

PROC REPORT in *Color ... What's Your STYLE?*

Wendy Boberg, Arkansas Foundation for Medical Care, Little Rock, Arkansas

ABSTRACT

Are you using the Output Delivery System (ODS)? Have you found the STYLE, layout and color scheme, you like? Do you know how to edit the colors on a table using ODS PDF? Do you know how to add background color to highlight a row, column, or cell? I used to get a highlighter out of my desk and go to work, but now I let SAS® and the printer do all of the work.

This paper will discuss a few different ways you can customize your reports. I will share my favorite styles and show you how to look at your tables in the different styles available. I will briefly explain how you can create your own style by changing the color scheme of an existing style so that it can be used for all of your reports. You will learn by example how to truly customize your table using PROC REPORT. I will explain how I used the pieces of code to get the table with a row, a column and individual cells highlighted with several different colors.

Keywords: REPORT, ODS, ODS PDF, STYLE, TEMPLATE, CALL DEFINE, DEFINE, BACKGROUND, STOPLIGHT

INTRODUCTION

This paper provides examples for creating PDF files using ODS, but these methods can be used with other output destinations. I will share what styles I like to use, and how you can edit styles to change the appearance of your reports. In addition, I will discuss a couple of different methods to customize reports by adding colors with the STYLE option. This paper is for beginner to intermediate level programmers with experience creating tables with the Report procedure.

THE STYLE OPTION

If you use the ODS to create a PDF file, make sure you specify a STYLE. If you do not know what styles will work for you, ask yourself these questions: 1) Is there a particular color scheme that should be used or avoided? 2) Are you only creating a stand-alone table, or are you creating a report that consists of a table and graphs? 3) If you also have a graph, should the table have the same color scheme as the graph? 4) How will the report be used? 5) Will the report be viewed electronically? 6) Will the report be printed in black and white? 7) Will the report be inserted into a presentation?

I recommend that you take some time to look at different styles to determine your favorite ones. There are several styles available. To get a list, run this PROC TEMPLATE.

```
PROC TEMPLATE;  
  LIST styles;  
RUN
```

If you have never created an ODS PDF file, try running the code below, and replace the underline with the name of the style you want to see (for the default style, either do not include the STYLE= option or specify STYLE=DEFAULT).

```
ODS LISTING CLOSE;  
ODS PDF NOTOC STYLE= _____  
  FILE="C:\TEMP\TESTING _____ STYLE.PDF";  
  TITLE1 J=C "This is the _____ STYLE";  
  FOOTNOTE1 J=L "This is just a test";  
  Insert PROC REPORT (or PROC PRINT OR PROC SQL) statement.  
ODS _ALL_ CLOSE;  
ODS LISTING;
```

If you plan to create a report that includes graphics, I recommend reading the SUGI30 paper by Jeff Carter, titled Use of Styles in Graphics. In his paper, he lists the 16 ODS-supplied styles that are ready to be applied

to SAS/GRAPH®, SAS/STAT®, and SAS/ETS®. When I create a report with graphics, I typically use the SASWEB style for the table and the STATISTICAL or ANALYSIS style for the plot; however, there are times that I will use the STATISTICAL or ANALYSIS style for both the table and the plot. These are just my preferences. There are several styles to choose from. Figures 1-3 show some examples of these styles applied to a small table I created with the Print procedure.

FIGURE 1: SASWEB

This is the SASWEB STYLE

Obs	NAME	TITLE_DEPT
1	Wendy	Statistician
2	Amy	Statistician

FIGURE 2: STATISTICAL

This is the STATISTICAL STYLE

Obs	NAME	TITLE_DEPT
1	Wendy	Statistician
2	Amy	Statistician

FIGURE 3: ANALYSIS

This is the ANALYSIS STYLE

Obs	NAME	TITLE_DEPT
1	Wendy	Statistician
2	Amy	Statistician

PROC TEMPLATE

Did you find a perfect style? If you find a style that is almost perfect but want to change a few things, like the background color, consistently in all your reports, then I recommend using the Template procedure. If the color blue is the only thing you do not like about the SASWEB style, then you can edit and create your own style with the Template procedure. To create your own style, essentially all you do is get a copy of the style's code, paste it into the editor window, and then create a new template using this code.

