

Building a Better Dashboard Using Base SAS® Software

Kirk Paul Lafler, Software Intelligence Corporation, Spring Valley, California

Joshua M. Horstman, Nested Loop Consulting, Indianapolis, Indiana

Roger D. Muller, Data-To-Events, Indianapolis, Indiana

Abstract

Organizations around the world develop business intelligence dashboards, sometimes referred to as enterprise dashboards, to display the current status of “point-in-time” metrics and key performance indicators. Effectively designed dashboards extract real-time data from multiple sources for the purpose of highlighting important information, numbers, tables, statistics, metrics, performance scorecards and other essential content on a single screen. This presentation explores the basic rules for “good” dashboard design, the metrics frequently used in dashboards, and the use of best practice programming techniques in the design of highly interactive, filterable, and drill-down dashboards using Base-SAS® software. Attendees learn how to create a real-world dashboard using Base-SAS® programming techniques including the use of the DATA step, PROC FORMAT, PROC PRINT, PROC MEANS, PROC SQL, Enterprise Guide, ODS, ODS Statistical Graphics, PROC SGRENDER, PROC SGPLOT, PROC SGSCATTER, PROC SGPANEL, and PROC TEMPLATE.

Introduction

In a world of big data where data repositories and the demand placed on them are growing at explosive levels, organizations are faced with a number of decisions related to their information requirements:

- 1) What are the best ways to handle large amounts of information?
- 2) How should analytical data be processed?
- 3) What are the choices for constructing the most effective information delivery mechanisms?
- 4) How should analytical data and results be displayed?

To help answer these and other questions, this paper explains what a dashboard is, the dashboard’s elements, the do’s and don’ts for constructing effective dashboards, dashboard design techniques, an investigation of the various types of dashboards, the merits and strengths of using the base-SAS® software to construct dashboards, and an illustration of a few dashboard examples along with the base-SAS code used in their construction.

Example Table

The dashboard examples displayed in this paper reference a Movies table consisting of a number of movies that I’ve viewed over the years. The Movies table consists of six columns: Title, Length, Category, Year, Studio, and Rating. Title, Category, Studio, and Rating are defined as character columns, and the Length and Year are defined as numeric columns. The data contained in the Movies table is illustrated below.

Movies Table

	Title	Length	Category	Year	Studio	Rating
1	Brave Heart	177	Action Adventure	1995	Paramount Pictures	R
2	Casablanca	103	Drama	1942	MGM / UA	PG
3	Christmas Vacation	97	Comedy	1989	Warner Brothers	PG-13
4	Coming to America	116	Comedy	1988	Paramount Pictures	R
5	Dracula	130	Horror	1993	Columbia TriStar	R
6	Dressed to Kill	105	Drama Mysteries	1980	Filmways Pictures	R
7	Forrest Gump	142	Drama	1994	Paramount Pictures	PG-13
8	Ghost	127	Drama Romance	1990	Paramount Pictures	PG-13
9	Jaws	125	Action Adventure	1975	Universal Studios	PG
10	Jurassic Park	127	Action	1993	Universal Pictures	PG-13
11	Lethal Weapon	110	Action Cops & Robber	1987	Warner Brothers	R
12	Michael	106	Drama	1997	Warner Brothers	PG-13
13	National Lampoon's Vacation	98	Comedy	1983	Warner Brothers	PG-13
14	Poltergeist	115	Horror	1982	MGM / UA	PG
15	Rocky	120	Action Adventure	1976	MGM / UA	PG
16	Scarface	170	Action Cops & Robber	1983	Universal Studios	R
17	Silence of the Lambs	118	Drama Suspense	1991	Orion	R
18	Star Wars	124	Action Sci-Fi	1977	Lucas Film Ltd	PG
19	The Hunt for Red October	135	Action Adventure	1989	Paramount Pictures	PG
20	The Terminator	108	Action Sci-Fi	1984	Live Entertainment	R
21	The Wizard of Oz	101	Adventure	1939	MGM / UA	G
22	Titanic	194	Drama Romance	1997	Paramount Pictures	PG-13

