

UNDOCUMENTED FEATURES TO TEST FOR EXISTENCE OF DATA SETS

John Beach, BlueCross BlueShield of Connecticut Inc.
Presented by Frank Soloman, BlueCross BlueShield of Connecticut, Inc.

To test for the existence of a file or library there is an undocumented command DSNEXTST with the following syntax:

```
DSNEXTST data set name;
```

Please note that there are no quotations around the data set name. A value of 1 will be returned if true (data set exists) or 0 if false. The value is examined using a new macro &SYSDEXST which is also undocumented.

To illustrate the use of this feature, imagine that there is a program which will produce a report if a data set exists or an error message if it does not. The blocks of code used to produce the report or generate the error message are imbedded within macros. Only one of the macros is executed, based upon the value from the DSNEXTST command. For example:

```
%MACRO REPORTS;  
* NOTE: code to produce a report if data set  
exists;  
;  
%MEND REPORTS;  
  
%MACRO MESSAGES;  
DATA_NULL_;  
*NOTE: PUT statements to write message if  
data set does not exist;  
;  
%MEND MESSAGES;  
  
DSNEXTST data set name;  
  
%MACRO RUNREPT;  
%IF &SYSDEXST EQ 1 %THEN %REPORTS;  
%ELSE %MESSAGES;  
%MEND RUNREPT;  
  
%RUNREPT;
```

Both the DSNEXTST command and the SYSDEXST macro are available only in the MVS environment.

There is a new automatic macro variable available for use in Release 6.07 if more information is required as to why a file cannot be accessed, other than the fact that it does not exist. This macro is SYSFILRC and should be invoked immediately after a FILENAME statement has been executed. Its value is equal to the return code produced by the execution of the FILENAME statement; namely 0 if the

statement executed correctly or a value greater than 0 if there was a problem. This macro is documented in SAS Technical Report P-222 title "Changes and Enhancements to Base SAS Software Release 6.07", page 307.

This macro cannot be used in the Windows environment because an error condition is not created when a FILENAME statement references a non-existent file.

There is a companion macro, SYSLIBRC, which performs the same function for SAS data sets. This macro is not documented but performs in the same manner and is invoked after the execution of a LIBNAME statement.

This macro can also be used under Release 6.08 of PC SAS for Windows.

If FILENAME or LIBNAME testing is done in batch mode, the OPTIONS statement should contain a SAS System Option to prevent the job from being terminated if a data set is not found. The statement OPTIONS ERRORCHECK=NORMAL should be used if Release 6.08 or higher has been installed. This option is documented in SAS Technical Report P-242 title "SAS Software: Changes and Enhancements Release 6.08" on page 33. The statement OPTIONS NO\$SYNTAXCHECK, which is undocumented, should be used if Release 6.07 is in effect.

Both of these options are also available under Release 6.08 of PC SAS for Windows.

A four digit Reason Code is also produced if the FILENAME statement was not executed correctly. The value and descriptive text are printed in an error message, but no automatic macro variable exists to return an addressable value. Hopefully, a macro will be available in the near future to perform this function.

Please note: All of the features described in this paper were tested under Releases 6.07 and 6.08 of MVS and Release 6.08 of PC SAS for Windows.

AUTHOR

John Beach
Information Systems Group
BlueCross BlueShield of Connecticut
370 Bassett Road
North Haven, CT 06473
Phone (203) 234-5788
FAX (203) 234-5790