: optional macro variable

Applies to: %CALLRFC macro

See also:

“%CALLRFC” on page 22

“Using the R/3 BAPI Connector: Logon Window” on page 32

Syntax

```
%let RFC_LOGON_INFO=ID=value ID=value ... ID=value
```

Note: Values are case-sensitive. IDs are not. △

Details

RFC_LOGON_INFO is a macro variable that is used with the %CALLRFC macro. The RFC_LOGON_INFO variable uses the following parameters:

Parameter ID	Description
TYPE	identifies the type of SAP System. Valid values are 3—(default) identifies an R/3 server E—identifies an external server.
CLIENT	is a required value that identifies the logon client.
USER	is a required value that specifies the user logon ID.
PASSWD	is a required value that identifies the logon password. This value is not required if a PASSWDX value is specified.
LANG	is a required value that identifies the logon language. Specify the one of the following: 1-byte SAP language (E for English, D for German, and so on) 2-byte ISO language (EN for English, DE for German, and so on)

Parameter ID	Description
LCHECK	specifies the logon check option at OPEN time. Valid values are 0—logon without check 1—(default) logon with check.
TRACE	specifies the RFC trace option. Valid values are 0—(default) without trace 1—with trace.
DEST	identifies the logical destination in the saprfc.ini file, if applicable. This value is required when you use the saprfc.ini file.
GWHOST	specifies the host name of the SAP gateway.
GWSERV	identifies the service of the SAP gateway.
MSHOST	specifies the host name of the message server if you use Load Balancing. This value is required if you use Load Balancing.
R3NAME	specifies the name of the SAP System if you use Load Balancing. This value is required if you use Load Balancing.
GROUP	specifies the name of the group of application servers if you use Load Balancing. This value is required if you use Load Balancing.
ASHOST	identifies the host name of the specific application server to be used. This value is required if you use a specific application server.
SYSNR	identifies the SAP System number if you use a specific application server and do not use Load Balancing. This value is required if you use a specific application server.
ABAP_DEBUG	specifies the ABAP debugger option. Valid values are 0—(default) run without the ABAP debugger 1—run with the ABAP debugger. <i>Note:</i> If you use the ABAP_DEBUG option, the SAP GUI must be installed. Δ
PASSWDX	identifies the SAP logon password that is encrypted by SAS.

Example 1

The following example shows how to specify logon information for the `%CALLRFC` macro using the `RFC_LOGON_INFO` macro variable:

```
%let RFC_LOGON_INFO CLIENT=010 USER=USER1
                    PASSWD=USERPWD LANG=E
                    MSHOST=HostName R3NAME=BIN GROUP=Public;
%callrfc(RFC_SYSTEM_INFO
        IMPORTING RFCSI_EXPORT=WORK.RFCSI_EXPORT);
```

In this example, `USER1` is connecting to a server that uses Load Balancing to read R/3 data into a temporary data set called `WORK.RFCSI_EXPORT`.

Example 2

The following example shows how to specify logon information for the `%CALLRFC` macro using the `RFC_LOGON_INFO` macro variable:

```
%let RFC_LOGON_INFO=CLIENT=010 USER=USER1
                        PASSWD=USERPWD LANG=E
                        ASHOST=HostName SYSNR=02;
%callrfc(RFC_SYSTEM_INFO
         IMPORTING RFCSI_EXPORT=WORK.RFCSI_EXPORT);
```

In this example, `USER1` is connecting to a specific application server to read R/3 data into a temporary data set called `WORK.RFCSI_EXPORT`.

Using the R/3 BAPI Connector: Logon Window

The R/3 BAPI Connector: Logon window is automatically displayed if you do not use the `RFC_LOGON_INFO` macro string when you submit your `%CALLRFC` macro statement. The R/3 BAPI Connector: Logon window enables you to define the R/3 BAPI connection parameters that are required to access the SAP System.

Display 4.1 R/3 BAPI Connector: Logon Window

The screenshot shows a dialog box titled "R/3 BAPI Connector: Logon". It contains the following fields and buttons:

- Client:
- User:
- Password:
- Language:
- Destination:
- Host:
- System number:
- Gateway host:
- Gateway service:
- Buttons: OK, Cancel, Help

For more information about the `%CALLRFC` macro, see “`%CALLRFC`” on page 22. For more information about using the `RFC_LOGON_INFO` macro string, see “`RFC_LOGON_INFO`” on page 30.

The following fields are available on the R/3 BAPI Connector: Logon window:

- Client** enables you to identify the SAP logon client. This field is required.
- User** enables you to identify the SAP user logon ID. This field is required.

Password	enables you to enter the SAP user password. This field is required.
Language	enables you to specify the SAP logon language. Specify the 1-byte SAP language (E for English, D for German, and so on) or the 2-byte ISO language (EN for English, DE for German, and so on). This field is required.
Destination	enables you to identify the logical destination of the saprfc.ini file, if applicable. This field is required if you use the saprfc.ini file.
Host	enables you to identify the application server host. This field is required if a specific application server will be used.
System number	enables you to identify the SAP System number if you are using a specific application server and are not using Load Balancing. This field is required if you use a specific application server.
Gateway host	enables you to identify the host name of the SAP gateway.
Gateway service	enables you to identify the service of the SAP gateway.

The following buttons are available on the R/3 BAPI Connector: Logon window:

OK	submits the R/3 BAPI connection parameters that you entered and attempts to log on to the SAP System.
Cancel	closes the R/3 BAPI Connector: Logon window without submitting the connection parameters and attempting to logon to the SAP System.
Help	displays the SAS Help for the window.

Performing Batch Operations

You can use batch operations to automate processes; for example, you can use them for overnight operations or to simplify connections for end users.

To open the connection for a batch operation, use the following:

```
%r3connb(profile=profile-name, libref=SAS-libref, function=OPEN);
```

profile-name	specifies the name of the profile that is used to save the information in the Logon to R/3 window.
SAS-libref	specifies the SAS library where the profile was saved. If this parameter is not specified, the profile is looked for in the SAS libraries WORK, SASUSER, and R3LIB (in that order).
function=OPEN	opens the connection.

To close the connection, use the following:

```
%r3connb(profile=profile-name, libref=SAS-libref, function=CLOSE);
```

or

```
%r3connb(id=connection-id, function=CLOSE);
```

profile-name	specifies the name of the profile that is used to save the information in the Logon to R/3 window.
---------------------	--

- SAS-libref** specifies the SAS library where the profile was saved. If this parameter is not specified, the profile is looked for in the SAS libraries WORK, SASUSER, and R3LIB (in that order).
- function=CLOSE** disconnects from the SAP System.
- connection_id** specifies the connection ID.

Source Parameter

Previous versions of the SAS/ACCESS Interface to R/3 included a source parameter in the batch connection. For backward compatibility, use of the source parameter as described below is still supported, although not recommended.

Submit the following macro in SAS to set up the connection for batch:

```
%r3connb (source = libref.catalog.entry.SOURCE );
```

If operating in batch mode, issue the following command in SAS to disconnect before exiting:

```
%r3connb (source = libref.catalog.entry.SOURCE, function = close);
```

Passwords

In batch processing, all parameters to log on to an SAP System are generally stored in permanent SAS catalog entries. If you do not want to store all parameters (particularly the user ID, password, and client), use the following SAS code to prompt the user for this information. In this scenario, all other connection information—the function module name, the gateway host, or the gateway service—has to be entered in the Application Setup window when you create the SAS DATA step view.

```
/*-----*/
/* This SAS program demonstrates how to prompt a user */
/* for the SAP user ID, password, client, and          */
/* language information.                               */
/* The remote processing is enclosed in comments.    */
/*-----*/

*--- reset password ---*;
  %let pwd=;

*--- set the message text ---*;
  %let message=Please enter the SAP logon parameter;

*--- prompt the user for the parameters ---*;
  %macro secure;
    %global usr pwd cli lng;
    %window R3 columns=80 rows=15
      #2 @5 message 50 protect=yes
      #4 @5 "User   : " usr 10 required=yes
      #6 @5 "Password: " pwd 10 display=no required=yes
      #8 @5 "Client  : " cli 3  required=yes
      #10 @5 "Language : " lng 1  required=yes;
    %display R3;
  %mend secure;
```

```

%secure;

/* === for remote processing ===== */

*--- macro to pass macro variables to ---*;
*--- a remote SAS session ---*;
  %macro syslput(macvar,macval,remote=);
  %let str=%str(rsubmit &remote );
  %nrstr(%let) %str(&macvar = &macval ; endrsubmit);
  &str ;
  %mend syslput;

*--- pass macro variables to the remote SAS session ---*;
  %syslput(usr,&usr);
  %syslput(pwd,&pwd);
  %syslput(cli,&cli);
  %syslput(lng,&lng);

*--- submit the code to the remote SAS session ---*;
  rsubmit;

=== end of remote processing ===== */

*--- a temporary catalog entry for the connection ---*;
*--- parameters ---*;
  %let source=work.r3conn.conn1.source;

*--- the connection id ---*;
*--- ATTENTION: This has to be the same connection ID ---*;
*--- as the one that was used to create the views. ---*;
  %let cconn=conn1;

*--- the host/port where the SAS RFC server program ---*;
*--- is running, used only in z/OS operating environment ---*;
  %let tcphost=cafe7.eur.sas.com;
  %let port=6999;

*--- set the host parameter ---*;
  %let hst=hostname;

*--- save the parameters to the temporary catalog ---*;
*--- entry ---*;
  proc display c=sapr3.sr3dbi.savconnb.scl;
  run;

*--- connect to the SAP System ---*;
  %r3connb(source=&source);

*--- use the Data Step view you have created with the ---*;
*--- SAS/ACCESS Interface to R/3 ---*;

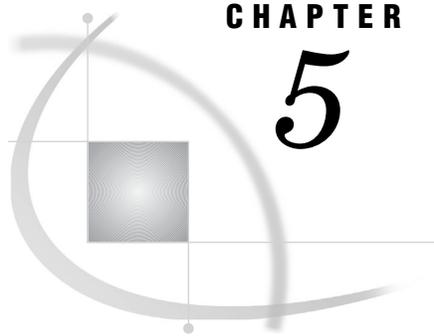
*--- close the connection to the SAP System ---*;
  %r3connb(source=&source, function=close);

