

## APPENDIX

## 6

## SAS/SHARE Version 6 to Version 7 and Version 8 Cross-Version Issues

<i>Limitations on Cross-Version Data Access</i>	213
<i>Audience</i>	213
<i>Cross-Conversion Compatibility</i>	214
<i>Backward and Forward Compatibility</i>	214
<i>Mixing Cross-Version Data Files in a Single Library</i>	216
<i>Explicitly Converting Data from Version 6 to Version 7 or Version 8 Format</i>	217
<i>Accessing Members in Restricted Mode</i>	217
<i>Accessing Data Files</i>	217
<i>Accessing DATA Step, PROC SQL, and SAS/ACCESS Views</i>	217
<i>Accessing Catalogs</i>	218

### Limitations on Cross-Version Data Access

There are no limitations on migrating from SAS/SHARE Version 7 to Version 8 because they are virtually identical. However, when migrating from SAS/SHARE Version 6 to Version 7 or Version 8, your overriding concern is probably whether you can continue to access your Version 6 data files and SAS software applications on a SAS/SHARE server.

You can continue to read Version 6 data in a Version 7 or Version 8 cross-version environment. Version 7 and Version 8 SAS applications have virtually unrestricted access to Version 6 SAS data files. Version 7 and Version 8 SAS may read, update, and write Version 6 data files with one exception: Version 7 or Version 8 SAS applications cannot update or write new Version 7 or Version 8 SAS I/O features to Version 6 data files. Version 7 and Version 8 SAS applications have read-only access to other Version 6 SAS member types.

Information in this appendix relates only to SAS/SHARE cross-version issues. For information about the SAS/CONNECT cross-version issues, see the *SAS/CONNECT User's Guide*. For information about cross-release issues that relate to single user SAS mode, see *SAS Language Reference: Concepts*. For complete details about moving and accessing SAS files across operating environments, see *Moving and Accessing SAS Files across Operating Environments*.

### Audience

The information in this appendix is for the person who is responsible for migrating the SAS Version 7 or Version 8 software into the SAS/SHARE client/server computing environment. This person also makes decisions about converting file formats from

Version 6 to Version 7 or Version 8, modifies client applications, and decides which version of SAS to use on client and server sides.

---

## Cross-Conversion Compatibility

The primary obstacles that limit compatibility between versions are the changes to the data file format and how applications exploit the new format. Version 7 and Version 8 offers several new features that make Version 7 and Version 8 data files incompatible with Version 6 data files. Some of the base SAS Software Version 7 and Version 8 features, which are supported by SAS/SHARE, that affect the data file format include:

- long names
- long labels
- data set generations
- integrity constraints.

Cross-version compatibility enables cross-version access that has some restrictions. Version 6 SAS sessions may be able to access Version 7 or Version 8 SAS data files if those files do not contain new Version 7 or Version 8 features.

For example, a Version 6 client can read, write, and update a Version 7 or Version 8 data file on a Version 7 or Version 8 server that has not implemented Version 7 or Version 8 features such as long variable names. Likewise, a Version 7 or Version 8 client can read, write and update a Version 6 data file on a Version 6 server if the client application adheres to the restricted mode (does not implement Version 7 or Version 8 features).

For complete information about Version 7 and Version 8 features, see *SAS Language Reference: Concepts*.