To get a copy of the style's code, right click on Results and select Templates. Then expand the SASHELP.TMPLMST node and select STYLES. You will see a list of all the styles available on your screen. To view the template code, double click on the name of the style. Select and copy the code you would like to edit from this window into the editor window. Below is the part of the SASWEB template style code that I will change using the Template procedure in the editor window.

```
STYLE color_list
  "Colors used in the default style" /
    'fgD1' = cx666666
    'fgC1' = cxCCCCCC
    'fgB1' = cx000000
    'bgA1' = cx6495ED
    'fgA'  = cx003399
    'bgA'  = cxffffff;
```

The 'bgA1' = cx6495ED is the background blue color, and the 'fgA' = cx003399 is the blue font color in the SASWEB style. I used the Web site <http://www.colorschemer.com/online.html> to determine what these cx colors look like on my computer. To create a new style from the SASWEB style, use the code below. Edit the cx values for a custom color combination. The code below will only change the blue font and blue background to shades of purple without changing the other colors. I selected these shades of purple from the Colorschemer Web site. This Web site is a convenient way to get HEX numbers. The Template procedure allows you to name your new custom style with the DEFINE statement. The name I chose for this new style is SASWEB_PURPLE. After running the Template procedure below, the SASWEB_PURPLE style will be available to use on your computer.

```
PROC TEMPLATE;  DEFINE STYLE Styles.SASWEB_PURPLE;
  PARENT = Styles.SASWEB;
  REPLACE color_list /
    'fgD1'=cx666666
```

```

'fgC1'=cxCCCCC
'fgB1'=cx000000
'bgA1'=cxBD64ED
'fgA'=cx660099
'bgA' =cxffffff;

end;

run;

```

Figure 4 gives an example of this SASWEB_PURPLE style as it is specified in the Template procedure above.

FIGURE 4: SASWEB_PURPLE

This is the SASWEB_PURPLE STYLE

Obs	NAME	TITLE_DEPT
1	Wendy	Statistician
2	Amy	Statistician

To learn more about the Template procedure, I recommend reading the SUGI30 paper by Lauren Haworth, titled [SAS[®] with Style: Creating your own ODS Style Template for PDF Output](#).

PROC REPORT

There are different ways to change the colors in the Report procedure. This paper only shows examples of two methods I used to create customized PDF reports with different background colors: The COMPUTE statement using the CALL DEFINE statement, and the style option in the DEFINE statement. Other than the BACKGROUND style attribute, which changes the background color, there is the FOREGROUND style attribute that will change the color of the text. Other style attributes that affect the layout, like OUTPUTWIDTH, can be edited using these methods. For more examples of how to use the Report procedure for creating PDF output, I recommend reading the SAS Global Forum 2008 Paper 033-2008 by Pete Lund, titled [PDF Can be Pretty Darn Fancy: Tips and Tricks for the ODS PDF Destination](#).

THE CALL DEFINE STATEMENT

My favorite use of the compute statement is to change the colors of rows, columns, and individual cells on my table with the CALL DEFINE statement. I will demonstrate how to change the color of a row, then a column, and then a cell. The following example is taken from a complex report I created to monitor quality measures for home health agencies. This table is on the following page under Figure 5. Below is an explanation of the code I used.

THE CALL DEFINE STATEMENT: APPLY STYLES TO ROWS

First, let us look at how to highlight the background color of a row. This example is based on a report I created, where possibly two or three rows need to be highlighted in light green (HONEYDEW). The COLUMN statement lists the order the report variables need to be processed:

```

COLUMN Add_poa Outcome Outcome_type BaselineRA month,RiskAdj ip_goal
("HHA rank" rank_May07_tie)
("STAR" star_target1_report star_target2_report start2_date);

```

The default is for each variable listed in the COLUMN statement to be displayed in the output; therefore, to keep a variable from being displayed in the DEFINE statement, use the NOPRINT option.

```

DEFINE Add_poa /NOPRINT GROUP MISSING 'Additional POA';
DEFINE Outcome /NOPRINT GROUP MISSING 'Outcome Measure';
DEFINE Outcome_type /GROUP MISSING 'Outcome Desc (Type)';