```

```
/* === for remote processing
=====
    endrsubmit;

=== end of remote processing

===== */
*--- end of sample program ---*;
```



CHAPTER

5

Using the LIBNAME Statement for SAP

<i>Overview of the LIBNAME Engine for SAP</i>	37
<i>LIBNAME Statement Syntax for SAP</i>	37
<i>SAP LIBNAME Engine Options</i>	38
<i>SAS LIBNAME Statement Options for the SAP Engine</i>	50
<i>SAS Data Set Options for the SAP Engine</i>	54
<i>Examples</i>	58
<i>Example 1: Accessing the SAP Server From a Local Host</i>	58
<i>Example 2: Changing the User Logon Language</i>	58
<i>Example 3: Defining an SAP Destination from the SAP Logon Utility</i>	58
<i>Example 4: Defining an SAP Destination from the saprfc.ini File</i>	58
<i>Example 5: Accessing the SAP Server With Credentials Stored in Metadata</i>	58

Overview of the LIBNAME Engine for SAP

The SAP LIBNAME engine is a read-only engine that allows you to access tables and views from SAP Systems. For most operating systems, the SAP LIBNAME engine makes a direct connection to the SAP System. For z/OS operating environments, the SAP LIBNAME engine accesses an SAP System through a SAS RFC Server. The SAP LIBNAME engine is part of the SAS solution to access data in an SAP System.

Note: Unlike other library engines, these interfaces are read-only and do not support any forms of data set creation, deletion, or modification. Δ

LIBNAME Statement Syntax for SAP

LIBNAME *libref* R3 <Options>;

libref	specifies a valid SAS name that serves as an alias to associate SAS with the SAP tables and view. It is any SAS name when you are assigning a new libref.
R3	explicitly specifies the R/3 library engine.
<Options>	names one or more options honored by the R/3 engine, delimited with blanks.

The SAP LIBNAME statement assigns a library and allows the SAP engine to establish the connection to the SAP System. The SAP System checks the logon information submitted in the LIBNAME statement. The checked information includes the values for user, password, language, and client.

A user specifies a LIBNAME statement for the engine, and then can use that libref throughout SAS, wherever a libref is valid.

SAP LIBNAME Engine Options

The following LIBNAME options apply to the SAP LIBNAME engine.

- “ABAPFM= abap_function_name” on page 39
- “ABAP_NAMESPACE = namespace” on page 39
- “ABAPPROG= abap_program” on page 39
- “ASHOST= application_server_host” on page 40
- “AUTHDOMAIN= authentication_domain” on page 40
- “BATCH= 0 | 1 | Y | N” on page 40
- “BUFFER_SIZE= buffersize” on page 41
- “CLIENT= client” on page 41
- “DESTGROUP= destination_group” on page 42
- “DESTINATION= destination” on page 42
- “GROUP= application_server_group” on page 42
- “GWHOST= gateway_host_name” on page 42
- “GWSERV= gateway_service” on page 43
- “HOST= rfc_server_host” on page 43
- “IEEE_REVERSE= Y | N” on page 43
- “INENCODING = code_page” on page 44
- “LANGUAGE = language” on page 44
- “MAX_TABLE_JOINS = number” on page 45
- “MODE= DIRECT_SAP | DIRECT | RFC_SERVER | SERVER” on page 45
- “MSHOST= message_server_host” on page 45
- “NUMC_SAS_TYPE = N | C | \$” on page 46
- “PASSWORD= password” on page 46
- “PASSWORDX= encrypted_password” on page 46
- “PORT= rfc_server_port” on page 47
- “R3NAME= system_name” on page 47
- “RFC_STRING= additional_rfc_options” on page 48
- “SAPLOGON_ID= saplogon_id” on page 48
- SYSNR=“SYSNR= system_number” on page 48
- “TRACE= 0 | 1 | Y | N” on page 49
- “USER= user” on page 49

ABAPFM= abap_function_name

Indicates the name of the ABAP function module that SAS uses.

Valid in: SAS/ACCESS LIBNAME statement for SAP

Default value: /SAS/Z_SAS_DIALOG

Syntax

ABAPFM | ABAPFUNCTION | ABAPFUNC = abap_function_name

Indicates the name of the ABAP function module that SAS uses. The default value for this option is /SAS/Z_SAS_DIALOG.

ABAP_NAMESPACE = namespace

Indicates the namespace for ABAP functions and programs that are used by SAS.

Valid in: SAS/ACCESS LIBNAME statement for SAP

Default value: /SAS/

Syntax

ABAPNAMESPACE | ABAP_NAME_SPACE | ABAPNS | ABAP_NS |
ABAP_NAMESPACE = namespace

If the ABAP programs are installed in the customer namespace rather than in the /SAS/ namespace, then this parameter identifies where the ABAP programs are installed.

ABAPPROG= abap_program

Indicates the name of the ABAP program that SAS uses

Valid in: SAS/ACCESS LIBNAME statement for SAP

Default value: /SAS/Z_SAS_READ

Syntax

ABAPPROG | ABAPREPORT | ABAPPROGRAM = abap_program

Indicates the name of the ABAP program SAS uses. The default value for this option is /SAS/Z_SAS_READ and is set by the ABAP function module.

ASHOST= application_server_host

Indicates the host name or IP address of a specific SAP application server.

Valid in: SAS/ACCESS LIBNAME statement for SAP

Default value: none

Syntax

ASHOST | HST | RFCHOST | R3HOST = application_server_host

Indicates the host name or IP address of a specific SAP application server.

AUTHDOMAIN= authentication_domain

Indicates the name of the metadata-based authentication domain to use for associating user name and password credentials with an identity.

Valid in: SAS/ACCESS LIBNAME statement for SAP

Default value: none

Syntax

AUTHDOMAIN= authentication_domain

Indicates the metadata-based authentication domain to use for associating user name and password credentials with an identity. In a metadata-based environment, credentials for an SAP System can be stored in a metadata repository. The user can reference the authentication domain in the LIBNAME statement so that the connection to the SAP System uses the credentials stored in metadata. This is used to provide user name and password information without using the USER= and PASSWORD= LIBNAME options.

BATCH= 0|1|Y|N

Indicates whether SAS should use SAP batch jobs for the data extracts.

Valid in: SAS/ACCESS LIBNAME statement for SAP

Default value: N

Syntax

BATCH | BATCH_MODE | BATCHMODE = 0|1|Y|N

Y
specifies that SAS uses batch jobs to extract R/3 data.

N
specifies that SAS uses dialog processes to extract R/3 data.

Indicates whether SAS should use SAP batch jobs for the data extracts. The default value for this option is N.

BUFFER_SIZE= buffersize

Sets the minimum buffer size for data transfers in batch and dialog modes.

Valid in: SAS/ACCESS LIBNAME statement for SAP

Default value: 100,000 bytes

Syntax

BUFFERSIZE | BUFSIZE | BLOCK_SIZE | BLOCKSIZE | BUFFER_SIZE =
buffersize

The number of bytes must be greater than 10,000 and no more than eight digits.

CLIENT= client

Specifies the SAP logon parameter client.

Valid in: SAS/ACCESS LIBNAME statement for SAP

Default value: SAP System default

Syntax

CLIENT | CLI | RFCCLIENT | RFCCLI = client

Indicates the SAP logon parameter client. Examples for a client are 000 or 800. The default value for this option is the SAP System default. When accessing the SAP R/3 system or the SAP BW system via the LIBNAME engine, you must specify valid logon information including client, user name, password, and language. The SAP LIBNAME engine performs a logon check at OPEN time.

DESTGROUP= destination_group

Indicates the name of the destination group for batch access to the SAP System when the SAP LIBNAME engine uses direct RFC calls.

Valid in: SAS/ACCESS LIBNAME statement for SAP

Default value: SAS1

Syntax

DESTGROUP= destination_group

Destination groups are defined in table /SAS/DESTS in the SAP System. Table /SAS/DESTS defines a group of RFC destinations.

DESTINATION= destination

Indicates the destination in `saprfc.ini` if working with an `saprfc.ini` file.

Valid in: SAS/ACCESS LIBNAME statement for SAP

Default value: none

Syntax

DESTINATION | DEST | DST | DSTN = destination

Indicates the destination in `saprfc.ini` if working with an `saprfc.ini` file. If the system is an R/3 system, then this destination must also be defined in the `SIDEINFO` file for the SAP gateway.

