

Finding Out about Formats and Their Attributes

Richard D. Langston, SAS Institute Inc., Cary NC

ABSTRACT

This paper discusses format enumeration (via the `DICTIONARY.FORMATS` view) and the new `FMTINFO` function that gives information on a format, such as whether it is a date or currency format.

INTRODUCTION

If you want to find out all of the formats that are available to you in a SAS® session, you can use the special view called `DICTIONARY.FORMATS` to learn what formats you can use. Further, you can use the new `FMTINFO` function to obtain information about each format that is available.

USING `DICTIONARY.FORMATS`

`DICTIONARY.FORMATS` is a special SQL view that can be referenced only within `PROC SQL`. But you can create SAS data sets by referencing it and use those data sets subsequently. For example:

```
proc sql;
  create table work.myfmts as select * from dictionary.formats;
quit;
```

Note also that you can access this logical view via the `SASHELP.VFORMAT` view. The corresponding code for accessing that is:

```
data work.myfmts; set sashelp.vformat; run;
```

Both examples produce the same data set contents in `WORK.MYFMSTS`. If you were to run `PROC CONTENTS` on `WORK.MYFMSTS`, using the `POSITION` option, you would see the following list of variables and label descriptions:

1	libname	Char	8	Library Name
2	memname	Char	32	Member Name
3	path	Char	1024	Pathname
4	objname	Char	32	Object Name
5	fmtname	Char	32	Format Name
6	fmttype	Char	1	Format Type
7	source	Char	1	Format Source
8	minw	Num	8	Minimum Width
9	mind	Num	8	Minimum Decimal Width
10	maxw	Num	8	Maximum Width
11	maxd	Num	8	Maximum Decimal Width
12	defw	Num	8	Default Width
13	defd	Num	8	Default Decimal Width

The data set contains information about all available formats. There are the "intrinsic" formats, which are the ones provided to you as part of SAS software, such as `BEST`, `COMMA`, and `DATE`. But the data set also contains information on those formats available to you based on the `FMTSEARCH=` option. You can indicate any number of catalogs of formats that were created by `PROC FORMAT`, including those in `WORK.FORMATS` that would have been created within your current SAS session.

So the `libname` and `memname` variables listed above pertain only to those catalog-based formats, and provide the catalog name. `fmtname` is the actual format or informat name, and `fmttype` is 'F' for formats

and 'I' for informats. Minw, maxw, and defw are the minimum, maximum, and default widths for the format, while mind, maxd, and defd are the same for decimal widths.

To better understand the variables path, objname and source, you need to first understand how formats can be packaged in SAS software. A product provided by SAS may have packages of formats written in a programming language such as C. These formats are compiled and linked into an executable image (such as a DLL on Windows). The naming convention for these images has the first characters as UW (for "user written") followed by F for a format package and I for an informat package. The remaining letters (up to five) in naming the package are the first five letters of all the formats defined therein. So, for example, the S370F* formats such as S370FPIB and S370FIB, are packaged together into an image named UWFS370F.DLL on Windows. This naming convention allows for the dynamic addition of such images as needed.

With that information known, the path variable is the actual path location of the UWF* or UWI* image. Objname is the name of the image (such as UWFS370F), and source is the type of image. The value of source can be U (for UW* image) or C (for catalog). If the format is catalog-based, path is blank and objname is the same name as the format (but without a leading \$ if it is a character format).

Every time you access SASHELP.VFORMAT or DICTIONARY.FORMATS, an action called "enumeration" takes place. It is done every time because there can be changes to the FMTSEARCH= option, or to the catalogs that it references. And the first time that the enumeration takes place in a SAS session, more execution time is taken because each of the UWI* and UWF* images have to be individually loaded to obtain metadata.

If you reference the view for format enumeration, there are over 1000 formats, not including any that you would add using PROC FORMAT.