```

Use a separate COMPUTE statement for each report-item variable that you want to customize. Conditional variables that are used to change the color (highlight) of the desired rows must be listed before the COMPUTE report-item variable in the COLUMN statement. Notice that the first two variables are necessary to indicate which rows will be highlighted, and neither variable is displayed on the table. The third variable

listed is labeled 'Outcome Desc (Type)' on the table. In the CALL DEFINE statement below, _ROW_ indicates that the entire row should have the style changed.

```

COMPUTE Outcome_type;
  IF Add_poa = 'Yes'
    or Outcome in("Acute Care Hospitalization"
                  "Improvement in Management of Oral Medications")
  THEN CALL DEFINE(_ROW_, "style", "STYLE=[BACKGROUND=HONEYDEW]");
ENDCOMP;

```

THE CALL DEFINE STATEMENT: APPLY STYLES TO COLUMNS

Now let us look at how to highlight the background color of the columns. This example is based on a report where the background colors of the columns are changed for different sections of the table. The variable ip_goal is a light pink (LAVENDERBLUSH); the variable for the rank is white (SNOW), which needs to be specified since I do not want any of the rows to be green in this column; and the last section of the table labeled STAR is light blue (ALICEBLUE). In the CALL DEFINE statement, _COL_ indicates the entire column should have the style changed:

```

COMPUTE ip_goal;
  CALL DEFINE(_COL_, "style", "STYLE=[BACKGROUND=LAVENDERBLUSH]");
ENDCOMP;
COMPUTE rank_May07_tie;
  CALL DEFINE(_COL_, "style", "STYLE =[BACKGROUND=SNOW]");
ENDCOMP;
COMPUTE star_target1_report;
  CALL DEFINE(_COL_, "style", "STYLE =[BACKGROUND=ALICEBLUE]");
ENDCOMP;
COMPUTE star_target2_report;
  CALL DEFINE(_COL_, "style", "STYLE =[BACKGROUND=ALICEBLUE]");
ENDCOMP;
COMPUTE start2_date;
  CALL DEFINE(_COL_, "style", "STYLE =[BACKGROUND=ALICEBLUE]");
ENDCOMP;

```

Figure 5 has two rows highlighted in light green and the last five columns highlighted in light pink, white, and light blue, using the compute blocks shown above.

FIGURE 5: CHANGE COLOR OF ROWS AND COLUMNS

Good Health HHA (047xxx) Risk Adjusted Rates (%)																			
		Period Ending																	
		JUL06	AUG06	SEP06	OCT06	NOV06	DEC06	JAN07	FEB07	MAR07	APR07	MAY07			HHA rank	STAR			
Outcome Desc (Type)	Jul04 Baseline	%	%	%	%	%	%	%	%	%	%	%	Goal	in AR May07		1st Target	2nd Target	Date Set	2nd
Acute Care Hospitalization (Risk Adjusted)	33.42	35.98	35.79	35.44	35.76	35.40	34.47	34.97	35.77	34.97	33.95	35.08	28.21	0		30.00	28.50	01/24/07	
Any Emergent Care (Risk Adjusted)	23.09	22.24	22.08	21.12	21.72	21.63	21.86	22.26	23.23	22.85	22.48	23.59		0					
Discharge to Community (Risk Adjusted)	62.54	58.57	58.40	58.70	58.49	58.94	60.21	59.75	58.91	59.62	60.47	59.75	72.15	0					
Improvement in Ambulation/Locomotion (Risk Adjusted)	34.94	36.96	37.53	37.73	37.27	38.00	38.93	38.64	37.27	37.63	38.36	40.06	45.95	0					
Improvement in Bathing (Risk Adjusted)	57.29	56.25	56.12	56.39	55.98	56.55	56.61	55.79	55.31	56.19	55.29	57.93	68.41	0					
Improvement in Dyspnea (Risk Adjusted)	44.01	43.83	43.67	43.64	44.71	45.09	46.08	45.43	45.02	47.40	49.34	53.10	62.86	0					
Improvement in Management of Oral Medications (Risk Adjusted)	32.58	45.68	45.80	45.75	46.11	48.05	47.85	47.59	48.39	47.32	42.58	46.75	42.92	0		37.00	40.00	01/24/07	
Improvement in Pain Interfering with Activity (Descriptive)	55.66	57.95	59.51	60.16	60.60	60.98	59.44	60.31	61.46	62.33	63.02	62.22	69.74	0					
Improvement in Status of Surgical Wounds (Descriptive)	78.94	82.05	81.86	82.31	81.96	82.29	82.47	82.72	80.60	84.16	82.61	83.39	86.94	0					
Improvement in Transferring (Risk Adjusted)	46.12	51.80	49.61	49.11	48.42	48.23	48.04	47.98	48.04	50.21	50.05	51.95	59.73	0					
Improvement in Urinary Incontinence (Risk Adjusted)	30.80	33.88	33.63	34.05	35.28	35.06	35.45	36.12	37.22	38.48	36.88	39.14	50.93	0					