GROUP= application_server_group

Indicates the name of the group of SAP application servers.

Valid in: SAS/ACCESS LIBNAME statement for SAP

Default value: none

Syntax

GROUP= application_server_group

Indicates the name of the group of SAP application servers if using load balancing.

GWHOST= gateway_host_name

Indicates the host name of the SAP gateway.

Valid in: SAS/ACCESS LIBNAME statement for SAP

Default value: none

Syntax

GWHOST | GATEWAY_HOST = gateway_host_name

Indicates the host name of the SAP gateway.

GWSERV= gateway_service

Indicates the service of the SAP gateway.

Valid in: SAS/ACCESS LIBNAME statement for SAP

Default value: none

Syntax

GWSERV | GATEWAY_SERVICE = gateway_service

Indicates the service of the SAP gateway.

HOST= rfc_server_host

This option applies to z/OS operating environments only. Indicates the host of the SAS RFC server to be used to connect to the SAP System.

Valid in: SAS/ACCESS LIBNAME statement for SAP

Default value: localhost

Syntax

HOST | RFC_SERVER_HOST | RFC_SERVER | SASRFC_SERVER |

SASRFC_SERVER_HOST = rfc_server_host

Indicates the host of the SAS RFC server to be used to connect to the SAP System. The default value for this option is localhost.

IEEE_REVERSE= YIN

Indicates whether floating-point numbers are byte reversed.

Valid in: SAS/ACCESS LIBNAME statement for SAP

Default value: Y or N depending on platform

Syntax

IEEE_REVERSE = Y|N

Y

Indicates that floating-point numbers are byte reversed.

N

Indicates that floating-point numbers are not byte reversed.

Indicates whether floating point-numbers are byte reversed. The default value for this option is Y for an SAP application server on Windows, and N for the other platforms.

INENCODING = code_page

Indicates the code page.

Valid in: SAS/ACCESS LIBNAME statement for SAP

Default value: none

Syntax

INENCODING = code_page

Indicates the code page. The code page identifies characters and symbols that can be printed, displayed on terminals, and used in SAP programs.

LANGUAGE = language

Indicates the SAP logon parameter language.

Valid in: SAS/ACCESS LIBNAME statement for SAP

Default value: SAP System default

Syntax

LANGUAGE | LANG | LNG | RFCLANG | RFCLNG = language

Indicates the SAP logon parameter language. The value for language is either the 2-byte ISO-language key or the 1-byte SAP language. Examples for the language include EN, DE, or E, D. The default value for this option is the SAP System default. When accessing the SAP R/3 system or the SAP BW system via the SAP LIBNAME engine, you must specify valid logon information including client, user name, password, and language. The SAP LIBNAME engine performs a logon check at OPEN time.

MAX_TABLE_JOINS = number

Indicates the number of tables that can be used in a left-outer join or an inner join in ABAP Open SQL.

Valid in: SAS/ACCESS LIBNAME statement for SAP

Default value: 25

Syntax

MAX_TABLE_JOINS | MAX_TABLES_JOIN | MAX_TABLES_JOINS
| MAX_TABLE_JOIN = number

Indicates the number of tables that can be used in a left-outer join or an inner join in ABAP Open SQL. The default value for this option is 25.

MODE= DIRECT_SAP | DIRECT | RFC_SERVER | SERVER

Indicates the mode for the SAP LIBNAME engine.

Valid in: SAS/ACCESS LIBNAME statement for SAP

Default value: DIRECT_SAP for operating environments other than z/OS. The default value for z/OS operating environments is RFC_SERVER.

Syntax

MODE = DIRECT_SAP | DIRECT | RFC_SERVER | SERVER

Values DIRECT_SAP and DIRECT are supported on all platforms except z/OS.
Values RFC_SERVER and SERVER are supported on z/OS only.

MSHOST= message_server_host

Indicates the host name of the message server.

Valid in: SAS/ACCESS LIBNAME statement for SAP

Default value: none

Syntax

MSHOST= message_server_host

Indicates the host name of the message server (if using load balancing).

NUMC_SAS_TYPE = NICI\$

Indicates the SAS type for ABAP type NUMC.

Valid in: SAS/ACCESS LIBNAME statement for SAP

Default value: C

Syntax

NUMC_SAS_TYPE | NUMC_SASTYPE | NUMC_TYPE | NUMC = N | C | \$

N

Number

C

Character

\$

Character

Indicates the SAS type for ABAP type NUMC. The default is C.

PASSWORD= password

Indicates the SAP logon parameter password.

Valid in: SAS/ACCESS LIBNAME statement for SAP

Default value: none

Syntax

PASSWORD | PASSWD | PWD | PW | PASS = password

Indicates the SAP logon parameter password. When accessing the SAP R/3 system or the SAP BW system via the LIBNAME engine, you must specify valid logon information including client, user name, password, and language. The SAP LIBNAME engine performs a logon check at OPEN time.

PASSWORDX= encrypted_password

Indicates the SAP logon parameter password in an encrypted form.

Valid in: SAS/ACCESS LIBNAME statement for SAP

Default value: none

Syntax

PASSWORDX | PASSWDX | PWDX | PWX | PASSX= encrypted password

Indicates the SAP logon parameter password in an encrypted form. It uses the encryption type that is used in the SAS/ACCESS Interface to R/3, Version 8. It is supported for compatibility with the Version 8 connection profile.

PORT= rfc_server_port

This option applies to z/OS operating environments only. Indicates the port number of the SAS RFC server to be used to connect to the SAP System.

Valid in: SAS/ACCESS LIBNAME statement for SAP

Default value: 6999

Syntax

PORT | RFC_SERVER_PORT | RFCSERVERPORT | SASRFC_SERVER_PORT |
SASRFCSERVERPORT = rfc_server_port

Indicates the port number of the SAS RFC server to be used to connect to the SAP System. The default value for this option is 6999.

R3NAME= system_name

Indicates the name of the R/3 system.

Valid in: SAS/ACCESS LIBNAME statement for SAP

Default value: none

Syntax

R3NAME= system_name

Indicates the name of the R/3 system (if using load balancing).

RFC_STRING= additional_rfc_options

Indicates additional logon or connection parameters for the RfcOpenEx() call.

Valid in: SAS/ACCESS LIBNAME statement for SAP

Default value: none

Syntax

RFC_STRING | RFCSTRING | RFC_OPTIONS_EXT | RFCOPENEX |
ADDITIONAL_RFC_OPTIONS = additional_rfc_options

Indicates additional log on or connection parameters for the RfcOpenEx() call. The connection parameters are used in the RfcOpenEx() call to log on to the SAP System. With this option, parameters that are not SAP LIBNAME options can be passed to the RfcOpenEx() call.

RFC_STRING Example

```
RFC_STRING = "ABAP_DEBUG=1"
```

Note: When using RFC_STRING="ABAP_DEBUG=1", the ABAP debugger is invoked to debug the ABAP programs used by the engine. The SAP GUI on the SAS RFC Server host is required to use the ABAP_DEBUG=1 option. △

SAPLOGON_ID= saplogon_id

Indicates the string defined for SAPLOGON on 32-bit Windows.

Valid in: SAS/ACCESS LIBNAME statement for SAP

Default value: none

Syntax

SAPLOGON_ID= saplogon_id

Indicates the string defined for SAPLOGON on 32-bit Windows.

SYSNR= system_number

Indicates the SAP System number.

Valid in: SAS/ACCESS LIBNAME statement for SAP

Default value: none

Syntax

SYSNR | SYS | SYSTEM | SYSNO = system_number

Indicates the R/3 system number. It is the 2-byte code that identifies the system on the host. For example: 00 or 01.

TRACE= 0 |1|Y|N

Indicates whether SAS should trace the RFC requests.

Valid in: SAS/ACCESS LIBNAME statement for SAP

Default value: 0

Syntax

TRACE= 0 |1|Y|N

- 0**
Specifies that RFC trace is switched off.
- 1**
Specifies that RFC trace is switched on.
- Y**
Specifies that RFC trace is switched on.
- N**
Specifies that RFC trace is switched off.

Indicates whether SAS should trace the RFC requests. If the trace option is switched on, SAS writes log information into a file. The RFC library logs messages in the file dev_rfc. The default value for this option is 0.

USER= user

Indicates the SAP logon parameter user.

Valid in: SAS/ACCESS LIBNAME statement for SAP

Default value: none

Syntax

USER | USR | RFCUSER | USERNAME | USERID= user

When accessing the SAP System via the LIBNAME engine, you must specify valid logon information including the client, user name, password, and language. The SAP System performs a logon check at OPEN time.

SAS LIBNAME Statement Options for the SAP Engine

The following list describes the SAP interface support for SAS /ACCESS LIBNAME options and presents default values where applicable. For further information, refer to *SAS/ACCESS for Relational Databases: Reference*.

- “CONNECTION= LIBNAME Option” on page 50
- “CONNECTION_GROUP= LIBNAME Option” on page 50
- “DBGEN_NAME= LIBNAME Option” on page 51
- “DBPROMPT= LIBNAME Option” on page 51
- “DBSASLABEL= COMPAT | NONE | DBMS ”on page 52
- “DEFER= LIBNAME Option” on page 52
- “DIRECT_SQL= LIBNAME Option” on page 53
- “MULTI_DATASRC_OPT= LIBNAME Option” on page 53
- “REREAD_EXPOSURE= LIBNAME Option” on page 53
- “SPOOL= LIBNAME Option” on page 54

CONNECTION= LIBNAME Option

Specifies whether operations on a single libref share a connection to the DBMS and whether operations on multiple librefs share a connection to the DBMS.

