# FASHION, STYLE "GOTTA HAVE IT" COMPUTE DEFINE BLOCK

Christopher J. Boniface, U.S. Census Bureau;

## ABSTRACT

Do you create complex reports using PROC REPORT?  Are you confused by the COMPUTE BLOCK feature of PROC REPORT?  Are you even aware of it?  Maybe you already produce reports using PROC REPORT, but suddenly your boss needs you to modify some of the values in one or more of the columns.  Maybe your boss needs to see the values of some rows in boldface and others highlighted in a stylish yellow.  Perhaps one of the columns in the report needs to display a variety of fashionable formats (some with varying decimal places and some without any decimals).  Maybe the customer needs to see a footnote in specific cells of the report.  Well, if this sounds familiar then come take a look at the CALL DEFINE statement within a COMPUTE BLOCK of PROC REPORT.

This paper will show a few tips and tricks of using the CALL DEFINE statement with conditional IF/THEN logic to make your reports stylish and fashionable.  The COMPUTE BLOCK allows you to use data step code within PROC REPORT to provide customization and style to your reports.  We'll see how the Census Bureau produces a stylish demographic profile for customers of its Special Census program using PROC REPORT with the COMPUTE BLOCK.  The paper will focus on how to use the COMPUTE DEFINE statement to create this stylish Special Census profile.  The paper will show quick tips and simple code to handle multiple formats within the same column, make the values in the total rows boldface, traffic lighting, and how to add footnotes to any cell based on the column or row. The Special Census demographic profile report is an Excel table created with the Output Delivery System (ODS) (tagsets.ExcelXP) that is stylish and fashionable thanks in part to the CALL DEFINE statement.

## INTRODUCTION

A Special Census is a basic enumeration of population, housing units, group quarters and transitory locations, conducted by the U.S. Census Bureau at the request of a governmental unit.  The Census Bureau conducts Special Censuses between decennial censuses, starting in years ending in "2" and ending in years ending in "8".  When local officials believe there has been a significant population change in their community due to growth or annexation, a Special Census may be in order.  The end products of a Special Census are Excel files that show population and housing counts by block and one-page demographic profiles for the governmental unit and associated tracts or part tracts. Standard information includes age, sex, race, Hispanic origin, occupancy or vacancy status, type of vacancy, and tenure for housing units. Most governmental units prefer to receive these data in Excel or PDF formats.

The Census Bureau tabulates and produces these reports in SAS without any post-processing or manual touch-ups. The reports are Excel files produced using PROC REPORT with ODS.  The touch-ups are accomplished with the very powerful COMPUTE BLOCK feature of PROC REPORT.  The touch-ups are the formatting, the bolding, the footnotes and the traffic-lighting of the report.  Until recently, the touch-ups were done by non-IT subject matter staff who would manually do the touch-ups in Excel after the fact.  This was cumbersome, because in many cases there would be a few dozen reports that needed these manual facelifts.  Mistakes were made and sometimes not all reports were looking their Sunday best.  Now with the power of COMPUTE BLOCK and CALL DEFINE, these touch-ups can all be done with simple BASE SAS® code.  The power of the CALL DEFINE eliminated manual touch-ups after the fact saving hours of work for the subject matter folks.  Moreover, the power of the CALL DEFINE creates a snazzy, fashionable and stylish report.  The fashion police say "gotta have it" on the CALL DEFINE.

This paper will show that we did not do any post-processing to the Excel files. We did not create any empty Excel shells beforehand that had cosmetic touch-ups applied.  I will show a few examples of simple BASE SAS® code using the COMPUTE BLOCK with the CALL DEFINE statement to add fashion and style to any report.

PROC REPORT, the COMPUTE BLOCK and the CALL DEFINE statement were first introduced in SAS® v6. The ODS with style elements and the ability to use them with CALL DEFINE were introduced in version 8.  Lastly, ExcelXP tagsets to create XML were introduced in SAS® 9.1. ExcelXP tagsets was experimental in SAS® 8.2.

