

Paper 086-2011

## Using Multi-label Formats to Create Subtotals in PROC TABULATE

Christina Morris, Defense Manpower Data Center, Seaside, CA

### ABSTRACT

PROC TABULATE is a procedure in SAS® that is used to create a table of statistical data. Multi-label formats can easily help create otherwise difficult subtotals in a PROC TABULATE table. In the PROC FORMAT procedure, you can create format values based on multiple values of the variable being formatted. When you define a multi-label format, you can then use the format with PROC TABULATE to provide subtotals for your data. This paper will illustrate situations where a subtotal is needed in a report. It will demonstrate how to use multi-label formats with PROC FORMAT and PROC TABULATE to create tables with subtotals in Base SAS® 9.2 in a Z/OS environment.

### INTRODUCTION

The Defense Manpower Data Center (DMDC) is a federal agency that works with large files. We often receive requests for counts of military Service Members by Service and some other variable the customer is seeking. DMDC guidelines require reports to be presented using a specific reporting style. For example, the Armed Services must be listed in a specific order, and subtotals must be generated for each Department of Defense (DoD) Service before generating a total for all Uniformed Services. By using multi-label formats along with other features in PROC TABULATE, subtotals can be created using the format. Another example for using these options might be to format States and Territories then to create a subtotal for the 50 States, a subtotal for the Territories, and a Grand Total for all.

### CREATING THE FORMAT

The syntax for creating a user defined format is as follows:

```
PROC FORMAT <library=libref.catname>;  
VALUE $FMTNAME (<options>)  
'val1' = 'formatted name 1'  
'val2' = 'formatted name 2'  
'valn' = 'formatted name n';
```

The PROC FORMAT statement is required to create user-defined formats. Multiple formats can be created within a single format procedure. It is important to remember when working with formats that without the LIBRARY=option, formats are stored in a catalog called FORMATS in the temporary WORK library so exist only for the duration of the SAS session.

The VALUE statement will define the formatted names for the input values and is required for each character or numeric format created. Format names must start with a letter or an underscore (no numeric values). The exception to this rule is that character format names must begin with a dollar sign (\$). The length of the format name can be no longer than 32 characters (including the \$ in character formats). If the input values are formatted as character, they must be enclosed in quotes. Note that many values can be mapped to a single formatted value, but a single value cannot be mapped to more than one format. The maximum length of a formatted value is 200 characters.

The VALUE statement has several options available. For the purpose of this paper, NOTSORTED and MULTILABEL will be examined.

In some cases the order of the labels is important. In the example here, the standards require that the Uniformed Services be displayed in a specific order. To accomplish this we use the NOTSORTED option in the format. This option stores values for formats in the order in which they are defined in the format. If the NOTSORTED option is not used, the values will be sorted in default order (the default is alphabetical).

The option MULTILABEL allows the assignment of multiple labels to the values from your input file. It also allows the assignment of multiple labels to the overlapped internal values. For this example the formatted label "DoD Total" will apply to all records with the values "A", "N", "F", and "M" in addition to the formatted label already defined for each of these values. When this format is used in PROC TABULATE, a separate count of DoD Total will be displayed as well as the individual service counts. Note that the MULTILABEL option will not appear in the results of PROC FREQ. If the format is applied to the variable, the first formatted label applied will be used in the frequency.

The example used in this paper creates a format for Branch of Service that will group Army, Navy, Air Force, and Marine Corps into a "DoD Total" and group all values into a "Grand Total". The code for the format is shown in the next section. The NOTSORTED option is used to retain the specified order required for reporting. The MULTILABEL option is used to create a 'subtotal' that includes the four Department of Defense Services. This method can be applied to any format when subtotals are required.

## CODE

```
PROC FORMAT;
  VALUE $FSVC (NOTSORTED MULTILABEL)
    'A' = 'Army'
    'N' = 'Navy'
    'M' = 'Marine Corps'
    'F' = 'Air Force'
    'A','N','M','F' = 'DoD Total'
    'C' = 'Coast Guard'
    'O' = 'NOAA'
    'H' = 'Public Health'
    'A','N','M','F','C','O','H' = 'Grand Total'
  ;
```

## SETTING UP PROC TABULATE

For the purpose of this example, the table created is two dimensional and only represent a count of the variables described. The tabulate will show the number of military retirees by service and gender.

There are several display options that can be used in PROC TABULATE. For the purpose of this paper only the options used in the example will be discussed in the code provide below. The first line calls the tabulate procedure as well as the data set to be used in the tabulate. The option F=COMMA11. formats the cells of the table to ensure that up to 11 digits of the number will be displayed with a comma. The option MISSING tells SAS that missing values are acceptable and to allow them in the tabulate.

The CLASS statement is where the options are defined that will allow the tabulate to work with the multi-label format and display the values in the order defined in the format. The option ORDER=DATA tells the tabulate to group the class variables by their order in the input data-set. Next the PRELOADFMT option indicates that the formats for the class variables are already loaded. This is used together with the ORDER=DATA option to ensure that the variables are displayed in the order the format has created. The MLF option indicates that the format is a multi-label format and allows for overlapping ranges to be displayed in the table. The MISSING option indicates that missing values should be displayed.