“Brief” History of Dashboards

In the world of information technology, a dashboard serves as a user interface to organize and display information visually in the simplest way possible. Dashboards originated in the 1970’s as decision support tools and systems that served management, operations, and organizational planning. In the 1980’s, dashboards came of age as executive information systems emphasizing graphical displays and simple user interfaces to assist with management decision making. In the 1990’s, dashboards experienced a growing interest with the rise of the Internet. As information technology and the Internet entered the 2000’s, vendors including SAS Institute, and others, offered high-end easy-to-use products for the development of comprehensive “custom” dashboards. The dashboards being built today offer users the ability to monitor key metrics, information summaries, and reports in a single easy-to-use user interface. As a result, dashboards are designed to alert users to key business issues that impact an organization’s tactics and strategies by facilitating improved decision making activities.

So exactly what is a dashboard? In the paper, “Building Your First Dashboard Using the SAS® 9 Business Intelligence Platform: A Tutorial,” by Gregory S. Nelson (2009), Nelson describes a dashboard as a visualization technique that provides an immediate view or snapshot of exactly where you are in a specific process relative to your stated goals and objectives. He adds that, Visual indicators, such as temperature gauges, traffic lights and speedometers, help give a real-world sense of present progress and assists in making decisions, adapting to current conditions or drilling into more detailed information. As a user interface, dashboards display performance indicators (PIs), key performance indicators (KPIs), and other relevant information.

Types of Dashboards

The first step in dashboard design is to understand the purpose and type of dashboard you will need. With three types of dashboard designs available, users are encouraged to select the dashboard type that best meets your needs. The following table describes the three types of enterprise dashboards and their purpose.

Dashboard Type	Purpose
Strategic Dashboards	Strategic dashboards provide executives and managers with visual information to determine and support goals and objectives within an organization. This type of dashboard facilitates monitoring an organization’s health, progress, performance, and areas where improvement can be made. There is typically no need for interactive features with this type of dashboard. Strategic dashboard examples include: Sales, Human Resources, Manufacturing, and Services.
Analytical Dashboards	Analytical dashboards provide users with visual information to help gain a better understanding with historical, present and future data; understand trends; allow comparisons to be made; and determine the type of adjustments that are needed. Analytical dashboards should allow interactive features such as drill-down capabilities, as needed, to access more detailed information. Dashboard examples include: obtaining real-time data and information, determining why some things are working and others are not, identifying patterns and opportunities with your data, and aligning strategic objectives with performance initiatives.
Operational Dashboards	Operational dashboards provide users with visual information to concentrate on performance monitoring and measurements, monitor the efficiency and effectiveness of their organization. There is typically a need to update information displayed in an operational dashboard frequently to make it relevant to the users’ needs. Dashboard examples include: improved understanding of performance, better focus and alignment, and faster and better decision making.

Dashboard Elements

In Malik Shadan’s (2007) paper, Elements for an Enterprise Dashboard, he mentions that there are basic and advanced characteristics specific to an enterprise dashboard. The basic characteristics encompass the acronym, SMART, and the advanced characteristics of an enterprise dashboard encompass the acronym, IMPACT. The elements associated with each acronym appear in the following tables.

SMART Basic Elements	
	Description
Synergetic	Synergize information in a single screen view.
Monitor KPIs	Display critical KPIs for effective decision making.
Accurate	Dashboard must be well tested and validated, and information must be accurate.
Responsive	Respond to user alerts and visual content to draw immediate attention to critical matters.
Timely	Display information that is real-time and right-time for effective decision making.