Valid in: SAS/ACCESS LIBNAME statement

DBMS support: DB2 UNIX/PC, DB2 z/OS, HP Neoview, Informix, Microsoft SQL Server, MySQL, Netezza, ODBC, OLE DB, Oracle, Sybase, Teradata, SAP

Default value: DBMS-specific. The default value for SAP is SHAREDREAD

Syntax

CONNECTION= SHAREDREAD | UNIQUE | SHARED | GLOBALREAD | GLOBAL

For more information about the CONNECTION= option, see the LIBNAME statement for relational databases in *SAS/ACCESS for Relational Databases: Reference*.

CONNECTION_GROUP= LIBNAME Option

Causes operations on multiple librefs and on multiple Pass-Through Facility CONNECT statements to share a connection to the DBMS.

Valid in: SAS/ACCESS LIBNAME statement

DBMS support: DB2 UNIX/PC, DB2 z/OS, HP Neoview, Informix, Microsoft SQL Server, MySQL, Netezza, ODBC, OLE DB, Oracle, Sybase, Teradata, SAP

Default value: none

Syntax

CONNECTION_GROUP= *connection-group-name*

For more information about the CONNECTION_GROUP= option, see the LIBNAME statement for relational databases in *SAS/ACCESS for Relational Databases: Reference*.

DBGEN_NAME= LIBNAME Option

Specifies how SAS automatically renames DBMS columns (when they contain characters that SAS does not allow, such as \$) to valid SAS variable names.

Valid in: SAS/ACCESS LIBNAME statement

DBMS support: DB2 UNIX/PC, DB2 z/OS, HP Neoview, Informix, Microsoft SQL Server, MySQL, Netezza, ODBC, OLE DB, Oracle, Sybase, Teradata, SAP

Default value: DBMS

Syntax

DBGEN_NAME= DBMS | SAS

For more information about the DBGEN_NAME= option, see the LIBNAME statement for relational databases in the *SAS/ACCESS for Relational Databases: Reference*.

DBPROMPT= LIBNAME Option

Specifies whether SAS displays a window that prompts the user to enter DBMS connection information before connecting to the DBMS in interactive mode.

Valid in: SAS/ACCESS LIBNAME statement

DBMS support: DB2 UNIX/PC, HP Neoview, Informix, Microsoft SQL Server, MySQL, Netezza, ODBC, Oracle, Sybase, Teradata, SAP

Default value: NO

Syntax

DBPROMPT=YES | NO

For more information about the DBPROMPT= option, see the LIBNAME statement for relational databases in the *SAS/ACCESS for Relational Databases: Reference*.

DBSASLABEL= COMPAT | NONE | DBMS

Specifies the column labels an engine uses.

Valid in: SAS/ACCESS LIBNAME statement for SAP

DBMS support: DB2 UNIX/PC, DB2 z/OS, HP Neoview, Informix, Microsoft SQL Server, MySQL, Netezza, ODBC, OLE DB, Oracle, Sybase, Teradata, SAP

Default value: COMPAT

Syntax

DBSASLABEL= COMPAT | NONE

COMPAT	specifies that the column labels are compatible with the engine's behavior in previous SAS releases. The SAP engine returns the short descriptive text for the columns of the SAP table. It reads the column label from the SAP data dictionary. This is the default value.
NONE	specifies that no column label information is returned.

For more information about the DBSASLABEL= option, see the LIBNAME statement for relational databases in the *SAS/ACCESS for Relational Databases: Reference*.

DEFER= LIBNAME Option

Specifies when the connection to the DBMS occurs.

Valid in: SAS/ACCESS LIBNAME statement

DBMS support: DB2 UNIX/PC, DB2 z/OS, HP Neoview, Informix, Microsoft SQL Server, MySQL, Netezza, ODBC, OLE DB, Oracle, Sybase, Teradata, SAP

Default value: NO

Syntax

DEFER= NO | YES

For more information about the DEFER= option, see the LIBNAME statement for relational databases in the *SAS/ACCESS for Relational Databases: Reference*.

DIRECT_SQL= LIBNAME Option

Lets you specify whether generated SQL is passed to the DBMS for processing.

Valid in: SAS/ACCESS LIBNAME statement

DBMS support: DB2 UNIX/PC, DB2 z/OS, HP Neoview, Informix, Microsoft SQL Server, MySQL, Netezza, ODBC, OLE DB, Oracle, Sybase, Teradata, SAP

Default value: YES

Syntax

DIRECT_SQL= YES | NO | NONE | NOGENSQL | NOWHERE | NOFUNCTIONS
| NOMULTOUTJOINS

For more information about the DIRECT_SQL= option, see the LIBNAME statement for relational databases in the *SAS/ACCESS for Relational Databases: Reference*.

MULTI_DATASRC_OPT= LIBNAME Option

Used in place of DBKEY to improve performance when processing a join between two data sources.

Valid in: SAS/ACCESS LIBNAME statement

DBMS support: DB2 z/OS, DB2 UNIX/PC, Informix, ODBC, OLE DB, Oracle, Microsoft SQL Server, Sybase, Teradata, SAP

Default value: NONE

Syntax

MULTI_DATASRC_OPT=NONE | IN_CLAUSE

For more information about the MULTI_DATASRC_OPT= option, see the LIBNAME statement for relational databases in the *SAS/ACCESS for Relational Databases: Reference*.

REREAD_EXPOSURE= LIBNAME Option

Specifies whether the SAS/ACCESS engine behaves like a random access engine for the scope of the LIBNAME statement.

Valid in: SAS/ACCESS LIBNAME statement

DBMS support: DB2 UNIX/PC, DB2 z/OS, HP Neoview, Informix, Microsoft SQL Server, MySQL, Netezza, ODBC, OLE DB, Oracle, Sybase, Teradata, SAP

Default value: NO

Syntax

REREAD_EXPOSURE= NO | YES

For more information about the REREAD_EXPOSURE= option, see the LIBNAME statement for relational databases in the *SAS/ACCESS for Relational Databases: Reference*.

SPOOL= LIBNAME Option

Specifies whether SAS creates a utility spool file during read transactions that read data more than once.

Valid in: SAS/ACCESS LIBNAME statement

DBMS support: DB2 UNIX/PC, DB2 z/OS, HP Neoview, Informix, Microsoft SQL Server, MySQL, Netezza, ODBC, OLE DB, Oracle, Sybase, Teradata, SAP

Default value: YES

Syntax

SPOOL= YES | NO

For more information about the SPOOL= option, see the LIBNAME statement for relational databases in the *SAS/ACCESS for Relational Databases: Reference*.

SAS Data Set Options for the SAP Engine

The following list describes the SAP interface support for SAS /ACCESS data set options and presents default values where applicable. For further information, refer to the *SAS/ACCESS for Relational Databases: Reference*.

“BATCH= 0|1|Y|N ”on page 54

“DBCONDITION= Data Set Option” on page 55

“DBGEN_NAME= Data Set Option” on page 55

“DBKEY= Data Set Option” on page 56

“DBMASTER= Data Set Option” on page 56

“NULLCHAR= Data Set Option” on page 56

“NULLCHARVAL= Data Set ”on page 57

“REREAD_EXPOSURE= Data Set Option” on page 57

BATCH= 0|1|Y|N

Indicates whether the SAS should use SAP batch jobs for the data extracts.

Valid in: DATA and PROC steps (when accessing DBMS data using SAS/ACCESS software)

Default value: N

Syntax

BATCH | BATCH_MODE | BATCHMODE = 0|1|Y|N

Y

SAS uses batch jobs to extract SAP data.

N

SAS uses dialog processes to extract SAP data.

Indicates whether SAS should use SAP batch jobs for the data extracts. The default value for this option is N.

Note: The BATCH= option is SAP-specific. Δ

DBCONDITION= Data Set Option

Specifies criteria for subsetting and ordering DBMS data.

Valid in: DATA and PROC steps (when accessing DBMS data using SAS/ACCESS software)

DBMS support: DB2 UNIX/PC, DB2 z/OS, HP Neoview, Informix, Microsoft SQL Server, MySQL, Netezza, ODBC, OLE DB, Oracle, Sybase, Teradata, SAP

Default value: none

Syntax

DBCONDITION="*DBMS-SQL-query-clause*"

For more information about the DBCONDITION= option, see the data set options for relational databases section in *SAS/ACCESS for Relational Databases: Reference*.

DBGEN_NAME= Data Set Option

Specifies how SAS automatically renames columns (when they contain characters that SAS does not allow, such as \$) to valid SAS variable names.

Valid in: DATA and PROC steps (when accessing DBMS data using SAS/ACCESS software)

DBMS support: DB2 UNIX/PC, DB2 z/OS, HP Neoview, Informix, Microsoft SQL Server, MySQL, Netezza, ODBC, OLE DB, Oracle, Sybase, Teradata, SAP

Default value: DBMS

Syntax

DBGEN_NAME=DBMS | SAS

For more information about the DBGEN_NAME= option, see the data set options for relational databases section in *SAS/ACCESS for Relational Databases: Reference*.