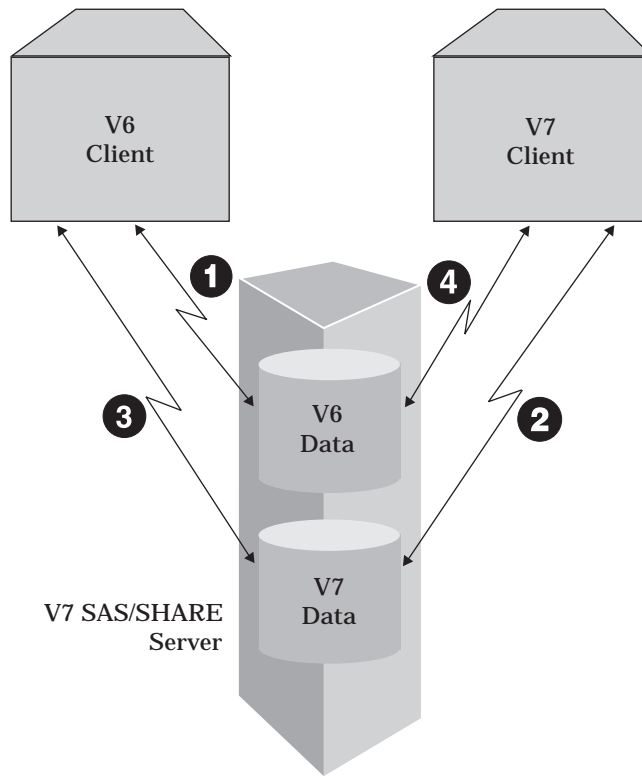
---

## Backward and Forward Compatibility

Figure A6.1 on page 215 shows a client/server computing environment that contains a Version 7 server.

*Note:* Because Version 7 and Version 8 features are virtually identical, references to Version 7 in the figure also apply to Version 8.  $\triangle$

The server contains both Version 6 data and Version 7 or Version 8 data that is accessed by a Version 6 client and a Version 7 or Version 8 client.

**Figure A6.1** Forward and Backward Compatibility in a Cross-Release Environment

- ❶ A Version 6 client can read, write, and update a Version 6 data file.
- ❷ A Version 7 or Version 8 client can read, write, and update a Version 7 or Version 8 data file.
- ❸ A Version 6 client can read, write, and update a Version 7 or Version 8 data file that has not implemented Version 7 or Version 8 features such as long variable names. In this instance, a Version 6 client and its application are *forward compatible* with a Version 7 or Version 8 server and its data because the data does not contain Version 7 or Version 8 features. However, a Version 6 client and application are not forward compatible with a Version 7 or Version 8 server and data that does include Version 7 or Version 8 features.
- ❹ A Version 7 or Version 8 client can read, write, and update a Version 6 data file as long as the application does not implement Version 7 or Version 8 features such as long variable names. In this instance, a Version 7 or Version 8 client and application are *backward compatible* with Version 6 data if the application does not exploit Version 7 or Version 8 features. If the program does exploit Version 7 or Version 8 features, the program's attempt to open a Version 6 data set fails.

*Note:* You can access Version 6 and Version 7 or Version 8 data in a mixed-mode library if you intentionally update or write the data in the correct format. See “Mixing Cross-Version Data Files in a Single Library” on page 216 for more information about accessing data files in a mixed-mode library. △

Backward and forward compatibility allows cross-version access in a *restricted mode*. This means that Version 6 clients can operate on Version 7 or Version 8 data that does not contain Version 7 or Version 8 features, and Version 7 or Version 8 applications that do not exploit Version 7 or Version 8 features can operate on Version 6 data.

## Mixing Cross-Version Data Files in a Single Library

Version 7 or Version 8 SAS contains a compatibility engine for accessing Version 6 data files. If a physical library contains only Version 6 files, Version 7 or Version 8 uses the Version 6 engine to access the SAS files in that library. Otherwise, the Version 7 or Version 8 engine is used.

*Note:* The V7 and V8 engines are interchangeable.  $\Delta$

Version 6 and Version 7 or Version 8 data files can reside in the same library. Such a library is called *mixed mode*. For example, a mixed-mode library is created when a Version 7 or Version 8 client updates a data file entry in or adds a data file entry to a Version 6 library.

You may access data files in mixed-mode libraries by

- specifying the library name in the LIBNAME statement and letting SAS select the correct engine to invoke.

In this example, SAS analyzes the filename extensions of the files that are contained in library **v6lib** to determine what type of engine to invoke.

```
LIBNAME v6lib 'host-path';
```

If the library contains files that are in Version 6 format, SAS invokes the Version 6 engine automatically.

- explicitly specifying the RENGINE= option in the LIBNAME statement.

This example shows how to access a specific SAS data library explicitly by specifying the appropriate engine in the RENGINE= option in the LIBNAME statement.

```
LIBNAME mylib 'host-path' server=shr1 RENGINE=V6;
```

To access a Version 6 file, specify RENGINE=V6; to access a Version 7 or Version 8 file, specify RENGINE=V7 or RENGINE=V8.

