Changes and Enhancements

Changes and Enhancements for All Operating Environments for Versions 7 and 8

Information about changes and enhancements that were implemented in Version 8 is preceded by V8. All other changes and enhancements described in this section were implemented in Version 7. In other words, if your site upgraded directly from Version 6 to Version 8, then all of the changes and enhancements described here are new to you. If you upgraded from Version 7 to Version 8, then only the items preceded by V8 are new to you.

If you use SAS software under OS/2, OpenVMS Alpha, OpenVMS VAX, UNIX, or Windows, then see also "Additional Version 7 Changes and Enhancements for OS/2, OpenVMS Alpha, OpenVMS VAX, UNIX, and Windows" on page xviii.

If you use SAS software under OS/390, CMS, or OpenVMS VAX, then see also "Additional Version 7 Changes and Enhancements for OS/390, CMS, and OpenVMS VAX" on page xix.

For information about changes and enhancements to base SAS software that are relevant only to a particular operating environment, see the SAS documentation for that operating environment.

SAS System Enhancements and New Features

Overview

New SAS System features, which are described in more detail in the following sections, include

- Year 2000 compliance
- SAS output:
  - more efficient and powerful control of output with the Output Delivery System (ODS), including the production of HTML output and output for high-resolution printers
  - new DATA step reporting features.

- SAS Files and I/O:
  - expanded rules for SAS names, including long variable names and mixed-case variable names
  - ability to refer to a permanent SAS data set with the actual physical filename rather than a SAS library reference
  - enhancements to stored compiled DATA step programs and DATA step views
  - logical concatenation of SAS libraries
  - logical concatenation of SAS catalogs
  - generation data sets
  - integrity constraints
  - indexing and WHERE processing enhancements
  - enhanced compression routines (including one option that is new for Version 8)
  - an audit trail.
- SAS windowing environment:
  - SAS Explorer window
  - SAS Registry Editor
  - Results window.

Year 2000 Compliance

For information concerning year 2000 compliance of SAS software products, see the Institute's web site:

```
http://www.sas.com/y2k
```

Also, the Professional Services Division at SAS Institute offers a course that explains the issues and describes SAS tools and techniques for achieving year 2000 compliance. See "The Year 2000: Achieving Date Compliance" in *SAS Applications Course Notes*.

With regard to changes in Version 7 of base SAS software, the default value for the YEARCUTOFF= system option is now 1920.

SAS Output: Output Delivery System

In previous versions of SAS, printed output from SAS procedures and from the DATA step was designed for a traditional line-printer. In Version 7, printed output is much more flexible. The new Output Delivery System (ODS) enhances your ability to manage both DATA step and procedure output. ODS features include the ability to

- combine the raw data that is produced with one or more templates to produce one or more output objects that contain the formatted results
- store a link to each output object in the Results folder in the Results window
- generate HTML files that contain the formatted results and that contain links to the results in the form of a table of contents
- generate output for a high-resolution printer
- generate output data sets from procedure output
- allow you to customize the procedure output by creating templates that you can use whenever you run the procedure.

This section briefly illustrates these features. For more information on the Output Delivery System, see *The Complete Guide to the SAS Output Delivery System* and see the FILE, ODS and PUT, ODS statements in this manual.

SAS Output: DATA Step Reporting

Reports that are created with the FILE and PUT statements in a DATA step have two new features:

- You can use footnotes in addition to titles.
- Reports can take advantage of the new Output Delivery System.

SAS I/O: Long Variable Names and Expanded Rules for SAS Names

The rules for SAS names have been expanded to provide more functionality. Some SAS names, such as DATA step variables, members of SAS data libraries, and catalog entries, can be up to 32 bytes long. A variable name can be mixed case.

In a DATA step or a PROC SQL step, you can more easily use DBMS files. You can use a new *name literal* to express a DBMS column name that may contain characters that are not normally allowed in SAS names. The SAS system option VALIDVARNAME= enables you to specify what rules apply to the variable names that you can create and process in a SAS session.

Referencing a Permanent SAS Data Set Without a Libref

You can create or reference a permanent SAS data set without first assigning a libref. You can refer to a SAS data set directly by specifying in quotes the name by which the operating system refers to the file.

Enhancements to Stored Compiled DATA Step Programs and DATA Step Views

In Version 8, SAS no longer requires that you recompile stored compiled DATA step programs or recreate DATA step views that were created in previous versions or releases of SAS. Stored programs and views created in Version 6 or Version 7 run as is in Version 8, provided that you run them on the same operating system on which they were created.