DBKEY= Data Set Option

Specifies a key column to optimize DBMS retrieval.

Valid in: DATA and PROC steps (when accessing DBMS data using SAS/ACCESS software)

DBMS support: DB2 UNIX/PC, DB2 z/OS, HP Neoview, Informix, Microsoft SQL Server, MySQL, Netezza, ODBC, OLE DB, Oracle, Sybase, Teradata, SAP

Default value: none

Syntax

DBKEY=(<'>column-1<'> <... <'>column-n<'>>)

For more information about the DBKEY= option, see the data set options for relational databases section in *SAS/ACCESS for Relational Databases: Reference*.

DBMASTER= Data Set Option

Designates which table is the larger table when you are processing a join that involves tables from two different types of databases.

Valid in: DATA and PROC steps (when accessing DBMS data using SAS/ACCESS software)

DBMS support: DB2 UNIX/PC, DB2 z/OS, HP Neoview, Informix, Microsoft SQL Server, MySQL, Netezza, ODBC, OLE DB, Oracle, Sybase, Teradata, SAP

Default value: none

Syntax

DBMASTER=YES

For more information about the DBMASTER= option, see the data set options for relational databases section in *SAS/ACCESS for Relational Databases: Reference*.

NULLCHAR= Data Set Option

Indicates how missing SAS character values are handled during insert, update, DBINDEX=, and DBKEY= processing.

Valid in: DATA and PROC steps (when accessing DBMS data using SAS/ACCESS software)

DBMS support: DB2 UNIX/PC, DB2 z/OS, HP Neoview, Informix, Microsoft SQL Server, MySQL, Netezza, ODBC, OLE DB, Oracle, Sybase, Teradata, SAP

Default value: SAS

Syntax

NULLCHAR= SAS | YES | NO

For more information about the NULLCHAR= option, see the data set options for relational databases section in *SAS/ACCESS for Relational Databases: Reference*.

NULLCHARVAL= Data Set

Defines the character string that replaces missing SAS character values during insert, update, DBINDEX=, and DBKEY= processing.

Valid in: DATA and PROC steps (when accessing DBMS data using SAS/ACCESS software)

DBMS support: DB2 UNIX/PC, DB2 z/OS, HP Neoview, Informix, Microsoft SQL Server, MySQL, Netezza, ODBC, OLE DB, Oracle, Sybase, Teradata, SAP

Default value: a blank character

Syntax

NULLCHARVAL='character-string'

For more information about the NULLCHARVAL= option, see the data set options for relational databases section in *SAS/ACCESS for Relational Databases: Reference*.

REREAD_EXPOSURE= Data Set Option

Specifies whether the SAS/ACCESS engine behaves like a random access engine for the scope of the LIBNAME statement.

Valid in: SAS/ACCESS LIBNAME statement

DBMS support: DB2 z/OS, DB2 UNIX/PC, Informix, ODBC, OLE DB, Oracle, Microsoft SQL Server, Sybase, Teradata

Default value: NO

Syntax

REREAD_EXPOSURE= NO | YES

For more information about the REREAD_EXPOSURE= option, see the LIBNAME statement for relational databases section in *SAS/ACCESS for Relational Databases: Reference*.

Examples

Example 1: Accessing the SAP Server From a Local Host

In this example for the z/OS operating environment, the SAS RFC server has been started on the local host on port 6998. You want to access the SAP System on sapr3srv.sup.com. The system number for the system is 03. You specify the entire set of user information, for example, client, user, password, and language.

```
libname mylib r3 user=TEST password=MYPASS client=800 language=EN
      ashost=sapr3srv.sup.com sysnr=03
      host=localhost port=6998;
```

Note: If the SAS RFC server has been started on the SAS system host on port 6999, you do not need to define the host or port because 6999 is the default. \triangle

Example 2: Changing the User Logon Language

You want to access the BW system on whjapp01 and the system number is 06. This time you want to log on using the German language. If you log on using that language, the column labels will all be in German.

```
libname bwides r3 user=Test password=pwd client=800 language=DE
      ashost=whjapp01 sysnr=06;
```

Example 3: Defining an SAP Destination from the SAP Logon Utility

The SAP System you want to access is defined as SI9 in the SAP logon utility.

```
libname mylib r3 user=test password=secret client=800 language=E
      saplogon_id=SI9;
```

Example 4: Defining an SAP Destination from the saprfc.ini File

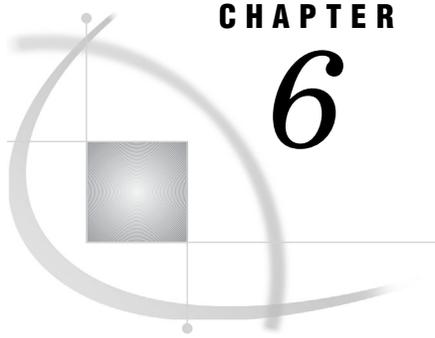
The SAP System you want to access is defined by the logical destination bwides in the file saprfc.ini.

```
libname mylib r3 user=test password=secret client=800 language=EN
      destination=bwides;
```

Example 5: Accessing the SAP Server With Credentials Stored in Metadata

User name and password credentials for the SAP System you want to access are stored in metadata. The credentials are associated with an authentication domain named SAPAuth. The SAS Metadata Server is named meta.sup.com.

```
options metaserver=meta.sup.com metauser=test metapass=secret;  
libname bwides r3 authdomain=SAPAuth client=800  
      language=EN ahost=sapr3srv.sup.com sysnr=03;
```

CHAPTER

6

The CALLRFC Procedure

Overview: CALLRFC Procedure	61
Syntax: CALLRFC Procedure	61
PROC CALLRFC Statement	62
CALL FUNCTION Statement	66
DESCRIBE FUNCTION Statement	67
Examples: CALLRFC Procedure	70
Example 1: Displaying RFC_SYSTEM_INFO	70
Example 2: Using the Output Delivery System with the DESCRIBE FUNCTION statement	71

Overview: CALLRFC Procedure

The CALLRFC procedure executes Remote Function Calls (RFC) or RFC-compatible functions on an SAP System.

Any Advanced Business Applications Programming (ABAP) function modules called by the CALLRFC procedure must be:

- RFC enabled
- have no dialog boxes
- synchronous

Syntax: CALLRFC Procedure

```

PROC CALLRFC<options>;
  CALL <FUNCTION> 'function-name'
    <EXPORTING parameter-1 |parameter-1.field-1=value-1 < ...
      parameter-n=value-n>>
    <IMPORTING parameter-1=SAS-dataset-name-1 < ...
      parameter-n=SAS-dataset-name-n>>
    <INTABLES table-parameter-1=SAS-dataset-name-1 <...
      table-parameter-n=SAS-dataset-name-n>>
    <TABLES table-parameter-1=SAS-dataset-name-1 <...
      table-parameter-n=SAS-dataset-name-n>>
    <CALLBACK <callback-parameter-1=SAS-dataset-name-1 <...
      table-parameter-n=SAS-dataset-name-n>>> </ default = SAS-library>;

```

DESCRIBE <FUNCTION> 'function-name' <SHORT>;

Task	Statement
Specify connection and log on information for an SAP System, in preparation for calling a Remote Function Call or RFC-enabled module on the SAP System.	"PROC CALLRFC Statement" on page 62
Call an RFC on the RFC System	"CALL FUNCTION Statement" on page 66
Read metadata about an RFC and write the information to the SAS log or a SAS data set	"DESCRIBE FUNCTION Statement" on page 67

PROC CALLRFC Statement

Invokes the CALLRFC procedure and specifies the connection and logon information of an SAP System. The connection to the SAP System is established and the logon information is verified.

PROC CALLRFC <options>;

Task	Option
Specify the name of the ABAP function module that SAS uses internally for metadata requests	on page 63
Specify the name of the ABAP program that the SAS uses	on page 63
Specify connection and logon parameters	
Accessing the SAP System using connection and logon parameters associated with a libref	on page 64
Specify additional logon or connection parameters	on page 65
Specify logon parameters	
Specify the SAP logon parameter for user	on page 66
Specify the SAP logon parameter for password	on page 65
Specify the SAP logon parameter for password in an encrypted form	on page 65
Specify the SAP logon parameter for client	on page 64
Specify the SAP logon parameter for language	on page 64
Specify connection parameters	

Task	Option
Specify the host name or IP address of a specific application server	on page 64
Specify the destination defined in saprfc.ini for connecting to the SAP System	on page 64
Specify the name for a group of application servers	on page 64
Specify the host name of the SAP gateway	on page 64
Specify the service of the SAP gateway	on page 64
Specify the host name of the Message Server for the SAP System	on page 65
Specify the name of the SAP System	on page 65
Specify the string defined for SAPLOGON on 32-bit Windows	on page 65
Specify the SAP System number	on page 65
Specify SAS RFC Server parameters (z/OS only)	
Specify the host name of the SAS RFC Server	on page 64
Specify the port number of the SAS RFC Server	on page 65
Specify the mode for the CALLRFC procedure	on page 65
Specify data type and handling options	
Specify whether floating-point numbers are byte-reversed	on page 64
Specify the SAS data type for the ABAP type NUMC	on page 65
Enable tracing of RFC requests	on page 66

Options

ABAPFM='abap-function-name'

specifies the name of the ABAP function module that SAS uses internally for metadata requests.