THE CALL DEFINE STATEMENT: APPLY STYLES TO CELLS

What if you decide not to highlight the last three columns in light blue, but want to highlight cells in these three columns depending on their values? Use the CALL DEFINE statements shown above, where the _COL_ indicates the entire column should have the style changed, but add a conditional statement in the COMPUTE block so that the style of a cell is changed only if that cell meets the requirement specified. To change the BACKGROUND style of a non-blank cell to light blue, use the code below.

```
COMPUTE star_target1_report;  
  IF star_target1_report ^= ''  
  THEN CALL DEFINE(_COL_, "style", "STYLE=[BACKGROUND=ALICEBLUE]");  
ENDCOMP;  
COMPUTE star_target2_report;  
  IF star_target2_report ^= ''  
  THEN CALL DEFINE(_COL_, "style", "STYLE=[BACKGROUND=ALICEBLUE]");  
ENDCOMP;  
COMPUTE start2_date;  
  IF start2_date ^= ''  
  THEN CALL DEFINE(_COL_, "style", "STYLE=[BACKGROUND=ALICEBLUE]");  
ENDCOMP;
```

THE CALL DEFINE STATEMENT: APPLY STYLES TO ROWS AND CELLS IN COLUMNS

Take a simple table, such as the one in Figure 4, and update it with more staff members. Highlight new rows and change the font style in the second column if the title information was recently updated. The resulting table is in Figure 6 below.

**FIGURE 6: CHANGING BACKGROUND STYLE FOR ADDITIONAL ROWS
AND FONT STYLE FOR UPDATED COLUMNS**

This is the SASWEB_PURPLE STYLE

Analytic Staff	Title
Amy	<i>Statistician - Survey Team Leader</i>
Judy	Statistical Analyst
Marlo	<i>Director of Analytical Services</i>
Mayumi	Data Analyst
Melanie	Statistician
Teresa	Statistical Analyst
Toby	Statistician
Wendy	<i>Statistician - Data Mining Team Leader</i>

The code used to create a PDF file with this table is below. Look at the two compute blocks that were added to the Report procedure: one for the variable that determines whether the row should be highlighted (name) and one for the variable that determines whether the column should have a different font style and font weight (title).

```
ODS LISTING CLOSE;  
ODS PDF NOTOC STYLE=SASWEB_PURPLE FILE="C:\TEMP\Figure6 example.pdf";  
  
Title1 j=c "This is the SASWEB_PURPLE STYLE";  
Footnote1 j=l "This is just a test";  
  
PROC REPORT DATA=person NOWD;  
  COLUMNS name title;  
    DEFINE name /order "Analytic Staff";  
    DEFINE title /display "Title";  
  COMPUTE name;  
    IF name not in("Wendy" "Amy")  
    THEN CALL DEFINE(_ROW_, "style",  
                     "STYLE=[BACKGROUND=cxF7EDFD]");  
  ENDCOMP;
```

```

ENDCOMP;
COMPUTE title;
    IF title in("Statistician - Data Mining Team Leader"
               "Statistician - Survey Team Leader"
               "Director of Analytical Services")
    THEN CALL DEFINE(_COL_, "style",
                    "STYLE=[FONT_STYLE=italic
                        FONT_WEIGHT=bold]");
ENDCOMP;
RUN;

ODS _ALL_ CLOSE;
ODS LISTING;

```

THE CALL DEFINE STATEMENT: APPLY STYLES TO CELLS (CONTINUED)

In Figure 7, instead of highlighting a few rows all in one color, each cell with a monthly rate is highlighted depending on a performance variable (perform). I call Figure 7 my STOPLIGHT report, since a cell is colored red if the rate is worse than baseline, yellow if the rate is better than baseline but has not met the goal yet, and green if the rate is better than the goal.