By default, SAS also saves the source statements when you create stored compiled programs and DATA step views in Version 8. You can retrieve and/or execute the statements by using the new DESCRIBE and EXECUTE statements within the DATA step.

SAS I/O: Logical Concatenation of SAS Libraries

You can now reference two or more SAS libraries with a single libref in all operating environments. To make your code easier to use in multiple operating environments, you can use a libref, rather than only the physical name as in Version 6, to specify the aggregate storage location that contains your SAS files.

SAS I/O: Logical Concatenation of SAS Catalogs

You can now logically concatenate entries in two or more SAS catalogs. Use the LIBNAME statement to *implicitly* concatenate entries in SAS catalogs. Use the new CATNAME statement to *explicitly* concatenate entries in SAS catalogs.

SAS I/O: Generation Data Sets

Generation data sets enable you to keep multiple copies of a SAS data set. The copies represent versions of the same data set, which is archived each time it is

replaced. To request generation data sets, use the GENMAX= data set option. To reference a specific version of a data set, use the GENNUM= data set option.

SAS I/O: Integrity Constraints

Integrity constraints enable you to specify rules that guarantee the consistency and correctness of stored data. The rules restrict the data that can be added, updated, or deleted from a data set, and they also link the data in one data set to data in another data set.

You can specify integrity constraints either when a data set is created or after it already contains data. To create and delete integrity constraints, you can use the DATASETS procedure, the SQL procedure, or SCL functions. To list integrity constraints, use the CONTENTS procedure.

SAS I/O: Indexing Enhancements

Version 7 provides the following performance improvements for indexed data sets:

- When creating an index, no sort is performed if the data set is already in the correct order.
- The system's ability to determine the more cost-efficient way to access the data (either by sequential reading or by using an index) is improved by storing statistics that represent the distribution of values.
- WHERE processing for an indexed data set is enhanced for the pattern-matching operators LIKE and NOT LIKE.
- The system makes greater use of composite indexes for *compound optimization*, which is the process of optimizing multiple WHERE conditions with a single composite index. WHERE expressions that contain the EQ or IN operator now support three new operators: directional inequalities, NOT operations, and truncated comparisons.
- Two new data set options, IDXNAME= and IDXWHERE=, provide more control over index usage.
- The MSGLEVEL= system option provides more information about index usage.
- When you append to an indexed data set, SAS improves performance by not updating the index until all observations have been added.

SAS I/O: Enhancements for Compressed Files

Compressed data sets are more flexible to use. You can now access observations in a compressed SAS data set directly by using the POINT= or FIRSTOBS= option.

You can choose a compression algorithm that best fits your data. For compressing character data, specify CHAR to use the RLE (Run Length Encoding) algorithm. For compressing binary (for example, numeric) data, specify BINARY to use the RDC (Ross Data Compression) algorithm. If your application can benefit from a specialized algorithm, you can supply your own if you also license SAS/TOOLKIT software.

In Version 8, you can use the POINTOBS= data set option to control whether a compressed data set may be processed with random access rather than with sequential access only.

SAS I/O: Audit Trail

An audit trail enables you to log updates to a SAS data file in a separate audit file. Each time an observation is added, deleted, or updated, information is added to the audit file about who made the modification, what was modified, and when.

You can also define special variables that are stored in the audit file. The user can assign values to them just like regular data set variables. They are stored with each record written to the audit trail. For example, they could be used to log a "reason for modification."

The audit trail is initiated with the AUDIT statement in PROC DATASETS. See *SAS Procedures Guide* for more information.

Double-Byte Character Sets

Starting with Release 6.07, SAS has supported double-byte character sets (DBCS), which are used for many Asian languages, such as Japanese, Korean, simplified Chinese, and traditional or complex Chinese. The documentation for Version 8 contains the first complete version of the DBCS information in English. See the category tables at the beginning of the Formats, Functions, Informats, and System Options chapters for a complete listing of DBCS language elements.

Windowing Environment: SAS Explorer Window

The SAS workspace is composed of windows that enable you to accomplish specific tasks. Along with the main programming windows (Program Editor, Log, and Output), the workspace now includes SAS Explorer, a central point for managing basic SAS software tasks such as

- viewing and managing tables, libraries, and members
- creating new libraries and file shortcuts
- creating new library members and catalog entries.

