CHAPTER 8

The CIMPORT Procedure

Overview

The CIMPORT procedure *imports* a transport file that was created (*exported*) by the CPORT procedure. PROC CIMPORT restores the transport file to its original form as a SAS catalog, SAS data set, or SAS data library. *Transport files* are sequential files that each contain a SAS data library, a SAS catalog, or a SAS data set in transport format. The transport format that PROC CPORT writes is the same for all environments and for many releases of the SAS System.

PROC CIMPORT can read only transport files that PROC CPORT creates. For information on the transport files that the transport engine creates, see the section on SAS files in *SAS Language Reference: Concepts*.

PROC CIMPORT also *converts* SAS files, which means that it changes the format of a SAS file from the format appropriate for one version of the SAS System to the format appropriate for another version. For example, you can use PROC CPORT and PROC CIMPORT to move files from earlier releases of the SAS System to more recent releases. In such cases, PROC CIMPORT automatically converts the contents of the transport file as it imports it.

PROC CIMPORT produces no output, but it does write notes to the SAS log.

To export and import files, follow these steps:

1. Use PROC CPORT to export the SAS files that you want to transport.
2. If you are changing operating environments, move the transport file to the new machine by using either communications software or a magnetic medium.

   Note: If you use communications software to move the transport file, be sure that it treats the transport file as a *binary* file and that it modifies neither the attributes nor the contents of the file. △

3. Use PROC CIMPORT to translate the transport file into the format appropriate for the new operating environnment or release.

Procedure Syntax

PROC CIMPORT *destination=libref* | <*libref.*>*member-name* <*option(s)*>;
  EXCLUDE *SAS file(s)* | *catalog entry(s)*</ MEMTYPE=*mtype*></
    ENTRYTYPE=*entry-type*>;
  SELECT *SAS file(s)* | *catalog entry(s)*</ MEMTYPE=*mtype*></
    ENTRYTYPE=*entry-type*>;

PROC CIMPORT Statement

PROC CIMPORT *destination=libref* | <*libref.*> *member-name*<*option(s)*>;

To do this	Use this option
Identify the input transport file	
Specify a previously defined fileref or the filename of the transport file to read	INFILE=
Read the input transport file from a tape	TAPE
Select files to import	
Exclude specified entry types from the import process	EET=
Specify entry types to import	ET=
Control the contents of the transport file	
Specify whether to extend by 1 byte the length of short numerics (less than 8 bytes) when you import them	EXTENDSN=
Specify that only data sets, only catalogs, or both, be moved when a library is imported	MEMTYPE=
Enable access to a locked catalog	FORCE
Create a new catalog for the imported transport file, and delete any existing catalog with the same name	NEW
Import SAS/AF PROGRAM and SCL entries without edit capability	NOEDIT
Suppress the importing of source code for SAS/AF entries that contain compiled SCL code	NOSRC

Required Arguments

destination=libref* | <*libref.* >*member-name
identifies the type of file to import and specifies the specific catalog, SAS data set, or SAS data library to import.

destination
identifies the file or files in the transport file as a single catalog, as a single SAS data set, or as the members of a SAS data library. The *destination* argument can be one of the following:

CATALOG | CAT | C

DATA | DS | D

LIBRARY | LIB | L

libref | *<libref. > member-name*
specifies the specific catalog, SAS data set, or SAS data library as the destination of the transport file. If the *destination* argument is CATALOG or DATA, you can specify both a *libref* and a member name. If the *libref* is omitted, PROC CIMPORT uses the default library as the *libref*, which is usually the WORK library. If the *destination* argument is LIBRARY, specify only a *libref*.

Options

EET=(*etype(s)*)
excludes specified entry types from the import process. If the *etype* is a single entry type, then you can omit the parentheses. Separate multiple values with spaces.

Interaction: You cannot specify both the EET= option and the ET= option in the same PROC CIMPORT step.

ET=(*etype(s)*)
specifies the entry types to import. If the *etype* is a single entry type, then you can omit the parentheses. Separate multiple values with spaces.