## COMPUTE BLOCK/CALL DEFINE

The COMPUTE BLOCK in PROC REPORT allows you to use data step code within PROC REPORT to provide customization and style to your reports.  A COMPUTE BLOCK contains one or more programming statements that PROC REPORT executes as it builds the report.  A COMPUTE BLOCK can be associated with a report item or with a location (at the top or bottom of a report; at the top or bottom of a page; before or after a set of observations).  You create a COMPUTE BLOCK with the COMPUTE window or with the COMPUTE statement.  One form of the COMPUTE statement associates the COMPUTE BLOCK with a report item.  Another form associates the COMPUTE BLOCK with a location.  The focus of this paper will be on the former, where the COMPUTE statement associates the COMPUTE BLOCK with a report item.  Moreover, the focus of this paper is on the CALL DEFINE statement within a COMPUTE BLOCK. Here is the general syntax (BASE SAS® 9.2 Procedures Guide).

```
COMPUTE report-item </ type-specification>;
   CALL DEFINE (column-id, 'attribute-name', value);
   . . . select SAS language elements . . .
   ENDCOMP;
```

## Sample Case Study

Sheriff Andy Taylor of Mayberry, NC has requested that the Census Bureau conduct a Special Census of Mayberry. He feels that the population of Mayberry has grown in leaps and bounds since the last Census and wants a Special Census to prove it.  His deputy, Barney Fife, insists "This is big.  Really big!"  Deputy Fife wants to see a full demographic profile of Mayberry.  He demands that the demographic profile report be stylish and fashionable with boldface for all total rows, varying formats, footnotes and lots of traffic lighting.[1]

## Total Row Bolded

The customer tells you that they want to see all the total rows for each demographic characteristic in boldface, so that these totals stand out.  Have no fear, a simple COMPUTE BLOCK with a CALL DEFINE statement with conditional IF/THEN/ELSE logic and a couple of style attributes can solve this simple problem.  In our report, we have various total rows that have the keyword "total" in them.  Luckily, none of the other rows of our table have this keyword in them.   Table 1 below shows the output that we want with the total row in boldface.

**Table DP-1. Profile of General Demographic Characteristics for the Mayberry town (full), North Carolina**

Special Census of Mayberry town, North Carolina: April 1, 2016

| Subject | Special Census | |
| --- | --- | --- |
| | Number | Percent |
| **SEX AND AGE** | | |
| **Total population** | **5,360** | **100.0** |
| Under 5 years | 441 | 8.2 |
| 5 to 9 years | 332 | 6.2 |
| 10 to 14 years | 150 | 2.8 |
| 15 to 19 years | 150 | 2.8 |
| 20 to 24 years | 300 | 74.1 |

**Table 1.  Total Row in Boldface**

---

[1] All data and numbers in the tables of this paper in the Sample Case Study are fictitious.  The fictitious town of Mayberry is from the "Andy Griffith" television series

The below PROC REPORT code solves the problem.  We define the variables, number and pc (Number and Percent columns), on two DEFINE statements in the code below.  In order to get just the total rows of the table bold, we will need to add a COMPUTE BLOCK with a CALL DEFINE statement.  We will need to add one COMPUTE block for the number variable and another for the pc variable.   With a CALL DEFINE statement with a sub setting IF statement and an INDEX function on the display (or row) variable, we can define a particular style for different cells of the table.  In this case, the display variable is the name variable.  The values of the name variable are essentially the row headers in the table.  The value of the name variable for all total rows in our table, contain the word "total" in them.  Thus, we can use the INDEX function on the name variable and pass 'total' as our search string.  Thus, whenever the name variable contains the string "total", we want SAS to set a style of bold to the values of the variable(s) defined in our COMPUTE BLOCK.  Thus, the values of the Number and Percent columns in Table 1 are in bold.

```
proc report data=section1 style(header)=[fontweight=bold font=(Times, 8pt) ] ;
 column name ("Special Census" number pc);
 define name /"Subject"  f=$table1f. style(column)=[font=(TimesNewRoman, 10pt)];
 define number /'Number' style(column)=[tagattr="format:#,##0"] ;
 define pc /'Percent' style(column)=[tagattr="format:##0.0"] ;

 compute number;
   if index(name,'total') then
       call define(_col_,'style','style={fontweight=bold}');
 endcomp;

 compute pc;
   if index(name,'total') then
        call define(_col_,'style','style={fontweight=bold}');
 endcomp;
```

## Multiple Formats in the Same Column

Our demographic profile report has varying formats for values in a single column (the Number column) – some whole numbers and some with decimals.  Table 2 below shows part of our report with varying formats in the Number column.   By default, the DEFINE statement in PROC REPORT will use only one format.  Since we are creating Excel tables using ODS tagsets.ExcelXP, our DEFINE statements in PROC REPORT use tagattr formats (see "Census Retires PROC COMPUTAB" http://support.sas.com/resources/papers/proceedings13/027-2013.pdf) instead of SAS formats.  For the number column, we define a tagattr format of "format#,##0"  which is a format for a whole number.  However, a few of the rows within the Number column are displayed with one or two decimal places.   How can you get these rows to display with one or even two decimals?