If you issue the CAT, DIR, LIB, or FILENAME commands, an Explorer window opens. For example, if you type CAT WORK.TEST, you will see the contents of your TEST catalog in a modified Explorer window.

Windowing Environment: SAS Registry Editor

The SAS Registry stores configuration data about the SAS session and about various applications. You can use the SAS Registry Editor window to

- view the contents of the registry, showing keys as well as the values that are stored in the keys
- add, modify, and delete keys and key values
- export and import registry files to and from the registry
- compare the contents of a file to the registry
- install and uninstall registry files.

You can also use PROC REGISTRY to accomplish these tasks.

Windowing Environment: Results Window

The Results window contains pointers to various locations in the Output window. When procedure code executes, pointers to specific output are generated and are listed in the Results window. You can click the navigation pointers to locate the desired output.

New and Enhanced Base SAS Language Elements

Data Set Options

- Changes to data set options include the following:

- COMPRESS= accepts additional values.
- RENAME= accepts variable lists.

- These are the new data set options for data set control:

  DLDMGACTION=
  specifies what type of action to take when a SAS file in a SAS data library is detected as damaged.

  GENMAX=
  requests generations for a data set and specifies the maximum number of versions.

  GENNUM=
  references a specific generation of a data set.

  OUTREP=
  specifies an operating environment's requirements vector for an output file.

  V8 REPEMPTY=
  controls replacement of like-named temporary or permanent SAS data sets when the new one is empty.

  TRANTAB=
  specifies a translation table for character conversions.

- These are the new data set options for observation control:

  IDXNAME=
  directs the SAS System to use a specific index to satisfy the conditions of a WHERE expression.

  IDXWHERE=
  overrides the SAS System decision about whether to use an index to satisfy the conditions of a WHERE expression.

  V8 POINTOBS=
  controls whether a data set may be processed with random access (by observation number) rather than with sequential access only.

  WHEREUP=
  specifies whether to evaluate added observations and modified observations against a WHERE expression.

Formats

- Changes to formats include the following:
  - Character formats can handle strings up to 32,767 characters (32K) long.
  - The DOLLAR*w.d*, DOLLARX*w.d*, COMMA*w.d*, and COMMAX*w.d* formats no longer restrict *d* to 0 or 2.
- The following formats are new:

  DDMMYY*xw*.
  writes date values in the form *ddmmyy* or *ddmmyyyy* and specifies separators or no separators between date values.

  IBR*w.d*
  writes integer binary (fixed-point) values in Intel and DEC format.

  MMDDYY*xw*.
  writes date values in the form *mmddyy* or *mmddyyyy* and specifies separators or no separators between date values.

PIBR*w.d*
writes positive integer binary (fixed-point) values in Intel and DEC formats.

PDJULG*w.*
writes packed Julian date values in the hexadecimal format *yyyyddd*F for IBM.

PDJULI*w.*
writes packed Julian date values in the hexadecimal format *ccyyddd*F for IBM.

YYMMDD*xw*.
writes date values in the form *yymmdd* or *yyyymmdd* and specifies separators or no separators between date values.

Functions and CALL Routines

- Changes to existing functions include the following:
  - The PUT function now supports alignment of values with formats.
  - The QUOTE function no longer strips trailing blanks from values.
  - You can use the SCAN function to scan from right to left by specifying a negative number as an argument.
- The following mathematical and probability functions have been added to base SAS software:

  COMB
  computes the number of combinations of *n* elements taken *r* at a time and returns a value.

  CONSTANT
  computes some machine and mathematical constants and returns a value.

  DEVIANCE
  computes the deviance and returns a value.

  FACT
  computes a factorial and returns a value.

  PERM
  computes the number of permutations of *n* elements taken *r* at a time and returns a value.

  PROBBNRM
  computes a probability from the bivariate normal distribution and returns a value.

  PROBMC
  computes a probability or a quantile from various distributions for multiple comparisons of means and returns a value.
- The following new RX functions and CALL routines provide character-string matching functionality. That is, they enable you to search for (and, optionally, to replace) patterns or characters in a string.

  CALL RXCHANGE
  changes one or more substrings that match a pattern.

  CALL RXFREE
  frees memory that is allocated by other regular expression (RX) functions and CALL routines.

CALL RXSUBSTR
finds the position, length, and score of a substring that matches a pattern.