Interaction: You cannot specify both the EET= option and the ET= option in the same PROC CIMPORT step.

EXTENDSN=YES | NO
specifies whether to extend by 1 byte the length of short numerics (fewer than 8 bytes) when you import them. You can avoid a loss of precision when you transport a short numeric in IBM format to IEEE format if you extend its length. You cannot extend the length of an 8-byte short numeric.

Default: YES

Restriction: This option applies only to data sets.

Tip: Do not store fractions as short numerics.

FORCE
enables access to a locked catalog. By default, PROC CIMPORT locks the catalog that it is updating to prevent other users from accessing the catalog while it is being updated. The FORCE option overrides this lock, which allows other users to access the catalog while it is being imported, or allows you to import a catalog that is currently being accessed by other users.

CAUTION:
The FORCE option can lead to unpredictable results. The FORCE option allows multiple users to access the same catalog entry simultaneously. △

INFILE=*fileref* | '*filename*'
specifies a previously defined fileref or the filename of the transport file to read. If you omit the INFILE= option, then PROC CIMPORT attempts to read from a transport file with the fileref SASCAT. If a fileref SASCAT does not exist, then PROC CIMPORT attempts to read from a file named SASCAT.DAT.

Alias: FILE=

Featured in: Example 1 on page 217.

MEMTYPE=*mtype*

specifies that only data sets, only catalogs, or both, be moved when a SAS library is imported. Values for *mtype* can be

ALL
both catalogs and data sets

CATALOG | CAT
catalogs

DATA | DS
SAS data sets

NEW

creates a new catalog to contain the contents of the imported transport file when the destination you specify has the same name as an existing catalog. NEW deletes any existing catalog with the same name as the one you specify as a destination for the import. If you do not specify NEW, and the destination you specify has the same name as an existing catalog, PROC CIMPORT appends the imported transport file to the existing catalog.

NOEDIT

imports SAS/AF PROGRAM and SCL entries without edit capability.

You obtain the same results if you create a new catalog to contain SCL code by using the MERGE statement with the NOEDIT option in the BUILD procedure of SAS/AF Software.

Note: The NOEDIT option affects only SAS/AF PROGRAM and SCL entries. It does not affect FSEDIT SCREEN and FSVIEW FORMULA entries. △

Alias: NEDIT

NOSRC

suppresses the importing of source code for SAS/AF entries that contain compiled SCL code.

You obtain the same results if you create a new catalog to contain SCL code by using the MERGE statement with the NOSOURCE option in the BUILD procedure of SAS/AF Software.

Alias: NSRC

Interaction: PROC CIMPORT ignores the NOSRC option if you use it with an entry type other than FRAME, PROGRAM, or SCL.

TAPE

reads the input transport file from a tape.

Default: PROC CIMPORT reads from disk.

EXCLUDE Statement

Excludes specified files or entries from the import process.

Tip: There is no limit to the number of EXCLUDE statements you can use in one invocation of PROC CIMPORT.

Interaction: You can use either EXCLUDE statements or SELECT statements in a PROC CIMPORT step, but not both.

EXCLUDE *SAS file(s)* | *catalog entry(s)*</ MEMTYPE=*mtype*></ ENTRYTYPE=*entry-type*>;

Required Arguments

SAS file(s) | catalog entry(s)

specifies either the name(s) of one or more SAS files or the name(s) of one or more catalog entries to be excluded from the import process. Specify SAS file names if you import a data library; specify catalog entry names if you import an individual SAS catalog. Separate multiple file names or entry names with a space. You can use shortcuts to list many like-named files in the EXCLUDE statement. For more information, see "Shortcuts for Specifying Lists of Variable Names" on page 58.

Options

ENTRYTYPE=*entry-type*

specifies a single entry type for the catalog entry(s) listed in the EXCLUDE statement. See *SAS Language Reference: Concepts* for a complete list of catalog entry types.

Restriction: ENTRYTYPE= is valid only when you import an individual SAS catalog.