*Note:* The V7 and V8 engines are interchangeable.  $\Delta$

If you do not explicitly specify the RENGINE= option, then the engine that is appropriate to the files in the library is invoked. For example, in a Version 7 session, the Version 7 engine is invoked by default.

- concatenating a Version 6 library with a Version 7 or Version 8 library to form a mixed library.

In this example, the third LIBNAME statement concatenates the Version 7 or Version 8 library with the Version 6 library to produce a library that contains data files in both Version 6 and Version 7 or Version 8 format.

*Note:* Version 7 and Version 8 formats are identical.  $\Delta$

```
LIBNAME v6lib 'host-path' SERVER=shr1 RENGINE=V6';
LIBNAME v8lib 'host-path' SERVER=shr1 RENGINE=V8;
LIBNAME mylib (v8lib v6lib);
```

The order in which you specify the librefs within parenthesis in the LIBNAME statement determines what engine is invoked to access the wanted files. For example, specifying **v8lib** first invokes the Version 8 engine, which only accesses files that are appended with Version 8 filename extensions. The Version 6 files are ignored. Likewise, specifying **v6lib** first invokes the Version 6 engine, which only accesses files that are appended with Version 6 filename extensions. In this case, Version 8 files are ignored.

*Note:* Version 7 and Version 8 filename extensions are identical.  $\Delta$

---

## Explicitly Converting Data from Version 6 to Version 7 or Version 8 Format

Rather than limiting your access to Version 6 formatted files or Version 7 or Version 8 formatted files, you can convert all your data from one format to another. Conversion procedures are based on several factors, including what your host platform is and the SAS member that is being converted. See *Moving and Accessing SAS Files across Operating Environments* for details about strategies for converting data from Version 6 to Version 7 or Version 8 format.

---

## Accessing Members in Restricted Mode

If you do not convert data files from Version 6 to Version 7 or Version 8 format, you can continue to access members in restricted mode. This section discusses these member types:

- data files
- DATA step, PROC SQL, and SAS/ACCESS views
- catalogs.

---

### Accessing Data Files

Table A6.1 on page 217 summarizes the limitations to accessing data files in a cross-version environment.

**Full support** means that reading, writing, and updating are allowed.

**No access** means that no reading, writing, or updating is allowed.

**Table A6.1** Version 6-Version 7 or Version 8 Data File Cross-Version Support

	V6 Server		V7 or V8 Server	
	V6 Data Set	V7 or V8 Data Set	V6 Data Set	V7 or V8 Data Set
V6 Client	Full support	No access	Full support	Full support*
V7 or V8 Client	Full support*	No access	Full support	Full support

\* indicates that features that are specific to Version 7 or Version 8, such as long names, are not supported.

---

### Accessing DATA Step, PROC SQL, and SAS/ACCESS Views

Table A6.2 on page 218 summarizes the limitations on access to DATA step and PROC SQL views in a cross-version environment.

**Full support** means that reading, writing, and updating are allowed.

**No access** means that no reading, writing, or updating is allowed.

R indicates that a Version 7 or Version 8 client can read a Version 6 DATA step view or an SQL view.

**Table A6.2** Version 6-Version 7 or Version 8 Data Set and PROC SQL View Cross-Version Support

	V6 Server		V7 or V8 Server	
	V6 View	V7 or V8 View	V6 View	V7 or V8 View
V6 Client	Full support	No access	Full support*	R**
V7 or V8 Client	R**	No access	R***	Full support

\* At either the client or the server, the RMTVIEW=NO option and argument to the LIBNAME statement has been set.

\*\* At either the client or the server, the RMTVIEW=YES option and argument to the LIBNAME statement has been set.

\*\*\*Applies to PROC SQL views only.

Table A6.3 on page 218 summarizes the limitations to accessing SAS/ACCESS views in a cross-version environment.