RXMATCH
finds the beginning of a substring that matches a pattern and returns a value.

RXPARSE
parses a pattern and returns a value.

- The following functions return information about the attributes of variables. The functions with names that end in an "X" can accept an expression as an argument; the others require a variable name as an argument:

  VARRAY
  returns a value that indicates whether the given variable is an array.

  VARRAYX
  returns a value that indicates whether the value of a given argument is an array.

  VFORMAT
  returns the format that is associated with the given variable.

  VFORMATD
  returns the format decimal value that is associated with the given variable.

  VFORMATDX
  returns the format decimal value that is associated with the value of a given argument.

  VFORMATX
  returns the format that is associated with the value of a given argument.

  VFORMATN
  returns the format name that is associated with the given variable.

  VFORMATNX
  returns the format name that is associated with the value of a given argument.

  VFORMATW
  returns the format width that is associated with the given variable.

  VFORMATWX
  returns the format width that is associated with the value of a given argument.

  VINARRAY
  returns a value that indicates whether the given variable is a member of an array.

  VINARRAYX
  returns a value that indicates whether the value of a given argument is a member of an array.

  VINFORMAT
  returns the informat that is associated with the given variable.

  VINFORMATX
  returns the informat that is associated with the value of a given argument.

  VINFORMATD
  returns the informat decimal value that is associated with the given variable.

VINFORMATDX
returns the informat decimal value that is associated with the value of a given argument.

VINFORMATN
returns the informat name that is associated with the given variable.

VINFORMATNX
returns the informat name that is associated with the value of a given argument.

VINFORMATW
returns the informat width that is associated with the given variable.

VINFORMATWX
returns the informat width that is associated with the value of a given argument.

VLABEL
returns the label that is associated with the given variable.

VLABELX
returns the variable label for the value of a given argument.

VLENGTH
returns the compile-time (allocated) size of the given variable.

VLENGTHX
returns the compile-time (allocated) size for the value of a given argument.

VNAME
returns the name of the given variable.

VNAMEX
validates the value of a given argument as a variable name.

VTYPE
returns the type (character or numeric) of the given variable.

VTYPEX
returns the type (character or numeric) for the value of a given argument.

- These date and time functions are new:

DATDIF
returns the number of days between two dates.

JULDATE7
returns a seven digit Julian date from a SAS date value.

YRDIF
returns the difference in years between two dates.

- The following functions are also new:

IORCMSG
returns a formatted error message for _IORC_.

MISSING
returns a numeric result that indicates whether the argument contains a missing value.

- The following web-based functions have been added to base SAS software:

HTMLDECODE
decodes a string containing HTML numeric character references or HTML character entity references and returns the decoded string.

HTMLENCODE
encodes characters using HTML character entity references and returns the encoded string.

URLDECODE
returns a string that was decoded using the URL escape syntax.

URLENCODE
returns a string that was encoded using the URL escape syntax.

- In Version 8, the following financial functions have been added to base SAS software:

CONVX
returns the convexity for an enumerated cashflow.

CONVXP
returns the convexity for a periodic cashflow stream, such as a bond.

DUR
returns the modified duration for an enumerated cashflow.

DURP
returns the modified duration for a periodic cashflow stream, such as a bond.

PVP
returns the present value for a periodic cashflow stream, such as a bond.

YIELDP
returns the yield-to-maturity for a periodic cashflow stream, such as a bond.

Informats

- Character informats can handle strings up to 32,767 characters (32K) long.
- The following informats are new:

IBR*w.d*
reads integer binary (fixed-point) values in Intel and DEC format.

V8 JDATEYMD
reads Japanese kanji date values in the format *yymmmdd* or *yyyymmmdd*.

V8 JNENGO
reads Japanese Kanji date values in the form *yymmdd*.

PDJULG*w*.
reads packed Julian date values that are produced on IBM mainframe systems in the hex representation form of *yyyyddd*F. and converts the values to SAS date values.

PDJULI*w*.
reads packed Julian date values that are produced on IBM mainframe systems in the form *ccyydd*F. and converts the values to SAS date values.

PIBR*w.d*
reads positive integer binary (fixed-point) values in Intel and DEC formats.

SHRSTAMP.*w*
reads date and time values of SHR records.

YYMMN*w*.
reads date values that are in the form *yymm* or *yyyymm* and that contain no delimiters, then adds a date value of 01 so that these values become valid SAS date values.