Alias: ETYPE=, ET=

MEMTYPE=*mtype*

specifies a single member type for the SAS file(s) listed in the EXCLUDE statement. Values for *mtype* can be

ALL
both catalogs and data sets

CATALOG
catalogs

DATA
SAS data sets.

You can also specify the MEMTYPE= option, enclosed in parenthesis, immediately after the name of a file. In parenthesis, MEMTYPE= identifies the type of the filename that just precedes it. When you use this form of the option, it overrides the MEMTYPE= option that follows the slash in the EXCLUDE statement, but it must match the MEMTYPE= option in the PROC CIMPORT statement.

Restriction: MEMTYPE= is valid only when you import a SAS data library.

Alias: MTYPE=, MT=

Default: ALL

SELECT Statement

Specifies individual files or entries to import.

Tip: There is no limit to the number of SELECT statements you can use in one invocation of PROC CIMPORT.

Interaction: You can use either EXCLUDE statements or SELECT statements in a PROC CIMPORT step, but not both.

Featured in: Example 2 on page 218

SELECT *SAS file(s)* | *catalog entry(s)*</ MEMTYPE=*mtype*></
ENTRYTYPE=*entry-type*>;

Required Arguments

SAS file(s) | catalog entry(s)
specifies either the name(s) of one or more SAS files or the name(s) of one or more catalog entries to import. Specify SAS filenames if you import a data library; specify catalog entry names if you import an individual SAS catalog. Separate multiple filenames or entry names with a space. You can use shortcuts to list many like-named files in the SELECT statement. For more information, see "Shortcuts for Specifying Lists of Variable Names" on page 58.

Options

ENTRYTYPE=*entry-type*
specifies a single entry type for the catalog entry(s) listed in the SELECT statement. See *SAS Language Reference: Concepts* for a complete list of catalog entry types.

Restriction: ENTRYTYPE= is valid only when you import an individual SAS catalog.

Alias: ETYPE=, ET=

MEMTYPE=*mtype*
specifies a single member type for the SAS file(s) listed in the SELECT statement. Valid values are CATALOG or CAT, DATA, or ALL.

You can also specify the MEMTYPE= option, enclosed in parenthesis, immediately after the name of a file. In parenthesis, MEMTYPE= identifies the type of the file name that just precedes it. When you use this form of the option, it overrides the MEMTYPE= option that follows the slash in the SELECT statement, but it must match the MEMTYPE= option in the PROC CIMPORT statement.

Restriction: MEMTYPE= is valid only when you import a SAS data library.

Alias: MTYPE=, MT=

Default: ALL

Results

Data Control Block Characteristics for Mainframe Environments

A common problem when you create or import a transport file under the OS/390 or CMS environments is a failure to specify the correct Data Control Block (DCB) characteristics. When you reference a transport file you must specify the following DCB characteristics:

LRECL: 80

BLKSIZE: 8000

RECFM: FB

Note: A BLKSIZE value of less than 8000 may be more efficient for your storage device in some cases. The BLKSIZE value must be an exact multiple of the LRECL value. △

Another common problem can occur if you use communications software to move files from another environment to OS/390 or CMS. In some cases, the transport file does not have the proper DCB characteristics when it arrives on OS/390 or CMS. If the communications software does not allow you to specify file characteristics, try the following approach for OS/390:

1. Create a file under OS/390 with the correct DCB characteristics and initialize the file.
2. Move the transport file from the other environment to the newly created file under OS/390 using binary transfer.

Try this approach for CMS:

1. Log onto CMS.
2. Connect to the other operating environment.
3. Pull the file from the other operating environment.

Examples

Example 1: Importing an Entire Data Library

Procedure features:

PROC CIMPORT statement option:

INFILE=

This example shows how to use PROC CIMPORT to read from disk a transport file, named TRANFILE, that PROC CPORT created from a SAS data library in another operating environnment. The transport file was moved to the new operating environnment by means of communications software or magnetic medium. PROC CIMPORT imports the transport file to a SAS data library, called NEWLIB, in the new operating environnment.