**Full support** means that reading, writing, and updating are allowed.

**No access** means that no reading, writing, or updating is allowed.

**Table A6.3** Version 6-Version 7 or Version 8 SAS/ACCESS View Cross-Version Support

	V6 Server		V7 or V8 Server	
	V6 View	V7 or V8 View	V6 View	V7 or V8 View
V6 Client	Full support	No access	Full support	Full support
V7 or V8 Client	Full support	No access	Full support	Full support

## Accessing Catalogs

Table A6.4 on page 218 summarizes the limitations on access to catalogs in a cross-version environment.

**Full support** means that reading, writing, and updating are allowed.

**No access** means that no reading, writing, or updating is allowed.

**Table A6.4** Version 6-Version 7 or Version 8 Catalog Cross-Version Support

	V6 Server		V7 or V8 Server	
	V6 Catalog	V7 or V8 Catalog	V6 Catalog	V7 or V8 Catalog
V6 Client	Full support	No access	R	Full support*
V7 or V8 Client	RW**	No access	RW**	Full support

\* This UPDATE operation produces a Version 7 or Version 8 catalog that contains Version 6 data.

\*\* This WRITE operation allows only a limited copy. You can use PROC COPY to copy a Version 6 catalog to another Version 7 or Version 8 library.

# Glossary

---

**access descriptor**

a SAS/ACCESS file that describes data to the SAS System that is managed by an external software vendor's interface product. You use an access descriptor as a master to create view descriptors. See also view and view descriptor.

**batch mode**

a method of executing SAS programs in which you prepare a file that contains SAS statements and any necessary operating environment commands, and submit the program to the computer's batch queue. While the program executes, control of the SAS System returns to the user. Batch mode is sometimes referred to as running-in-background. The job output can be written to files or printed on an output device.

**browsing data**

the process of viewing the observations in a file. Depending on how the file is accessed, observations may be viewed one at a time or as a group (in a tabular format). You cannot update data that you are browsing.

**client**

the computer on which you use a SAS session to access a SAS/SHARE server. See also SAS/SHARE server.

**communications access method**

the method that your local session uses to communicate with a remote host. Supported methods include NetBIOS, TCP/IP, TELNET, and APPC. Values for the communications access method are specified by using the COMAMID= system option.

**concurrency**

the shared use of resources by multiple clients or applications at the same time.

**control level**

one of the determinants in the type of lock that a task obtains on a SAS data set or on an observation in the data set. For a SAS data set, control level specifies how other SAS tasks should be allowed to access the SAS data set concurrently. Every SAS task has an open mode and a default control level for each SAS data set that it accesses, based on how the task will operate on that data set. See also locking and open mode.

**DATA step view**

a DATA step program that generates a SAS data view. Like other SAS data views, a DATA step view contains a definition of data that is stored elsewhere; it does not

contain the physical data. DATA step views can only function as input data sets in Release 6.07.

An input DATA step view is generated from a DATA step program. The view's input data can come from one or more sources, including external files and other SAS data sets. Because a DATA step view only reads (opens for input) other files, you cannot update this view's underlying data.

**data value**

a unit of character or numeric value information in a SAS data set. A data value represents one variable in one observation.

**database management system (DBMS)**

an integrated software package that enables you to create and manipulate data in the form of databases. See also relational database management systems.

**display manager mode**

an interactive method of running SAS programs in which you edit a group of statements, submit the statements, and then review the results of the statements in various windows.

**editing data**

the process of viewing a file and having the ability to change its data. You might see the data one observation at a time or in a tabular format.

**engine**

a part of the SAS System that reads from or writes to a file. Each engine allows the SAS System to access files that have a particular format. There are several types of engines. See also interface view engine and REMOTE engine.

**execution**

① the process in the DATA step in which the SAS System carries out statements for each observation or record in the file.

② in contexts other than the DATA step, such as SAS macros, procedures, and global statements, the process in which the SAS System performs the actions indicated.

**external file**

① a file maintained by the host operating environment rather than by the SAS System. External files can contain raw data, SAS programming statements, procedure output, or output created by the PUT statement.

② in a DATA step, a file that the SAS System can read by using INFILE and INPUT statements, or a file that the SAS System can write by using FILE and PUT statements.