Statements

- There are three new statements:

  CATNAME
  logically concatenates the contents of two or more SAS catalogs. It can also assign or de-assign a catref or list all currently assigned catrefs.

  V8 DESCRIBE
  retrieves source code from a stored compiled DATA step program or a DATA step view and writes it to the SAS log. It is used within a DATA step with the DATA statement's PGM= or VIEW= options.

  V8 EXECUTE
  executes a stored compiled DATA step program. It is used within a DATA step with the DATA statement's PGM= option.

- Enhancements to statements include the following:
  - The FILE statement now has a FOOTNOTES option, which allows DATA step reports that are created with a PUT statement to use footnotes. Using the DELIMITER= option, you can write data that are delimited by a specified character other than a blank. Using the DSD option, you can write data that contain a delimiter.
  - The FILE statement contains new options that allow you to produce output in HTML format.
  - In the INFILE statement, you can now use the _INFILE_= option to assign the contents of the current input buffer to a variable.
  - In the FILE statement, you can now use the _FILE_= option to assign the contents of the current output buffer to a variable.
  - New host options are available for the INFILE and FILE statements. See the SAS documentation for your operating environment.
  - The LIBNAME statement can implicitly concatenate two or more SAS catalogs. (See the CATNAME statement.)
  - The LIBNAME statement now has the REPEMPTY= option, which controls the replacement of like-named temporary or permanent SAS data sets when the new one is empty.
  - The LIBNAME statement provides SAS/ACCESS users with an easier way to access DBMS tables.

    Note: In order to use this new feature of the LIBNAME statement, you must license SAS/ACCESS software. △
  - The PUT statement contains new options that allow you to produce output in HTML format.
  - The RENAME statement accepts variable lists.
  - In the SET statement, you can use the OPEN= option to open only the first data set that is listed in the SET statement during compilation. Some restrictions apply.

System Options

- The following system options were changed:
  - The RESERVEDB1 system option is now called MFILE. It controls whether or not MPRINT output is directed to an external file.
  - The default value for YEARCUTOFF= is 1920.
- The following are new communication system options:

CONNECTREMOTE=
specifies the remote session ID that is used for SAS/CONNECT software.

CONNECTSTATUS
specifies whether or not to display the SAS/CONNECT transfer status window.

CONNECTWAIT
specifies whether or not to wait for the SAS/CONNECT remote submit statement (RSUBMIT) to complete before control returns to the local session.

NETENCRYPT
specifies encryption of all network communications.

NETENCRYPTALGORITHM=
specifies the algorithm to use for the encryption of data that are passed over the network.

NETENCRYPTKEYLEN=
specifies the key size for network data encryption.

NETMAC
controls whether SAS uses Message Authentication Codes (MACs) to detect message corruptions across a network.

TBUFSIZE=
specifies the buffer size to use when you transmit data with SAS/CONNECT or SAS/SHARE software.

TCPPORTFIRST=
specifies the first TCP/IP port for SAS/CONNECT software.

TCPPORTLAST=
specifies the last TCP/IP port for SAS/CONNECT software.

- The following are new environment control system options:

V8 APPLETLOC
specifies the location of Java applets.

V8 CMPOPT
controls whether SAS language compiler optimization is in effect.

DMSEXP
invokes SAS with the Explorer, Program, Log, Output, and Results windows.

DOCLOC=
specifies the base location of SAS online documentation.

EXPLORER
controls whether you invoke SAS with the Explorer window only.

V8 MERGENOBY
controls whether a message is issued when MERGE processing occurs without an associated BY statement.

REP_MGRLOC=
specifies the location of the repository manager for common metadata.

V8 SASCMD
used by the SIGNON portion of SAS/CONNECT to invoke a remote or server SAS session.

SOLUTIONS
specifies whether SOLUTIONS appears on menus in the SAS windows.

V8 STARTLIB
allows previous library references to persist in a new SAS session.

TRAINLOC=
specifies the base location of SAS online training courses.

VALIDVARNAME=
controls the type of SAS variable names that can be created or processed during a SAS session.

- The following are new log and procedure output control system options:

  CONSOLELOG=
  specifies the location of the console log.

  PRINTMSGLIST
  controls the printing of extended lists of messages to the SAS log.

- The following are new sort system options:

  SORTDUP=
  controls the SORT procedure's application of the NODUP option to physical or logical records.

  SUMSIZE=
  specifies a limit on the amount of memory that is available for data summarization procedures when class variables are active.