Notice in the COLUMN statement that the ACROSS variable (month) is followed by a comma and the variable(s) with values that depend on each unit of the ACROSS variable are listed after the comma. As in this example, when there are multiple variables, list all of them in parentheses after the comma. The months are displayed left to right across the page, the rate (%) is the value of the RiskAdj variable displayed in the cells, and the background color of these cells depend on the value of the perform variable. The COLUMN statement used for this STOPLIGHT example is shown here.

```

COLUMN Add_poa Outcome Outcome_type BaselineRA month,(perform RiskAdj)
         ip_goal
         ("HHA rank" rank_May07_tie)
         ("STAR" star_target1_report star_target2_report start2_date);

```

Notice that the conditional variable needed in the COMPUTE statement (perform) is listed before the COMPUTE report-item variable (RiskAdj). The order in the COLUMN statement is important, since it specifies the order that the Report procedure processes the data. This example is more complex since the variable Riskadj is under the ACROSS variable (month). Therefore, the CALL DEFINE statement will not have the generic _COL_, which indicates that the style is to be applied to the RiskAdj variable column. There is more than one column with rates. There is a set of columns for each of the two variables listed under the ACROSS variable (perform and RiskAdj) for each value of the ACROSS variable (month). In the example code below, the variables used in the Compute blocks to determine what style is applied are not printed in the table. Note that all these have the NOPRINT option in the DEFINE statement, but I do recommend checking the style by printing the variables (do not use the NOPRINT option until you know that your code is working correctly).

```

DEFINE Add_poa /NOPRINT GROUP MISSING 'Additional POA';
DEFINE Outcome /NOPRINT GROUP MISSING 'Outcome Measure';
DEFINE Outcome_type /GROUP MISSING 'Outcome Desc (Type)';
DEFINE BaselineRA /GROUP MISSING 'Jul04 Baseline' FORMAT=4.2;
DEFINE month /ACROSS 'Period Ending' ORDER=INTERNAL FORMAT=MONYY5.;
DEFINE perform /NOPRINT GROUP MISSING 'stoplight colors' ;
...

```

The Report procedure only processes the data cell by cell in the order specified; therefore, the basic IF-THEN logic using the name of the variable (perform) to condition on cannot be used in the COMPUTE block. A unique IF-THEN conditional statement will be needed for each column displayed under the ACROSS variable (month). There also needs to be a conditional statement for each possible background color. Use _Cxx_ to identify the column, where xx is the column number. Use the COLUMN statement to determine column numbers. Count every variable in the COLUMN statement, except the ACROSS variable, whether or not it is displayed in the report. In this example, the first set of conditional statements reference columns 5 (for the perform value) and column 6 (for the RiskAdj value) under the first month value (JUL06), and the second set of conditional statements reference columns 7 and 8 for the second month value (AUG06). The pattern is continued for all columns created under the ACROSS variable. The last set of conditional statements in this compute block reference columns 25 and 26 for the column labeled MAY07. The colors used in this Stoplight report are red (lightRED), yellow (cxFFFF99), green (lightGREEN) or white (SNOW) depending on the value of the perform variable.

```

COMPUTE RiskAdj;
  IF _c5_="WORSE THAN BASELINE"
  THEN CALL DEFINE("_c6_", "style", "STYLE=[BACKGROUND=lightRED]");
  IF _c5_="GOAL NOT MET"
  THEN CALL DEFINE("_c6_", "style", "STYLE=[BACKGROUND=cxFFFF99]");
  IF _c5_="GOAL MET"
  THEN CALL DEFINE("_c6_", "style", "STYLE=[BACKGROUND=lightGREEN]");
  IF _c5_ in("NO DATA" "")
  THEN CALL DEFINE("_c6_", "style", "STYLE=[BACKGROUND=SNOW]");

  IF _c7_="WORSE THAN BASELINE"
  THEN CALL DEFINE("_c8_", "style", "STYLE=[BACKGROUND=lightRED]");
  IF _c7_="GOAL NOT MET"
  THEN CALL DEFINE("_c8_", "style", "STYLE=[BACKGROUND=cxFFFF99]");
  IF _c7_="GOAL MET"
  THEN CALL DEFINE("_c8_", "style", "STYLE=[BACKGROUND=lightGREEN]");
  IF _c7_ in("NO DATA" "")
  THEN CALL DEFINE("_c8_", "style", "STYLE=[BACKGROUND=SNOW]");

  ...

  IF _c25_="WORSE THAN BASELINE"
  THEN CALL DEFINE("_c26_", "style", "STYLE=[BACKGROUND=lightRED]");
  IF _c25_="GOAL NOT MET"
  THEN CALL DEFINE("_c26_", "style", "STYLE=[BACKGROUND=cxFFFF99]");
  IF _c25_="GOAL MET"
  THEN CALL DEFINE("_c26_", "style", "STYLE=[BACKGROUND=lightGREEN]");
  IF _c25_ in("NO DATA" "")
  THEN CALL DEFINE("_c26_", "style", "STYLE=[BACKGROUND=SNOW]");
ENDCOMP;