| Subject | Special Census | |
|---|---|---|
| | Number | Percent |
| Homeowner vacancy rate (percent)[6] | 1.1 | (X) |
| Rental vacancy rate (percent)[7] | 7.4 | (X) |
| | | |
| **HOUSING TENURE** | | |
| **Occupied housing units** | **2,412** | **100.0** |
| Owner-occupied housing units | 1,206 | 50.0 |
| Population in owner-occupied housing units | 3,987 | (X) |
| Average household size of owner-occupied housing units | 3.14 | (X) |
| Renter-occupied housing units | 1,313 | (X) |
| Population in renter-occupied housing units | 3,777 | (X) |
| Average household size of renter-occupied units | 3.08 | (X) |

**Table 2.  Excel Table with Various Decimal Places/Formats in the same column**

Again, we can use the COMPUTE DEFINE feature of PROC REPORT to solve this.  The only way to conditionally apply a format is with a CALL DEFINE statement in a COMPUTE block.  The recommendation from SAS Institute is that on the DEFINE statement you put the format or tagattr value that applies to the most rows in the column.  Then in a COMPUTE block you overwrite/change the formats for the smaller set of rows that need to look different.  Thus, in the following COMPUTE DEFINE block, we conditionally apply a format to set the format to two decimal places when the value of the name variable contains the value "avg_hhsize".  Moreover, we set the format to one decimal place when the value of the name variable contains the value "rate".  Thus in Table 2, the values for the average household rows contain two decimal places and the values for the vacancy rate rows contain one decimal place.

```
compute number;
     if index(name,'avg_hhsize') then
       call define(_col_,"style","style={tagattr=""format:##0.00"" just=right}");
     else if index(name,'rate') then
       call define(_col_,"style","style={tagattr=""format:##0.0"" just=right}");
endcomp;
```

## Footnotes

Our demographic profile has footnotes in the Percent column for some rows where a percentage is not applicable. For these rows, a footnote of (X) is displayed in Table 2.  How can you display a footnote in the Percent column for just these rows?  Again, the solution is with a CALL DEFINE in a COMPUTE BLOCK.  For the percent column, or pc variable, we are defining a tagattr format of "format:##0.0".  This format will define most of the values in the Percent column.  However, we want to apply a SAS user-defined format to display the (X) footnote for certain rows.   The code to accomplish getting the footnote to display in the Percent column is below.  Note that the pc variable (percent column) in the underlying dataset contains missing values for the rows where a percent is not applicable.  Thus, in the format (pctf.) below, we want to display missing values with the footnote (X) in the table.

```
proc format;
  value pctf
  0-high=[11.1]
  .= '(X)';
run;

compute pc;
     if index(name,'avg_hhsize') then call define('_c3_','format','pctf.');
     if index(name,'vac_rate') then call define('_c3_','format','pctf.');
     if index(name,'rent_vac_rate') then call define('_c3_','format','pctf.');
     if index(name,'owner_occ_pop') then call define('_c3_','format','pctf.');
     if index(name,'avg_occpop') then call define('_c3_','format','pctf.');
     if index(name,'rent_pop') then call define('_c3_','format','pctf.');
     if index(name,'avg_rentpop') then call define('_c3_','format','pctf.');
endcomp;
```

In the above CALL DEFINE statements, we use _c3_ to reference the Percent column instead of _col_ as we did in previous COMPUTE BLOCKS.  The COMPUTE DEFINE statement accepts absolute column values.  In this case, the Percent column is the third column in our table.  Thus, we want to apply the SAS user-defined format, pctf., in column 3 whenever the value of the name variable contains the values specified in the INDEX functions above. Essentially, the name variable will always be missing for these rows and in order to display the footnote, we need to use the pctf. format.

## Traffic Lighting

Additionally, we can use the CALL DEFINE statement to conditionally apply "traffic lighting" to any cell of the table. Perhaps, you need to highlight some cells in red, yellow or green.  This is the case with our internal review file of our demographic profile for full Special Censuses.  Our analysts compare the Special Census counts to the counts of the governmental unit on April 1, 2010.  An important note here is that the 2010 counts are based on the current geographic boundaries of the governmental unit and not the boundaries used during the 2010 Decennial Census. This is called the Special Census base file that our analysts use to do their comparisons.  The analysts asked us to compute several extra columns showing the differences between the Special Census and the base file counts.