**file**

a collection of related records treated as a unit. SAS files are processed and controlled through the SAS System and are stored in a SAS data library.

**format**

an instruction that the SAS System uses to display or to write each value of a variable. Some formats are supplied by SAS software. Other formats can be written by the user by using the FORMAT procedure in base SAS software or by using SAS/TOOLKIT software.

**host**

the operating environment that provides facilities, computer services, and an environment for software applications.

**index**

a part of a SAS data file that stores both the values of a variable (in a SAS data file) and a set of directions that enable SAS, under certain circumstances, to locate



observations in a SAS data file more quickly and efficiently. Indexing variables usually makes data set processing faster, although SAS determines the most efficient way to process data that is maintained by the SAS System.

**informat**

an instruction that the SAS System uses to read raw data values to create variable values. Some informats are supplied by SAS software. Other informats can be written by the user by using the FORMAT procedure in base SAS software or by using SAS/TOOLKIT software.

**interactive line mode**

a method of running SAS programs in which you enter one line of a SAS program at a time at the SAS session prompt. SAS processes each line immediately after you press the ENTER or the RETURN key. Procedure output and informative messages are returned directly to the display monitor.

**interface view engine**

a SAS system engine that retrieves data directly from files formatted by other software vendors and presents the data to the SAS System in the form of a SAS data set. Interface view engines are transparent to the user and are not specified in the LIBNAME statement.

**library engine**

an engine that accesses groups of files and puts them into the correct form for processing by SAS utility windows and procedures. A library engine also determines the fundamental processing characteristics of the library, presents lists of files for the library directory, and supports view engines. See also engine and REMOTE engine.

**libref**

- ❶ the name temporarily associated with a SAS data library. You assign a libref by using a LIBNAME statement or operating system control language.
- ❷ the first part of a multilevel SAS file name that indicates the SAS data library in which the file is stored. For example, in the name SASUSER.ACCOUNTS, the name SASUSER is the libref.

**local host**

the computer on which you use a SAS session to initiate a link with a remote host. See also remote host.

**local session**

a SAS session that runs on the local host. The local session accepts SAS statements and passes those statements that are remote-submitted to the remote host for processing. The local session manages the output and the messages from both the local and the remote sessions. See also remote host and remote session.

**locking**

a technique for detecting conflicts among the requests from different SAS tasks. A task obtains a lock on a member (SAS data set) or record (observations) based on the open mode and control level for that SAS data set. A lock can also be obtained on a SAS data library, a data set, a catalog, or a catalog entry by using the LOCK statement or the LOCK command. See also control level and open mode.

**member**

a file in a SAS data library.

**member name**

a name given to a file in a SAS data library. A member name can reference a SAS data set, a catalog, an access descriptor, or a stored program.

**member type**

a name assigned by SAS software that identifies the type of information that is stored in a SAS file. Member types include ACCESS, DATA, CATALOG, PROGRAM, and VIEW.

**missing value**

a value in SAS System that indicates that no data for the variable is stored in the current observation. By default, SAS prints a missing numeric value as a single period and a missing character value as a blank space.

**noninteractive mode**

a method of running SAS programs in which you prepare a file of SAS statements and submit the program to the computer system. The program runs immediately and occupies your current session.

**observation**

a row in a SAS data set. An observation is a collection of data values that are associated with a single entity, such as a customer or a state. Each observation contains one data value for each variable in the data set.

**open mode**

a method by which a SAS task accesses and operates on a member in a SAS data library. There are three open modes for SAS files: input, update, and output. See also control level and locking.

**PROC SQL view**

a definition of a virtual data set that is named and stored for later use. This file contains no data, but it defines data that is stored in the PROC SQL view's underlying SAS data files, or is described by SAS/ACCESS views, DATA step views, or other PROC SQL views. Its output table can be a subset or a superset of one or multiple underlying structures. However, in Release 6.06, you cannot reference a PROC SQL view to update its underlying data. See also view.

**quiescing**

a method for stopping an active resource, that is, a SAS/SHARE server, server library, or users of a SAS/SHARE server. A PROC OPERATE QUIESCE command terminates a resource gradually by disallowing new requests to use the resource.