```

Figure 7 below has the STOPLIGHT colors described above in the columns under the Period Ending Section (columns 6, 8, 10, 12, 14, 16, 18, 20, 22, 24 and 26 in this example). Notice that 'Period Ending' is the label for the across variable month in the DEFINE statement.

FIGURE 7: CHANGE COLOR OF CELLS

Good Health HHA (047xxx)																	
Risk Adjusted Rates (%)																	
Column: 3	4	6	8	10	12	14	16	18	20	22	24	26					
		Period Ending															
		JUL06	AUG06	SEP06	OCT06	NOV06	DEC06	JAN07	FEB07	MAR07	APR07	MAY07		HHA rank	STAR		
Outcome Desc (Type)	Jul04 Baseline	%	%	%	%	%	%	%	%	%	%	%	Goal	in AR May07	1st Target	2nd Target	Date Set 2nd
Acute Care Hospitalization (Risk Adjusted)	33.42	35.98	35.79	35.44	35.76	35.40	34.47	34.97	35.77	34.97	33.95	35.08	28.21	0	30.00	28.50	01/24/07
Any Emergent Care (Risk Adjusted)	23.09	22.24	22.08	21.12	21.72	21.63	21.88	22.25	23.23	22.85	22.48	23.59	-	0			
Discharge to Community (Risk Adjusted)	62.54	58.57	58.40	58.70	58.49	58.94	60.21	59.75	58.91	59.62	60.47	59.75	72.15	0			
Improvement in Ambulation/Locomotion (Risk Adjusted)	34.94	36.96	37.53	37.73	37.27	38.00	38.83	38.64	37.27	37.63	38.36	40.08	45.95	0			
Improvement in Bathing (Risk Adjusted)	57.29	56.25	56.12	56.39	55.98	56.55	56.81	55.79	56.31	56.19	55.29	57.93	68.41	0			
Improvement in Dyspnea (Risk Adjusted)	44.01	43.83	43.67	43.64	44.71	45.09	46.08	45.43	45.02	47.40	49.34	53.10	62.86	0			
Improvement in Management of Oral Medications (Risk Adjusted)	32.58	45.88	45.80	45.75	46.11	48.05	47.85	47.59	48.39	47.32	42.58	46.75	42.92	0	37.00	40.00	01/24/07
Improvement in Pain Interfering with Activity (Descriptive)	55.66	57.95	59.51	60.16	60.60	60.98	59.44	60.31	61.46	62.33	63.02	62.22	69.74	0			
Improvement in Status of Surgical Wounds (Descriptive)	78.94	82.05	81.86	82.31	81.96	82.29	82.47	82.72	80.00	84.16	82.61	83.39	86.94	0			
Improvement in Transferring (Risk Adjusted)	46.12	51.80	49.61	49.11	49.42	48.23	48.04	47.99	48.04	50.21	50.05	51.95	59.73	0			
Improvement in Urinary Incontinence (Risk Adjusted)	30.80	33.88	33.63	34.05	35.28	35.05	35.45	36.12	37.22	38.48	38.88	39.14	50.93	0			

THE DEFINE STATEMENT: APPLY STYLES TO CELLS

Another way to change the background color of an individual cell is to use the STYLE option in the DEFINE statement of PROC REPORT. To do this you will need to create a user-defined format.