Note: This option is not the name of the ABAP function that is called in the CALL FUNCTION statement. Δ

Aliases ABAPFUNCTION=, ABAPFUNC=

Default Z_SAS_DIALOG

ABAPPROG='abap-program'

specifies the name of the ABAP program that SAS uses.

Aliases ABAPREPORT=, ABAPPROGRAM=

Default Z_SAS_READ and is set by the ABAP function module

ASHOST='SAP-application-server-host'

specifies the host name or IP address of a specific application server. This option is used for SAP R/3 when it is not configured for load balancing.

Aliases HST=, RFCHOST=, R3HOST=

CLIENT=SAP-client-number

specifies the SAP logon parameter for client. Examples for a client are 000 or 800. This is a required SAP LIBNAME engine option. The SAP System performs a logon check at OPEN time.

Aliases CLI=, RFCCLIENT=, RFCCLI=

Default The default value for this option is the SAP System default.

DESTINATION='SAP-destination'

specifies the destination in `saprfc.ini` if working with a `saprfc.ini` file for connecting to the SAP System. If the SAP System is an R/3 system this destination must also be defined in the 'sideinfo' for the SAP gateway.

Aliases DEST=, DST=, DSTN=

GROUP='SAP-application-server-group'

specifies the name for the group of application servers when the SAP System is configured for load balancing.

GWHOST='SAP-gateway-host-name'

specifies the host name of the SAP gateway.

Alias GATEWAY_HOST=

GWSERV='SAP-gateway-service'

specifies the service of the SAP gateway.

Alias GATEWAY_SERVICE

HOST='RFC-server-host'

This option is supported in z/OS operating environments only. This option specifies the host name of the SAS RFC Server.

Aliases RFC_SERVER_HOST=, RFC_SERVER=, SASRFC_SERVER=, SASRFC_SERVER_HOST=

Default localhost

IEEE_REVERSE=flag

specifies whether floating-point numbers are byte-reversed.

Default Y when the SAP R/3 application server is running on Windows
N for all other platforms

Alias IE3_REVERSE

LANGUAGE= SAP-logon-language

specifies the SAP logon parameter for language. The value for language is either the 2-byte ISO-language key or the 1-byte SAP-language. Examples for the language are EN, DE or E, D. The SAP System performs a logon check at OPEN time.

Aliases LANG=, LNG=, RFCLANG=, RFCLNG=

Default The default value for this option is the SAP System default.

LIBREF=SAP-engine-libref

specifies the libref for the library used to access the SAP System. The connection and logon information is copied from the libref. Because the connection and logon information is copied from the libref, this is the easiest way of using PROC CALLRFC.

MODE=*DIRECT_SAP* | *RFC_SERVER*

specifies the mode for the SAP LIBNAME engine. If *DIRECT_SAP* is specified, the SAP LIBNAME engine uses RFC calls to the SAP System directly. If *RFC_SERVER* is specified, a SAS RFC Server is used to communicate with the SAP System.

Default The default for z/OS operating environments is *RFC_SERVER*, on all other hosts it is *DIRECT_SAP*.

MSHOST=*'SAP-message-server-host'*

specifies the host name of the Message Server when the SAP System is configured for load balancing.

NUMC_SAS_TYPE=*'N'* | *'C'* | *'\$'*

specifies the SAS data type for the ABAP type NUMC. In SAS 8 generated views, it was N, but C is a better match. Setting *NUMC_SAS_TYPE='N'* ensures SAS 8 compatibility. The macro %R3CONN sets this option to 'N' by default.

Aliases NUMC_SASTYPE=, NUMC_TYPE=, NUMC=

PASSWORD=*'SAP-password'*

specifies the SAP logon parameter for password. When using the SAP LIBNAME engine, either use the USER and PASSWORD options to specify logon credentials, or use the AUTHDOMAIN option to retrieve credentials from metadata. If single sign-on is used, this option is not needed. The SAP System performs a logon check at OPEN time.

Aliases PASSWD=, PWD=, PW=, PASS=

PASSWORDX=*'encrypted-SAP-password'*

specifies the SAP logon parameter for password in an encrypted form. It uses the encryption available with SAS/ACCESS Interface to R/3 for SAS version 8 and is supported for compatibility with the V8 connection profile.

Aliases PASSWDX=, PWDX=, PWX=, PASSX=

PORT=*'RFC-server-port'*

This option is supported in z/OS operating environments only. This option specifies the port number of the SAS RFC Server that is used to connect to the SAP System.

Aliases RFC_SERVER_PORT=, RFCSERVERPORT=, SASRFC_SERVER_PORT=, SASRFCSERVERPORT=

Default 6999

R3NAME=*'SAP-system-name'*

specifies the name of the SAP R/3 system when the SAP System is configured for load balancing.

RFC_STRING=*'additional-RFC-options'*

specifies additional logon or connection parameters for the RfcOpenEx() call. This call is used to log on the SAP System. Using this option enables passing parameters to the RfcOpenEx call that are not SAP LIBNAME engine options. This option can be used to support future extensions of the RfcOpenEx call.

Alias RFCSTRING=, RFC_OPTIONS_EXT=, RFCOPENEX=, ADDITIONAL_RFC_OPTIONS=

SAPLOGON_ID=*'SAP-logon-id'*

specifies the string defined for SAPLOGON on 32-bit Windows.

SYSNR=*SAP-system-number*

specifies the SAP R/3 system number. This option is used for SAP R/3 when it is not configured for load balancing.

Aliases SYS=, SYSTEM=, SYSNO=

TRACE='RFC-trace-flag'

determines whether the SAP RFC library traces the RFC requests. The default directory location is read from the RFC_TRACE_DIR environment variable. The rfc library logs messages in the file dev_rfc in the same directory.

- 0 or N : RFC tracing is disabled
- 1 or Y : RFC tracing is enabled

Default 0

USER=SAP-userID

specifies the SAP logon parameter for user. When using the SAP LIBNAME engine, either use the USER and PASSWORD options to specify logon credentials, or use the AUTHDOMAIN option to retrieve credentials from metadata. If single sign-on is used, this option is not needed. The SAP System performs a logon check at OPEN time.

Aliases USR=, RFCUSER=, USERNAME=, USERID=

CALL FUNCTION Statement

Call an RFC on the SAP System. The CALL FUNCTION statement can be called multiple times within a PROC CALLRFC step. The function calls are executed in sequence.

CALL <FUNCTION> 'function-name'

```
<EXPORTING parameter-1 | parameter-1.field-1=value-1 < ...
  parameter-n=value-n>>
<IMPORTING parameter-1=SAS-dataset-name-1 < ...
  parameter-n=SAS-dataset-name-n>>
<INTABLES table-parameter-1=SAS-dataset-name-1 <...
  table-parameter-n=SAS-dataset-name-n>>
<TABLES table-parameter-1=SAS-dataset-name-1 <...
  table-parameter-n=SAS-dataset-name-n>>
<CALLBACK <callback-parameter-1=SAS-dataset-name-1 <...
  table-parameter-n=SAS-dataset-name-n>>> </ default = SAS-library>;
```

Required Arguments

function-name

specifies the name of the ABAP function. The function name is a literal, and it must be enclosed in quotes. The function name is converted to uppercase characters. The ABAP function must be RFC callable. It must not have any dialog and it must be synchronous.

Options

EXPORTING

specifies fields or structures to be passed to the function module. The parameters parameter-1 through parameter-n are defined in the function interface as import parameters.

IMPORTING

specifies the fields or structures to be passed from the function module back to the SAS System. The parameters parameter-1 through parameter-n are defined in the function interface as export parameters.

INTABLES

specifies the SAS data sets to be passed to the function module. The parameters table-parameter-1 through table-parameter-n are defined in the function interface as table parameters. SAS converts the SAS data sets into internal tables and passes it to the function module. The variable names in the SAS data set must match the field names of the internal table as defined for the function interface.

TABLES

FIXME: totally changed... specifies names for the SAS data sets to be created from the internal tables in the function module. The parameters table-parameter-1 through table-parameter-n are defined in the function interface as table parameters.

CALLBACK

specifies names for the SAS data sets to be created from the internal tables in the function module. The parameters are not defined in the function interface. The function module must implement a callback mechanism.

If the callback table name matches any of the names in the list of callback-parameters specified in the function call, the data is saved into the specified SAS data set. If the name does not match, but the default library is specified, then the data is written to *libref.table-name*.

DESCRIBE FUNCTION Statement

Reads metadata about an RFC and writes the information to the SAS log or to a SAS data set (using ODS). The DESCRIBE FUNCTION statement can be called multiple times within a PROC CALLRFC step.

```
DESCRIBE <FUNCTION> 'function-name' <SHORT>;
```

Required Arguments*function-name*

specifies the name of the ABAP function. The function name is a literal and it must be enclosed in quotes. The function name is converted to uppercase characters.

Options**SHORT**

specifies that parameter metadata only is printed in the SAS log. The field metadata is not printed.

Details

General Attributes

In SAS listing output, the function name is printed. If ODS output is redirected to a SAS data set, the structure of the data set is as follows:

Name	Type	Label	Description
Label1			contains the attribute name, 'Function Name'
cValue1			contains the value for character attributes, for example, 'RFC_SYSTEM_INFO'
nValue1			contains the value for numeric attributes. For the CALLRFC procedure no numeric attributes are listed.