The final important piece of the tabulate is to call the formats using the FORMAT statement. All variables with a format can be called in the same FORMAT statement, and are listed with the variable followed by the format name, with a period after each format name. In this example, formats have been created for Gender and Service variables so both are included in this line of code. The KEYLABEL provides table titles for the resulting columns. Finally, the title of the table is designated.

## CODE

```
PROC TABULATE DATA=RET F=COMMA11. MISSING;
  CLASS SVC GENDER / ORDER=DATA PRELOADFMT MLF MISSING;
  TABLES (SVC='Service'), (GENDER='Gender' ALL='Total') /
  MISSTEXT='0' PRINTMISS;
  FORMAT GENDER $FGEND. SVC $FSVC.;
  KEYLABEL ALL='Total' N=' ' SUM='';
  TITLE 'Total Number of Retirees by Service and Gender';
RUN;
```

## BRINGING IT ALL TOGETHER

The final program will have two formats that are both defined under a single PROC FORMAT statement. The two formats are also defined using separate VALUE statements. The names of each of the formats begin with a dollar sign to indicate that they are both character formats. Neither format is sorted so the NOTSORTED option keeps the specific order required for the table. The format \$FSVC, also uses the MULTILABEL option to give a second format name to the group of values that are desired for the subtotal ("DoD Total").

Next, the data is read in from the external file. In this case, the external file is a fixed format flat file with each of the data elements beginning in a specific location within the record.

Finally, PROC TABULATE creates the table below displaying the counts of service members with their associated Branch of Service as row headings and Gender as column headings. The subtotal will be listed after the first four Services are displayed in the table. Note that the columns and row headings can be switched by changing the order of the SVC and Gender variables in the TABLE statement. Subtotals will then be shown across the column headings instead of in the row headings.

## CODE

```
PROC FORMAT;
  VALUE $FSVC (NOTSORTED MULTILABEL)
    'A' = 'Army'
    'N' = 'Navy'
    'M' = 'Marine Corps'
    'F' = 'Air Force'
    'A','N','M','F' = 'DoD Total'
    'C' = 'Coast Guard'
    'O' = 'NOAA'
    'H' = 'Public Health'
    'A','N','M','F','C','O','H' = 'Grand Total';

  VALUE $FGEND (NOTSORTED)
    'M' = 'Male'
    'F' = 'Female';

DATA RET;
  INFILE FILEIN;
  INPUT
    @001 SVC      $CHAR1.
    @098 GENDER  $CHAR1.
  ;

PROC TABULATE DATA=RET F=COMMA11. MISSING;
  CLASS SVC GENDER / ORDER=DATA PRELOADFMT MLF MISSING;
  TABLES (SVC='Service'), (GENDER='Gender' ALL='Total') /
  MISSTEXT='0' PRINTMISS;
  FORMAT GENDER $FGEND. SVC $FSVC.;
  KEYLABEL ALL='Total' N=' ' SUM='';
  TITLE 'Total Number of Retirees by Service and Gender';
RUN;
```

The following table is the result of this program:

	Gender		Total
	Male	Female	
Service			
Army	71,592	4,259	75,851
Navy	50,293	2,672	52,965
Marine Corps	12,047	309	12,356
Air Force	66,595	4,008	70,603
DoD Total	200,527	11,248	211,775
Coast Guard	4,135	158	4,293
NOAA	36	2	38
Public Health	406	89	495
Grand Total	205,104	11,497	216,601

## CONCLUSION

Using FORMAT and TABULATE procedures together can help create a table that is tailored to the needs of the customer. By creating a user-defined format, the data can be broken into categories that make the results more meaningful, such as subtotals for specific subgroups of the data. This paper illustrates just one of many possible situations when subtotals might be needed for a single variable. Other examples could include grouping by Counties based on City Name formats; creating Age group subtotals while still maintaining the visibility of each individual age; or any other subgroups that fit the needs of your customers.

## REFERENCES

"Base SAS(R) 9.2 Procedures Guide: Syntax: FORMAT Procedure." 2010. SAS® Knowledge Base. October 2010 <<http://support.sas.com/documentation/cdl/en/proc/61895/HTML/default/viewer.htm#a002473463.htm>>.

"Base SAS(R) 9.2 Procedures Guide: Syntax: TABULATE Procedure." 2010. SAS® Knowledge Base. October 2010 <<http://support.sas.com/documentation/cdl/en/proc/61895/HTML/default/viewer.htm#a000146761.htm>>.

"Introduction to PROC TABULATE." 2006. October 2010 <<http://www.nesug.org/proceedings/nesug06/dm/da05.pdf>>.

## RECOMMENDED READING

Bilenas, Jonas V. The Power of PROC FORMAT. SAS® Press, March 2005.

Haworth, Lauren. PROC TABULATE by Example. SAS® Press, February 1999.

## CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Christina Morris  
 Defense Manpower Data Center  
 400 Gigling Road  
 Seaside, CA 93955

Phone: (831) 583-2400 x4671  
 Fax: (831) 583-2340  
 E-mail: [christina.morris@osd.pentagon.mil](mailto:christina.morris@osd.pentagon.mil)

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.