4

Additionally, they requested that we traffic light the final percent column to highlight cells with low, modest and high percentage differences in green, yellow, and red, respectively.  We can accomplish this using a user defined format in SAS in conjunction with a CALL DEFINE statement in PROC REPORT.  The code below shows how to accomplish traffic lighting.  Table 3 shows the traffic lighting for the percent column.

```
proc format;
value trafficf
      low - <50 ='green'
      50 - <100='yellow'
      100 - high='red';
run;

/* COMPUTE DEFINE BLOCK in PROC REPORT */
 compute pct;
  call define(_col_,"style","style=[background=trafficf.]");
endcomp;
```

**HOUSEHOLDS AND GROUP QUARTERS**

| Subject | Special Census Number | 4/1/2010 SC Base Number | Differences between SC and 2010 Numeric | Percent |
|---|---|---|---|---|
| **Total population** | 100 | 88 | 12 | 13.6 |
| In households | 88 | 80 | 10 | 10.0 |
| In group quarters | 12 | 7 | 5 | 71.4 |
| Institutionalized population | 6 | 4 | 2 | 50.0 |
| Male | 3 | 2 | 1 | 50.0 |
| Female | 3 | 2 | 1 | 50.0 |
| Noninstitutionalized population | 6 | 3 | 3 | 100.0 |
| Male | 4 | 2 | 2 | 100.0 |
| Female | 2 | 1 | 1 | 100.0 |

**Table 3.  Traffic Lighting a Column in the Excel Table**

The PROC FORMAT code above uses ranges to allow us to specify a color code for that range.  Instead of using the color code as a label in our percent column, we will use it to dynamically generate the specific background color depending on the value in the percent cell.  We can do this by using a CALL DEFINE statement within a COMPUTE BLOCK in PROC REPORT.   We use a style element on the CALL DEFINE statement to specify a background color.  Instead of specifying a particular color, we specify the user defined format (traffic.) created in the PROC FORMAT code.  Thus, any values of the pct variable less than 50 will have a green background, any value that is less than 100 but greater than or equal to 50 will have a yellow background and any value that is 100 or greater will have a red background color.

With traffic lighting, the analysts can quickly see the cells in red where there have been increases of at least 100 percent.  In our example in Table 3, the Noninstitutionalized Group Quarters population in the Mayberry Special Census has increased by 100 percent  compared to the base file. The analysts may want to note this increase in the final deliverable letter to Sheriff Andy Taylor of Mayberry.

## CONCLUSION

As we have shown, the COMPUTE DEFINE statement within a COMPUTE BLOCK of PROC REPORT can make your tables look more stylish.  This powerful feature can dress up your reports to look their Sunday best. It helped us to eliminate all post-processing and manual cosmetic touch-ups after the fact saving us many hours of time and resources.  It allowed us to add boldface and traffic lighting to any row, column or cell of a table.  Moreover, its powerful ability to display any format, anywhere, anytime sold us on it.  We showed how you can conditionally apply varying formats to the same column of the report.  Ultimately, you could apply a different format to every cell of your table.  Additionally, we showed how to conditionally add footnotes to any cell of your table.  Lastly, we showed how to accomplish traffic-lighting with the report using a user-defined SAS format with the CALL DEFINE statement in a COMPUTE BLOCK.  The power of the CALL DEFINE feature creates a snazzy and stylish report.  The word is in on

the COMPUTE BLOCK from the fashion police.  They all say "gotta have it" when it comes to the COMPUTE BLOCK/CALL DEFINE when creating reports from SAS.

## ACKNOWLEDGMENTS

## REFERENCES

BASE SAS® 9.2 Procedures Guide

Jane Eslinger, SAS Institute Inc. 2014. "The REPORT Procedure: A Primer for the COMPUTE BLOCK"
http://support.sas.com/resources/papers/proceedings15/SAS1642-2015.pdf

Andrew H. Carp, Sierra Information Services, "Traffic-Lighting Your Reports the Easy Way with PROC REPORT and ODS"
http://support.sas.com/resources/papers/proceedings11/290-2011.pdf

Christopher J. Boniface, Hung Pham, Nora Szeto, U.S. Census Bureau "Census Retires PROC COMPUTAB"
http://support.sas.com/resources/papers/proceedings13/027-2013.pdf

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

Name: Christopher J. Boniface
U.S. Census Bureau
Washington D.C. 20233
Work Phone: (301)763-5769
E-mail:  christopher.j.boniface@census.gov