Parameters

For each parameter, the following information is provided. If ODS output is redirected to SAS data sets, the structure of the data set follows:

Name	Type	Label	Description
class	Text 1	Class	Parameter class: <input type="checkbox"/> I for input <input type="checkbox"/> O for output <input type="checkbox"/> T for table
name	Text 32	Parameter	Parameter name
intlen	Number 8	Internal length	Internal length in byte (ABAP)
leng	Number 8	Length (number of characters)	
outlen	Number 8	Output length	
decimals	Number 8	Number of decimal places	
datatype	Text 5	ABAP data type	
inttype	Text 1	ABAP data type (internal)	
label	Text 256	Label	
type	Number 8	Variable Type	SAS variable type <input type="checkbox"/> 0 - no type, might be a structure <input type="checkbox"/> 1 - numeric <input type="checkbox"/> 2 - character
length	Number 8	Variable Length	SAS variable length
format	Text 32	Variable Format	SAS variable format

Name	Type	Label	Description
formatl	Number 8	Format Length	SAS variable format length
formatd	Number 8	Number of Format Decimals	SAS variable format decimals
informat	Text 32	Variable Informat	SAS variable informat
informatl	Number 8	Informat Length	SAS variable informat length
informatd	Number 8	Number of Informat Decimals	SAS variable informat decimals

Columns

For each column of each structure (input, output, and tables) the following information (see table) is listed. If the ODS output is redirected to data sets the structure of the data set follows:

Name	Type	Label	Description
parm	Text 32	Parameter	
name	Text 32	Column	
offset	Number 8	Offset	
leng	Number 8	Length (number of characters)	
intlen	Number 8	Internal length	
outlen	Number 8	Output length	
decimals	Number 8	Number of decimal places	
datatype	Text 5	ABAP data type	
inttype	Text 1	ABAP data type (internal)	
label	Text 256	Label	
type	Number 8	Variable Type	SAS variable type <input type="checkbox"/> 0 - no type, might be a structure <input type="checkbox"/> 1 - numeric <input type="checkbox"/> 2 - character
length	Number 8	Variable Length	SAS variable length
format	Text 32	Variable Format	SAS variable format
formatl	Number 8	Format Length	SAS variable format length
formatd	Number 8	Number of Format Decimals	SAS variable format decimals
informat	Text 32	Variable Informat	SAS variable informat

Name	Type	Label	Description
informatl	Number 8	Informat Length	SAS variable informat length
informatd	Number 8	Number of Informat Decimals	SAS variable informat decimals

Examples: CALLRFC Procedure

Example 1: Displaying RFC_SYSTEM_INFO

This example shows an example of retrieving the RFC_SYSTEM_INFO and displaying the information.

Program

```
proc callrfc ashost="sapr3srv.sup.com" sysnr='03' user='USER'
    passwd='PASS' client='800';
    call 'RFC_SYSTEM_INFO' importing rfcsi_export=work.rfcsi;
run;

proc transpose data=rfcsi out=rfcsi_transposed
    (drop=_name_ rename=(col2=Value _label_=Column));
    var _character_ _numeric_;
run;

proc print data=rfcsi_transposed;
run;
```

Output: Listing

Obs	Column	Value
1	RFC log version	011
2	Character set (SAP name)	1100
3	Integer format (1 / 2 = little / big endian)	LIT
4	Floating point format (1=IEEE, 2=IBM/370 format)	IE3
5	Logical destination (specified in function call)	sapr3srv_SRV_03
6	Character field of length 8	sapr3srv
7	R/3 System, name of R/3 System	SRV
8	R/3 System, name of R/3 System	SRV
9	Database host name	sapr3srv
10	R/3 System, name of central database system	MSSQL
11	R/3 System, system release	46C
12	RFC: SAP machine ID	560
13	R/3 System, operating system of application server	Windows NT
14	Time zone (difference from UTC in seconds)	-18000
15	Date and time, Daylight savings time flag	X

16	IP address	192.168.1.111
17	Kernel Release	46D
18	R/3 System, Name of application server	sapr3srv
19	Reserve field in RFCSI	

Example 2: Using the Output Delivery System with the DESCRIBE FUNCTION statement

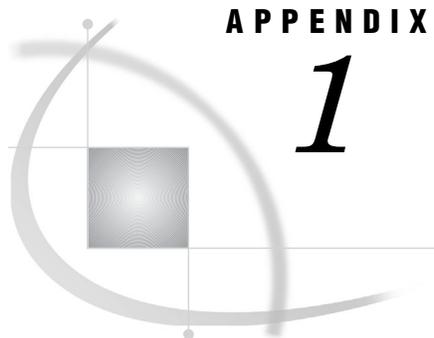
This example shows using the Output Delivery System (ODS) to redirect the output for the DESCRIBE FUNCTION statement to SAS data sets.

The output data sets WORK.RFC_PARM, WORK.RFC_ATTR and WORK.RFC_COLS are created with the metadata information about function RFC_SYSTEM_INFO.

Program

```
ods listing close;
ods output parameter=rfc_parm attributes=rfc_attr column=rfc_cols;
proc callrfc user='USER' passwd='PASS' client='800'
  lang='EN' ashost='sapr3srv.sup.com' sysnr='03';

  describe function 'RFC_SYSTEM_INFO';
run;
ods output close;
```

APPENDIX

1

Recommended Reading

Recommended Reading 73

Recommended Reading

Here is the recommended reading list for this title:

- *Base SAS Procedures Guide*
- *SAS Language Reference: Concepts*
- *SAS Language Reference: Dictionary*
- *SAS/Warehouse Administrator Metadata API Reference*
- *Step-by-Step Programming with Base SAS Software*
- SAS® Companion that is specific to your operating environment.

For a complete list of SAS publications, go to support.sas.com/bookstore. If you have questions about which titles you need, please contact a SAS Publishing Sales Representative at:

SAS Publishing Sales
SAS Campus Drive
Cary, NC 27513
Telephone: 1-800-727-3228
Fax: 1-919-531-9439
E-mail: sasbook@sas.com
Web address: support.sas.com/bookstore

Customers outside the United States and Canada, please contact your local SAS office for assistance.

Glossary

ABAP

See advanced business applications programming.

advanced business applications programming

A high-level language that is used for programming the SAP Web Application Server, part of its NetWeaver platform for building business applications. Its syntax is somewhat similar to COBOL. Short form: ABAP.

BAdI

See Business Add-Ins.

Business Add-Ins

Business Add-Ins are enhancements to the standard version of the system. They can be inserted into the SAP System to accommodate user requirements too specific to be included in the standard delivery. Since specific industries often require special functions, SAP allows you to predefine these points in your software.

Communications Interface

an interface through which various types of R/3 communication can occur. R/3 communicates with all systems that comply with Open Systems architecture. The Communications Interface enables users to work transparently with data from any one of several database management systems in any one of several operating environments. The Communications Interface also enables users to run R/3 in multiple operating environments.

CPI-C

Common Programming Interface-Communications. An application-level interface for direct program-to-program communication.

data dictionary

in R/3 applications, a central repository that stores metadata about the data sources, such as a database that contains information about company expenses and company profits or a database that contains human resources information. The data dictionary also stores information about the applications that use the data sources.

data model

a collection of data that is relevant to a particular facet of a specific business, such as finance or materials management. A data model can contain other data models. A data model is used to organize potentially complex relationships between entity types and other data models.

Database Interface

in SAP R/3, a logical application layer that is used to transport data between one or more physical databases and the R/3 applications. The R/3 Database Interface translates R/3 SQL statements into pure SQL statements and enables R/3 to access data that is in other database formats.

Enterprise Resource Planning system

an integrated application that controls day-to-day business operations such as inventory, sales, finance, human resources, and distribution. From the warehousing perspective, ERP systems differ from standard databases in that they have predefined data models that must be understood in order to successfully extract the data. Short form: ERP system.

entity

in SAP R/3 applications, an abstract or physical object, such as "Mr. Brown" or "Division Management Subsystem," which represents the smallest collection of data that has meaning within a business. An entity is categorized according to its entity type properties. See also entity type.

entity type

in R/3 applications, a category of entities that have the same definable characteristics. The definable characteristics are called attributes. Multiple entity types can be grouped into a single table. See also entity.

ERP system

See Enterprise Resource Planning system.

R/3

an enterprise resource planning (ERP) system that has been developed by SAP for use in client/server environments.

RFC

Remote Function Call. The RFC is the SAP implementation of a remote procedure call. RFCs enable external applications such as SAS and R/3 systems to access SAP BW and R/3 systems.

SAS data set

a term that describes what is typically referred to as a table in database management systems. There are two types of SAS data sets: SAS data files and SAS data views. SAS data files contain data values in addition to descriptor information that is associated with the data. SAS data views contain only the descriptor information plus other information that is required for retrieving data values from other SAS data sets or from files whose contents are in other software vendors' file formats.

sideinfo file

a text file that can be used to set logon parameters such as RFC host, system number, gateway service, and gateway host, for R/3 users.

variable

a column in a SAS data set or in a SAS data view. The data values for each variable describe a single characteristic for all observations.

view

a generic term (used by many software vendors) for a definition of a virtual data set (or table). The definition is named and stored for later use. A view contains no data; it merely describes or defines data that is stored elsewhere. See also SAS data view.