**relational database management system**

a database management system that organizes and accesses data according to relationships between data items.

**REMOTE engine**

a SAS library engine for SAS/SHARE software. Using it enables a client SAS session to access shared data by communicating with a SAS/SHARE server. See also SAS/SHARE server.

**remote host**

the computer on which processing occurs when you execute a PROC DOWNLOAD, a PROC UPLOAD, or other SAS statement that is executed by using the RSUBMIT command or the RSUBMIT statement. The term remote describes how you interact with a SAS session running on the computer; it is not related to the physical location of the computer. See also local host.

**remote session**

a SAS session running in a special mode on the remote host. No output or log messages are displayed on the remote host; instead, the results of a remote SAS session are transmitted back to the log and output files on the local host. See also local host.

**SAS/ACCESS views**

See SAS data view and view descriptor.

**SAS data file**

a SAS data set that contains both the data values and the descriptor information. SAS data files are of member type DATA.

**SAS data library**

a collection of one or more SAS files that are recognized by the SAS System. Each file is a member of the library.

**SAS data set**

a collection of information that is stored as a unit in SAS software. A SAS data set is arranged in a rectangular, two-dimensional format. Each item in a SAS data set is called a data value. Data values in a row comprise an observation. Data values in a column comprise a variable. See also SAS data file and SAS data view.

**SAS data view**

a SAS data set in which the descriptor information and the observations are obtained from other files. SAS data views store only the information that is required to retrieve data values or descriptor information. SAS data views are of member type VIEW.

**SAS Display Manager System**

an interactive windowing environment in which actions are performed by issuing commands. Display manager commands can be issued by typing them on the command line, pressing function keys, or selecting items from pull-down menus. Within one session, multiple tasks can be accomplished. Display manager can be used to prepare and submit programs, view and print the results, and debug and resubmit the programs. See also display manager mode.

**SAS/SHARE server**

an execution of the SERVER procedure. The SERVER procedure is part of SAS/SHARE software. A server runs in a separate SAS execution that services client's SAS sessions by controlling and executing input and output requests to one or more SAS data libraries.

**SAS session**

an environment created by invoking the SAS System in which you can give commands, submit SAS statements, receive responses to the commands, and receive results of the SAS statements until you exit the environment or until the environment is terminated.

**server library**

a SAS data library that is defined to a SAS/SHARE server. The server controls access to the library.

**Structured Query Language (SQL)**

the standardized, high-level query language that is used in relational database management systems to create and manipulate database management system objects. The SAS System implements SQL by means of the SQL procedure.

**variable**

a column in a SAS data set. A variable is a set of data values that describes a given characteristic across all observations.

**view**

a definition of a virtual data set that is named and stored for later use. This file contains no data but describes or defines data that is stored elsewhere. See also DATA step view, PROC SQL views, SAS data view, and view descriptor.

**view descriptor**

a SAS/ACCESS file that defines all or a subset of interface product data that is described by an access descriptor. The access descriptor describes the data in a single product table or view, when views are allowed in the product. See also access descriptor.

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

**SAS/SHARE User's Guide, Version 8**

Copyright © 1999 by SAS Institute Inc., Cary, NC, USA.

ISBN 1-58025-478-0

All rights reserved. Printed in the United States of America. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, or otherwise, without the prior written permission of the publisher, SAS Institute Inc.

**U.S. Government Restricted Rights Notice.** Use, duplication, or disclosure of the software by the government is subject to restrictions as set forth in FAR 52.227-19 Commercial Computer Software-Restricted Rights (June 1987).

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

1st printing, September 1999

SAS<sup>®</sup> 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. <sup>®</sup> indicates USA registration.

IBM<sup>®</sup>, AIX<sup>®</sup>, DB2<sup>®</sup>, OS/2<sup>®</sup>, OS/390<sup>®</sup>, RMT<sup>™</sup>, RS/6000<sup>®</sup>, System/370<sup>™</sup>, and System/390<sup>®</sup> are registered trademarks or trademarks of International Business Machines Corporation. ORACLE<sup>®</sup> is a registered trademark or trademark of Oracle Corporation. <sup>®</sup> indicates USA registration.

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

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