Here are two example formats. The first one, called color_fmt, is for a variable that has a Yes/No response coded as text (either a Y or N), and the plan is to change the background color to red if the response is 'N'. The second format defined below is for a numeric variable that is a ratio. For this one, the plan is to change the background color depending on the value of the variable.

```
PROC FORMAT;
  VALUE $color_fmt
    "Y" = "SNOW"
    "N" = "RED";

  VALUE c_ratio
    LOW  -< 0 = 'RED'
    0    -< 1 = 'YELLOW'
    1    - HIGH = 'GREEN'
    OTHER = 'SNOW';

RUN;
```

As the user-defined format c_ratio is written above, the color red would be for values that are less than zero, the color yellow would be for values greater than or equal to zero and less than one, and the color green would be for values greater than or equal to one. Note that if any large values are not acceptable, then the HIGH option should not be used, and if any small (including negative) numbers are not acceptable, then the LOW option should not be used. You might want to add a color for errors (unreasonable values), but this would take away from the stoplight effect. The color snow, which is a white, would be for any value not included in the other conditions (a missing value in this example).

Unlike the previous example, where the color applied to a variable is dependent on another variable, when using the DEFINE statement, the color depends on the value of the DEFINE report-item variable.

```
DEFINE flagifnotsame /DISPLAY MISSING CENTER "Same?"
                     STYLE={BACKGROUND=$color_fmt.};

DEFINE ratio          /DISPLAY MISSING          "Ratio"
                     STYLE={BACKGROUND=c_ratio.};
```

Only the cells in the column of the DEFINE variable that have the STYLE option will be affected by the attribute set to a user-defined format. This can be seen in Figures 8 and 9 below.

FIGURE 8: CHANGE COLOR OF CELLS BASED ON USER-DEFINED FORMAT \$COLOR_FMT

HIGHLIGHT RED WHEN CHANGE WAS MADE

Analytic Staff	Same?
Amy	Y
Judy	N
Marlo	N
Mayumi	N
Melanie	N
Teresa	N
Toby	N
Wendy	Y

FIGURE 9: CHANGE COLOR OF CELLS BASED ON USER-DEFINED FORMAT C_RATIO

HIGHLIGHT RED IF RATIO LESS THAN ZERO,
YELLOW IF LESS THAT ONE (GREATER THAN OR EQUAL TO ZERO),
and GREEN IF GREATER THAN OR EQUAL TO ONE

Analytic Staff	Ratio
Amy	0.5
Judy	10
Marlo	1
Mayumi	0
Melanie	0.75
Teresa	0.25
Toby	.
Wendy	-0.5

Use the STYLE options in the DEFINE statement of the Report procedure to highlight cells that are of interest by changing the background color (BACKGROUND), the font including the color (FONT_STYLE, FONT_WEIGHT, FOREGROUND ...), and even the column width (OUTPUTWIDTH). If the style you want to apply depends on the value of your variable, then a user-defined format is needed. In this next example, I have added bold font to the values in red and green and made the yellow values italic. The user-defined format is below:

```
PROC FORMAT;
  VALUE Fw_ratio
    LOW -< 0 = 'BOLD'
    1 - HIGH = 'BOLD';

  VALUE FS_ratio
    0 -< 1 = 'ITALIC';
RUN;
```