Program

The LIBNAME statement specifies a libname for the new SAS data library. The FILENAME statement specifies the filename of the transport file that PROC CPORT created and enables you to specify any operating environnment options for file characteristics.

```
libname newlib 'SAS-data-library';
filename tranfile 'transport-file'
                  host-option(s)-for-file-characteristics;
```

PROC CIMPORT imports the SAS data library into the library named NEWLIB.

```
proc cimport library=newlib infile=tranfile;
run;
```

SAS Log

```
 NOTE: Proc CIMPORT begins to create/update catalog NEWLIB.FINANCE
NOTE: Entry LOAN.FRAME has been imported.
NOTE: Entry LOAN.HELP has been imported.
NOTE: Entry LOAN.KEYS has been imported.
NOTE: Entry LOAN.PMENU has been imported.
NOTE: Entry LOAN.SCL has been imported.
NOTE: Total number of entries processed in catalog NEWLIB.FINANCE: 5

NOTE: Proc CIMPORT begins to create/update catalog NEWLIB.FORMATS
NOTE: Entry REVENUE.FORMAT has been imported.
NOTE: Entry DEPT.FORMATC has been imported.
NOTE: Total number of entries processed in catalog NEWLIB.FORMATS: 2
```

Example 2: Importing Individual Catalog Entries

Procedure features:

PROC CIMPORT statement options:

INFILE=

SELECT statement

This example shows how to use PROC CIMPORT to import the individual catalog entries LOAN.PMENU and LOAN.SCL from the transport file TRANS2, which was created from a single SAS catalog.

Program

The LIBNAME statement specifies a libname for the new SAS data library. The FILENAME statement specifies the filename of the transport file that PROC CPORT created and enables you to specify any operating environnment options for file characteristics.

```
libname newlib 'SAS-data-library';
filename trans2 'transport-file'
                 host-option(s)-for-file-characteristics;
```

PROC CIMPORT imports the individual catalog entries from the TRANS2 transport file and stores them in a new SAS catalog called NEWLIB.FINANCE. The SELECT statement selects only the two specified entries from the transport file to be imported into the new catalog.

```
proc cimport catalog=newlib.finance infile=trans2;
   select loan.pmenu loan.scl;
run;
```

SAS Log

```
NOTE: Proc CIMPORT begins to create/update catalog NEWLIB.FINANCE
NOTE: Entry LOAN.PMENU has been imported.
NOTE: Entry LOAN.SCL has been imported.
NOTE: Total number of entries processed in catalog NEWLIB.FINANCE: 2
```

Example 3: Importing a Single Indexed SAS Data Set

Procedure features:

PROC CIMPORT statement option:

INFILE=

This example shows how to use PROC CIMPORT to import an indexed SAS data set from a transport file that was created by PROC CPORT from a single SAS data set.

Program

The LIBNAME statement specifies a libname for the new SAS data library. The FILENAME statement specifies the filename of the transport file that PROC CPORT created and enables you to specify any operating environnment options for file characteristics.

```
libname newdata 'SAS-data-library';
filename trans3 'transport-file'
                 host-option(s)-for-file-characteristics;
```

PROC CIMPORT imports the single SAS data set that you identify with the DATA= specification in the PROC CIMPORT statement. PROC CPORT exported the data set NEWDATA.TIMES in the transport file TRANS3.

```
proc cimport data=newdata.times infile=trans3;
run;
```

SAS Log

```
NOTE: Proc CIMPORT begins to create/update data set NEWDATA.TIMES
NOTE: The data set index x is defined.
NOTE: Data set contains 2 variables and 2 observations.
      Logical record length is 16
```

The correct bibliographic citation for this manual is as follows: SAS Institute Inc., *SAS® Procedures Guide, Version 8,* Cary, NC: SAS Institute Inc., 1999. 1729 pp.

SAS® Procedures Guide, Version 8

ISBN 1–58025–482–9

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

1st printing, October 1999

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