As an example of how PROC FORMAT based formats and informats would appear in SASHELP.VFORMAT, here we create a numeric and a character format and informat.

```
proc format;
  invalue myninfmt 'one'=1 'two'=2;
  invalue $mycinfmt 'abc'='yes' 'def'='no';
  value mynfmt 1='xyz' 2='ijk';
  value $mycfmt 'ddd'='dees' 'eee'='ease';
run;
data temp; set sashelp.vformat;
  if source='C';
run;
proc print; run;
```

Obs	libname	memname	path	objname	fmtname	fmttype	source	minw	mind	maxw	maxd	defw	defd
1	WORK	FORMATS		MYCFMT	\$MYCFMT	F	C	0	0	0	0	0	0
2	WORK	FORMATS		MYCINFMT	\$MYCINFMT	I	C	0	0	0	0	0	0
3	WORK	FORMATS		MYNFMT	MYNFMT	F	C	0	0	0	0	0	0
4	WORK	FORMATS		MYNINFMT	MYNINFMT	I	C	0	0	0	0	0	0

Note also that when the format metadata is read from the catalog, the entire format is not read. So if you have a format with a large number of ranges, this will not add execution time when running the enumeration.

Although not further discussed in this paper, there are other similar types of views. For example, there is DICTIONARY.FUNCTIONS, which performs an enumeration of SAS functions.

USING THE FMTINFO FUNCTION

You can use the FMTINFO function to obtain information about any given format. This function was first made available in SAS 9.4M3.

FMTINFO has two arguments, the format/informat name, and the information that you want returned. For example:

```
data _null_; category = fmtinfo('mmdyy','cat'); put category=; run;
```

with the result being:

```
category=date
```

The possible values for the second argument to FMTINFO are: CAT (category), TYPE (format or informat, returned as F or I), DESC (short description), or for minimum/maximum default widths or decimals: MINW, MAXW, DEFW, MIND, MAXD, DEFD. (Note that the values are not case-sensitive, but are shown here in uppercase for clarity). The possible return values for CAT are: binary, char, curr (for currency), date, datetime, time, num, smf (for IBM SMF formats like SMFSTAMP), and stat. If the format originates with PROC FORMAT, the value for CAT is unknown.

FMTINFO can be very useful if you are examining a data set for associated formats to determine what the variables might contain. For example:

```
data temp; x=today(); format x mmdyy.; run;
data _null_; set temp;
array allnums _numeric_;
do over allnums;
  varname = vname(allnums);
  fmtname = vformatn(allnums);
  if fmtinfo(fmtname, 'CAT')='date' then do;
    put varname= ' has a format of ' fmtname
      ' which is a date category format';
  end;
end;
run;
```

with the result being:

```
varname=x has a format of MMDDYY which is a date category format
```

In this example, the VFORMAT function is run on all numeric variables (of which there is just one, x). The VFORMATN function returns the name of the format associated with the variable, in this case MMDDYY, and that category will be returned as 'date'. Code can be added to perform specific functionality based on this.

POSSIBLE ENHANCEMENTS

I am always interested in hearing what users have to say about features that I have added to Base SAS, such as the FMTINFO function. As I wrote this paper, I speculated about possible enhancements to DICTIONARY.FORMATS, the FMTINFO function, and to PROC FORMAT. Would you want to see a way to add categories and descriptions to formats that originate from PROC FORMAT, that would then be available via FMTINFO? Would you want to see categories and other metadata added to DICTIONARY.FORMATS? Let me know!

CONCLUSION

DICTIONARY.FORMATS is very useful in learning about available formats, and the new FMTINFO function allows for programmatic processing of attribute information. They are useful additions to the SAS format processing feature set.

CONTACT INFORMATION

As the developer for the FMTINFO function and other format functionality, your comments and suggestions are valued and encouraged, especially regarding suggestions for future enhancements. Contact me at:

Rick Langston

SAS Institute Inc.

100 SAS Campus Drive

Cary, NC 27513

Rick.Langston@sas.com

www.sas.com

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 trademarks of their respective companies.