Index

A

- ABAP Function module 10
 - name specification 39
- ABAP functions and programs
 - namespace for 39
- ABAP program name
 - RFC server 39
- ABAP Report module 10
- ABAP type NUMC
 - SAS type for 46
- ABAPFM= option
 - LIBNAME statement for SAP 39
 - PROC CALLRFC statement 63
- ABAP_NAMESPACE= option
 - LIBNAME statement for SAP 39
- ABAPPROG= option
 - LIBNAME statement for SAP 39
 - PROC CALLRFC statement 63
- accessibility features 8
- ASHOST= option
 - LIBNAME statement for SAP 40
 - PROC CALLRFC statement 64
- AUTHDOMAIN= option
 - LIBNAME statement for SAP 40
- authentication domain
 - name of 40
- authorization checks
 - for BW and BI objects 13

B

- BAPI connection parameters 32
- BATCH= data set option 55
- batch jobs, for data extracts 41, 55
- batch mode, RFC server in 19
- batch operations 33
 - passwords 34
 - source parameter 34
- BATCH= option
 - LIBNAME statement for SAP 41
- BI objects
 - authorization checks for 13
- buffer size
 - minimum for data transfers 41
- BUFFER_SIZE= option
 - LIBNAME statement for SAP 41
- BW objects
 - authorization checks for 13

C

- CALL FUNCTION statement
 - CALLRFC procedure 66
- CALLBACK option
 - CALL FUNCTION statement (RFC) 67
- %CALLRFC macro 22
- CALLRFC procedure 61, 62
 - displaying RFC_SYSTEM_INFO 70
 - ODS with DESCRIBE FUNCTION statement 71
 - syntax 62
 - task tables 62
- CLIENT= option
 - LIBNAME statement for SAP 41
 - PROC CALLRFC statement 64
- code page 44
- column information 69
- connecting to R/3 systems 27
- CONNECTION= LIBNAME option 50
- connection parameters 30
- CONNECTION_GROUP= LIBNAME option 51

D

- data access methods
 - direct access 9
 - extract management method 9, 11
 - ODBC 9
 - R/3 9
 - Remote Function Call (RFC) 10
 - secure network communications 12
- data extracts
 - batch mode 41, 55
- data set options
 - SAP engine 54
- data transfers
 - minimum buffer size for 41
- DBCONDITION= data set option 55
- DBGEN_NAME= data set option 56
- DBGEN_NAME= LIBNAME option 51
- DBKEY= data set option 56
- DBMASTER= data set option 56
- DBMS columns
 - labels 52
 - renaming 51, 56
- DBMS connections 50, 51
 - prompts for 51
 - timing of 52
- DBMS data
 - subsetting and ordering 55

DBPROMPT= LIBNAME option 51
 DBSASLABEL= LIBNAME option 52
 DEFER= LIBNAME option 52
 DESCRIBE FUNCTION statement
 CALLRFC procedure 67, 71
 DESTGROUP= option
 LIBNAME statement for SAP 42
 destination groups
 name of 42
 DESTINATION= option
 LIBNAME statement for SAP 42
 PROC CALLRFC statement 64
 direct access method 9
 DIRECT_SQL= LIBNAME option 53
 disconnecting from R/3 systems 29

E

exporting metadata
 to SAS Warehouse Administrator 7
 EXPORTING option
 CALL FUNCTION statement (RFC) 66
 extract management method 9, 11
 data transfer and use 11
 installation and setup 11

F

floating point numbers 44
 functions
 calling R/3 function modules 22
 name of 68

G

generated SQL
 passing to DBMS 53
 GROUP= option
 LIBNAME statement for SAP 42
 PROC CALLRFC statement 64
 GWHOST= option
 LIBNAME statement for SAP 43
 PROC CALLRFC statement 64
 GWSERV= option
 LIBNAME statement for SAP 43
 PROC CALLRFC statement 64

H

Help 8
 HOST= option
 LIBNAME statement for SAP 43
 PROC CALLRFC statement 64

I

IEEE_REVERSE= option
 LIBNAME statement for SAP 44
 PROC CALLRFC statement 64
 IMPORTING option
 CALL FUNCTION statement (RFC) 67
 INENCODING= option
 LIBNAME statement for SAP 44
 INTABLES option
 CALL FUNCTION statement (RFC) 67

J

joins
 number of tables 45
 performance and 53, 56
 table size 56

L

LANGUAGE= option
 LIBNAME statement for SAP 44
 PROC CALLRFC statement 64
 LIBNAME engine
 mode for 45
 LIBNAME statement for SAP 37
 engine options 38
 examples 58
 LIBNAME options 50
 syntax 37
 LIBREF= option
 PROC CALLRFC statement 64
 logging on 3
 logon parameter client 41

M

macro variables 21
 RFC_LOGON_INFO 30
 macros 21
 %CALLRFC 22
 %R3CONNB 26
 %R3CONN 27
 %R3CONNE 29
 MAX_TABLE_JOINS= option
 LIBNAME statement for SAP 45
 message server
 host name 45
 metadata 67
 exporting to SAS Warehouse Administrator 7
 searching 4
 MHOST= option
 LIBNAME statement for SAP 45
 missing values 57
 MODE= option
 LIBNAME statement for SAP 45
 PROC CALLRFC statement 65
 MSHOST= option
 PROC CALLRFC statement 65
 MULTI_DATASRC_OPT= LIBNAME option 53

N

namespace
 for ABAP functions and programs 39
 NULLCHAR= data set option 57
 NULLCHARVAL= data set option 57
 NUMC_SAS_TYPE= option
 LIBNAME statement for SAP 46
 PROC CALLRFC statement 65

O

ODBC (Open Data Base Connectivity) 9
 ODS
 with DESCRIBE FUNCTION statement (CALLRFC) 71
 ordering DBMS data 55

P

parameter information 68
 PASSWORD= option
 LIBNAME statement for SAP 46
 PROC CALLRFC statement 65
 passwords
 for batch operations 34
 SAP logon parameter 46
 SAP logon parameter, encrypted form 47
 PASSWORDX= option
 LIBNAME statement for SAP 47
 PROC CALLRFC statement 65
 PORT= option
 LIBNAME statement for SAP 47
 PROC CALLRFC statement 65
 PROC CALLRFC statement 62
 profiles
 connecting or disconnecting with 26
 creating 3
 predefined 3

R

R/3 1
 accessibility features 8
 accessing data with 9
 connecting to R/3 systems 27
 disconnecting from R/3 systems 29
 logging on 3
 starting 2
 system name 47
 system number 49
 R/3 BAPI Connector: Logon window 32
 %R3CONN macro 26
 %R3CONN macro 27
 %R3CONN macro 29
 R3NAME= option
 LIBNAME statement for SAP 47
 PROC CALLRFC statement 65
 random access engine
 SAS/ACCESS engine as 54
 read transactions
 utility spool file 54
 Remote Function Call (RFC) 10
 REREAD_EXPOSURE= data set option 58
 REREAD_EXPOSURE= LIBNAME option 54
 RFC calls 66
 RFC metadata 67
 RFC (Remote Function Call) 10
 RFC server 15
 ABAP Function module name 39
 ABAP program name 39
 batch jobs for data extracts 55
 batch mode 19
 host for connecting to SAP 43
 macros and macro variables 21
 port number 47
 R/3 BAPI Connector: Logon window 32
 tracing requests 49
 UNIX 15
 Windows 17
 RFC_LOGON_INFO macro variable 30
 RfcOpenEx() call
 additional logon or connection parameters 48

RFC_STRING= option 12
 LIBNAME statement for SAP 48
 PROC CALLRFC statement 65

S

SAP application servers
 host name or IP address 40
 name of the group of 42
 SAP destination
 defining from SAP logon utility 58
 defining from saprfc.ini file 58
 SAP engine
 data set options 54
 SAP gateway
 host name of 43
 service of 43
 SAP LIBNAME engine 38
 mode for 45
 SAP logon parameter
 language for 44
 password 46
 password, encrypted form 47
 user name 49
 SAP logon parameter client 41
 SAP logon utility
 defining SAP destination from 58
 SAP server
 accessing from local host 58
 SAPLOGON
 string defined for 48
 SAPLOGON_ID= option
 LIBNAME statement for SAP 48
 PROC CALLRFC statement 65
 saprfc.ini file 12
 defining SAP destination from 58
 destination indication 42
 SAS/ACCESS engine
 as random access engine 54, 58
 SAS Warehouse Administrator
 exporting metadata to 7
 searching metadata 4
 secure network communications (SNC) 12
 SHORT option
 DESCRIBE FUNCTION statement (CALLRFC) 67
 source parameter 34
 SPOOL= LIBNAME option 54
 subsetting DBMS data 55
 SYSNR= option
 LIBNAME statement for SAP 49
 PROC CALLRFC statement 65

T

TABLES option
 CALL FUNCTION statement (RFC) 67
 TRACE= option
 LIBNAME statement for SAP 49
 PROC CALLRFC statement 66
 tutorial 1

U

UNIX
 RFC server on 15
 user logon language 58

USER= option
LIBNAME statement for SAP 49
PROC CALLRFC statement 66
utility spool file
for read transactions 54

W

Windows
RFC server on 17

Your Turn

We welcome your feedback.

- If you have comments about this book, please send them to **yourturn@sas.com**. Include the full title and page numbers (if applicable).
- If you have comments about the software, please send them to **suggest@sas.com**.