- The following are new file control system options:

  V8 ASYNCHIO
  specifies whether asynchronous I/O is enabled.

  DLDMGACTION=
  specifies what type of action to take when a SAS file in a SAS data library is detected as damaged.

  V8 SYNCHIO
  specifies whether synchronous I/O is enabled.

- The following are new graphics system options:

  GISMAPS=
  specifies the location of the SAS data library that contains SAS/GIS-supplied TIGER map data sets.

- In V8, the following new system options control printing with the Output Delivery System (ODS):

  BINDING=
  specifies the binding edge for the ODS printer destination.

  BOTTOMMARGIN=
  specifies the size of the margin at the bottom of the page for the ODS printer destination.

  COLLATE
  specifies the collation of multiple copies for output for the ODS printer destination.

  COLORPRINTING
  specifies color printing, if it is supported, for the ODS printer destination.

  COPIES=
  specifies the number of copies to make when printing to the ODS printer destination.

DUPLEX=
specifies duplexing controls for the ODS printer destination.

LEFTMARGIN=
specifies the size of the margin on the left side of the page for the ODS printer destination.

ORIENTATION=
specifies the paper orientation to use when printing to the ODS printer destination.

PAPERDEST=
specifies the bin to receive printed output for the ODS printer destination.

PAPERSIZE=
specifies the paper size to use when printing to the ODS printer destination.

PAPERSOURCE=
specifies the paper bin to use for printing to the ODS printer destination.

PAPERTYPE=
specifies the type of paper to use for printing to the ODS printer destination.

PRINTERPATH=
specifies a printer for SAS print jobs directed to the ODS printer destination.

RIGHTMARGIN=
specifies the size of the margin at the right side of the page for printed output directed to the ODS printer destination.

TOPMARGIN=
specifies the size of the margin at the top of the page for the ODS printer destination.

- The following is a new system administration option:

  V8 SETINIT
  controls whether site license information can be altered.

Additional Version 7 Changes and Enhancements for OS/2, OpenVMS Alpha, OpenVMS VAX, UNIX, and Windows

This section describes changes and enhancements that are new for OS/2, OpenVMS Alpha, OpenVMS VAX, UNIX, and Windows, but which are not available on CMS or OS/390.

New and Enhanced Windowing Environment Features

Windowing Environment: External File Interface

The External File Interface (EFI) is a point-and-click graphical interface that you can use to read and write data that is not in SAS software's internal format. That is, you can read data from an external file and write it to a SAS data set, and you can read data from a SAS data set and write it to an external file.

The EFI now supports 32-character variable and data set names and is case sensitive for variable names. The EFI enables you to specify a logical record length for both list input data and list output data, and to store templates in a user-specified metabase

repository. The EFI also generates simpler export DATA step code by using the Version 7 FILE statement options DSD= and DELIMITER=.

Windowing Environment: Import/Export Wizard

The Import/Export Wizard is a tool that transfers data between external data sources and SAS data sets. Depending on the operating environment, SAS supports additional external sources. The details for each operating environment are provided in the online help.

You can also use the wizard to generate IMPORT or EXPORT procedure statements. You can then save these statements to a file for later use. That is, you can include the statements in SAS programs to import or export data without using the wizard.

Additional Version 7 Changes and Enhancements for OS/390, CMS, and OpenVMS VAX

Some changes and enhancements that were implemented for other operating environments in the 6.10, 6.11, and 6.12 releases were not implemented for OS/390, CMS, and OpenVMS VAX until Version 7. This section describes these additional features.

New and Enhanced Base SAS Language Elements

Formats

The following formats are new:

DATEAMPM*w.d*
writes datetime values (*ddmmmyy:hh:mm:ss.ss*) with AM or PM.

NUMX*w.d*
writes numeric values with a comma for the decimal point.

TIMEAMPM*w.d*
writes hours, minutes, and seconds with AM or PM.

Functions and CALL Routines

- The following CALL routine is new:

  CALL SET
  links SAS data set variables to DATA step or macro variables of the same name and data type.

- These data set I/O functions are new:

  ATTRC
  returns the value of a character attribute for a SAS data set.

  ATTRN
  returns the value of a numeric attribute for a SAS data set.

  CLOSE
  closes a SAS data set.

  CUROBS
  returns the observation number of the current observation.

DROPNOTE
deletes a note marker from a SAS data set or an external file.