Notice how these user-defined formats are being utilized in the style option of the DEFINE statement in the Report procedure below. Multiple style attributes can be used.

```
ODS LISTING CLOSE;
ODS PDF NOTOC STYLE=SASWEB_PURPLE
  FILE="C:\TEMP\DEFINE STATEMENT EXAMPLE 10.pdf";

PROC REPORT DATA=CKDAT NOWD;
  COLUMNS name ratio;

  DEFINE name /order "Analytic Staff" CENTER STYLE={OUTPUTWIDTH=10%};

  DEFINE ratio /DISPLAY MISSING "Ratio" CENTER STYLE={BACKGROUND=c_ratio.
    FONT_WEIGHT=FW_RATIO.
    FONT_STYLE=FS_RATIO.
    OUTPUTWIDTH=5%};

RUN;

ODS _ALL_ CLOSE;
ODS LISTING;
```

The resulting table in Figure 10 (on the next page) has the same background color as the table in Figure 9 above, but the font weight and font style are different. The column width is also different, since the OUTPUTWIDTH style attribute was used.

**FIGURE 10: CHANGE COLOR OF CELLS BASED ON USER-DEFINED FORMAT C_RATIO,
AND CHANGE FONT OF SOME CELLS BASED ON THE USER-DEFINED FORMATS
FW_RATIO AND FS_RATIO**

**HIGHLIGHT RED AND BOLD IF RATIO LESS THAN ZERO,
YELLOW AND ITALIC IF LESS THAN ONE (GREATER THAN OR EQUAL TO ZERO),
GREEN AND BOLD IF GREATER THAN OR EQUAL TO ONE**

Analytic Staff	Ratio
Amy	0.5
Judy	10
Marlo	1
Mayumi	0
Melanie	0.75
Teresa	0.25
Toby	.
Wendy	-0.5

CONCLUSION

If your resources allow for the use of color in either printed or electronic format, I challenge you to find the STYLE(s) that works best for you. The colors used in this paper display as expected in the PDF files I created, but if you are using another ODS output designation like HTML, then you are more limited in the colors that you can use. Always verify that your color choices will be displayed as expected before finalizing your code. If you get a gray where another color is expected then select another color since the one you specified could not be displayed. I prefer to customize the background color, but there are other ways to highlight information in a table. Other Style attributes like FOREGROUND can be used. Now go show everyone what your STYLE looks like!

REFERENCES

Carter, Jeff. 2005. "Use of Styles in Graphics." Proceedings of the thirtieth annual SAS® Users Group International Conference, Philadelphia PA. <http://support.sas.com/rnd/datavisualization/papers/sugi30/GraphStyles.pdf>

Code extract from SAS 9.1.3 Service Pack 4 Copyright (c) 2002-2003 by SAS Institute Inc., Cary, NC, USA.

Colorschemer Online v.2. <http://www.colorschemer.com/online.html>. Accessed 12/28/2006.

Haworth, Lauren. 2005. "SAS® with Style: Creating your own ODS Style Template for PDF Output." Proceedings of the thirtieth annual SAS Users Group International Conference, Philadelphia PA, 132-30. <http://www2.sas.com/proceedings/sugi30/132-30.pdf>

Lund, Pete. 2008. "PDF Can be Pretty Darn Fancy: Tips and Tricks for the ODS PDF Destination." Proceedings of the SAS Global Forum 2008, San Antonio, TX, 033-2008. <http://www2.sas.com/proceedings/forum2008/033-2008.pdf>

The data displayed in the examples are fictitious and should only be used for report layout examples.

ACKNOWLEDGMENTS

Special thanks to everyone at AFMC for all their helpful advice and support.

The content of this publication does not necessarily reflect the views or policies of Arkansas Foundation for Medical Care (AFMC). The author assumes full responsibility for the accuracy and completeness of the ideas presented.

RECOMMENDED READING

For more information about quality measures in the home health setting, go to the [Home Health Compare](http://www.medicare.gov/HHCompare/Home.asp?dest=NAV|Home|About#TabTop) Web site. <http://www.medicare.gov/HHCompare/Home.asp?dest=NAV|Home|About#TabTop>

CONTACT INFORMATION

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

Wendy Boberg
Statistician, Data Mining Team Leader
Arkansas Foundation for Medical Care (AFMC)
401 W Capitol Ave., Suite 410
Little Rock, AR 72201
Phone: 501-212-8716
Fax: 501-375-1201
E-mail: wboberg@afmc.org
Web: <http://www.afmc.org>

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.