DSNAME
returns the data set name that is associated with a data set identifier.

EXIST
verifies the existence of a SAS data library member.

FETCH
reads the next nondeleted observation from a SAS data set into the Data Set Data Vector (DDV).

FETCHOBS
reads a specified observation from a SAS data set into the Data Set Data Vector (DDV).

NOTE
returns an observation ID for the current observation of a SAS data set.

OPEN
opens a SAS data set.

POINT
locates an observation that is identified by the NOTE function.

REWIND
positions the data set pointer at the beginning of a SAS data set.

- These external file I/O functions are new:

DCLOSE
closes a directory.

DINFO
returns information about a directory.

DNUM
returns the number of members in a directory.

DOPEN
opens a directory.

DOPTNAME
returns directory attribute information.

DOPTNUM
returns the number of information items available for a directory.

DREAD
returns the name of a directory member.

FAPPEND
appends the current record to the end of an external file.

FCLOSE
closes an external file, directory, or directory member.

FCOL
returns the current column position in the File Data Buffer (FDB).

FDELETE
deletes an external file.

FEXIST
verifies the existence of an external file that is associated with a fileref.

FGET
copies data from the File Data Buffer (FDB) into a variable.

FILEEXIST
verifies the existence of an external file by its physical name.

FILENAME
assigns or deassigns a fileref for an external file, directory, or output device.

FILEREF
verifies that a fileref has been assigned for the current SAS session.

FINFO
returns a file information item.

FNOTE
identifies the last record that was read.

FOPEN
opens an external file.

FOPTNAME
returns the name of an item of information about a file.

FOPTNUM
returns the number of information items that are available about an external file.

FPOINT
positions the read pointer on the next record to be read.

FPOS
sets the position of the column pointer in the File Data Buffer (FDB).

FPUT
moves data to the File Data Buffer (FDB) of an external file starting at the FDB's current column position.

FREAD
reads a record from an external file into the File Data Buffer (FDB).

FREWIND
positions the file pointer at the start of the file.

FRLEN
returns the size of the last record read, or, if the file is opened for output, returns the current record size.

FSEP
sets the token delimiters for the FGET function.

FWRITE
writes a record to an external file.

MOPEN
opens a file by directory ID and member name.

PATHNAME
returns the physical name of a SAS data library or an external file.

- The following new functions are for libraries and catalogs:

CEXIST
verifies the existence of a SAS catalog or SAS catalog entry and returns a value.

LIBNAME
assigns or deassigns a libref for a SAS data library and returns a value.

LIBREF
verifies that a libref has been assigned and returns a value.

- The following are new special functions:

  GETOPTION
  returns the value of a SAS system option or graphics option.

  SPEDIS
  returns the likelihood of two words matching, expressed as the asymmetric spelling distance between two words.

  SYSGET
  returns the value of the specified operating environment variable.

  SYSMSG
  returns the text of error messages or warning messages from the last data set or external file function execution.

  SYSRC
  returns a system error number.

- These new functions return information about variables:

  GETVARC
  returns the value of a SAS data set character variable.

  GETVARN
  returns the value of a SAS data set numeric variable.

  VARFMT
  returns the format that is assigned to a SAS data set variable.

  VARINFMT
  returns the informat that is assigned to a SAS data set variable.

  VARLABEL
  returns the label that is assigned to a SAS data set variable.

  VARLEN
  returns the length of a SAS data set variable.

  VARNAME
  returns the name of a SAS data set variable.

  VARNUM
  returns the number of a SAS data set variable's position in a SAS data set.

  VARTYPE
  returns the data type of a SAS data set variable.

- The following new functions are used in probability and statistics:

  CDF
  computes cumulative distribution functions.

  LOGPDF | LOGPMF
  computes the logarithm of a probability (mass) function.

  LOGSDF
  computes the logarithm of a survival function.

  PDF | PMF
  computes probability density (mass) functions.

SDF
computes a survival function.

Informats

The following informat is new:

NUMX*w.d*
reads numeric values in which a comma marks the decimal place.

Statements

The FILENAME statement has been updated with a CATALOG access method and a URL access method.

The correct bibliographic citation for this manual is as follows: SAS Institute Inc., *SAS® Language Reference, Version 8,* Cary, NC: SAS Institute Inc., 1999.

SAS® Language Reference, Version 8

ISBN 1–58025–369–5

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

1st printing, October 1999