IMPACT Advanced Elements

	Description
Interactive	Allow user to drill-down and derive details, root causes and more.
More Data History	Allow users to review historical trends for any KPI.
Personalized	Display should be specific to each user's domain of responsibility, data restrictions, and privileges.
Analytical	Allow users to perform guided analysis, compare, contrast, and make analytical inferences.
Collaborative	Facilitate users' ability to exchange notes regarding observations on their dashboard.
Trackability	Allow each user to customize the metrics they would like to track.

13 Common Pitfalls to Avoid when Designing Dashboards

Successful dashboard design involves the transformation of quantitative data into meaningful and effective visual displays including graphs, maps, gauges and summary information. In his paper, "Common Pitfalls in Dashboard Design," Stephen Few (2006) proposes 13 common mistakes many make when designing dashboards. Instead of concentrating on what should be done when designing dashboards, Mr. Few's body of work espouses the most common mistakes along with detailed explanations to help educate current and future designers alike. I have listed the 13 common pitfalls from Mr. Few's seminal work, below, but readers are encouraged to read his entire paper, see the References section, for a complete perspective.

Stephen Few's 13 Common Pitfalls in Dashboard Design (cited from reference)

Pitfall	Description
Pitfall #1	Exceeding the Boundaries of a Single Screen
Pitfall #2	Supplying Inadequate Context for the Data
Pitfall #3	Displaying Excessive Detail or Precision
Pitfall #4	Expressing Measures Indirectly
Pitfall #5	Choosing Inappropriate Media of Display
Pitfall #6	Introducing Meaningless Variety
Pitfall #7	Using Poorly Designed Display Media
Pitfall #8	Encoding Quantitative Data Inaccurately
Pitfall #9	Arranging the Data Poorly
Pitfall #10	Ineffectively Highlighting What's Important
Pitfall #11	Cluttering the Screen with Useless Decoration
Pitfall #12	Misusing or Overusing Color
Pitfall #13	Designing an Unappealing Visual Display

Steps to Creating a Dynamic Dashboard using Base-SAS®

Follow these basic steps to successfully construct a dynamic dashboard using the Base-SAS software.

1. Connect to desired data sources using Libname statement.
2. Create user-defined formats containing URL links for dashboard and drill-down results.
3. Create Graphics, Bar Chart, Box Plot, Histogram, Pie, etc.
4. Produce Graph Template using PROC SGRENDER as a static or dynamic display.
5. Produce Drill-down Detail Reports, Summary Reports, Statistics, Tables, etc.

A number of static and interactive drill-down dashboard examples are illustrated, below.

Dashboard Examples

1. Static Dashboard – PROC PRINT Dashboard using PROC FORMAT



Key Points about Code

1. Base SAS provides users with several procedures for creating graphical output. The three procedures available are: SGPLOT, SGSCATTER and SGPANEL.
2. The type of dashboard created in this approach is a static, or non-drill-down, dashboard.
3. An ODS HTML statement tells SAS where to produce the horizontal bar chart.
4. An ODS GRAPHICS statement creates a “PNG” image and defines the dashboard’s dimensions.
5. PROC SGPLOT is specified to produce the horizontal bar chart.

Base-SAS Code:

```

*****
**** Program Name: Static Dashboard with PROC SGPLOT.SAS *****
**** Purpose.....: Create and display a simple dashboard interface using PROC *****
****                SGPLOT. *****
**** *****
**** Author.....: Kirk Paul Lafler, Software Intelligence Corporation *****
**** Date Written: 08/16/2017 *****
**** SAS Version.: SAS 9.4 *****
**** Input Files.: Workshop Data *****
****                Movies *****
**** Output Files: HTML Output (1 File) *****
****                Example 1 Static Dashboard.htm *****
**** Subroutines.: None *****
**** User-defined Formats: None *****
**** Macro Variables: None *****
**** Includes....: None *****
**** Modification History: *****
**** 08/16/2017 KPL Re-engineered code to work with PROC SGPLOT. *****
**** 02/23/2015 KPL Added Header information. *****
*****

```

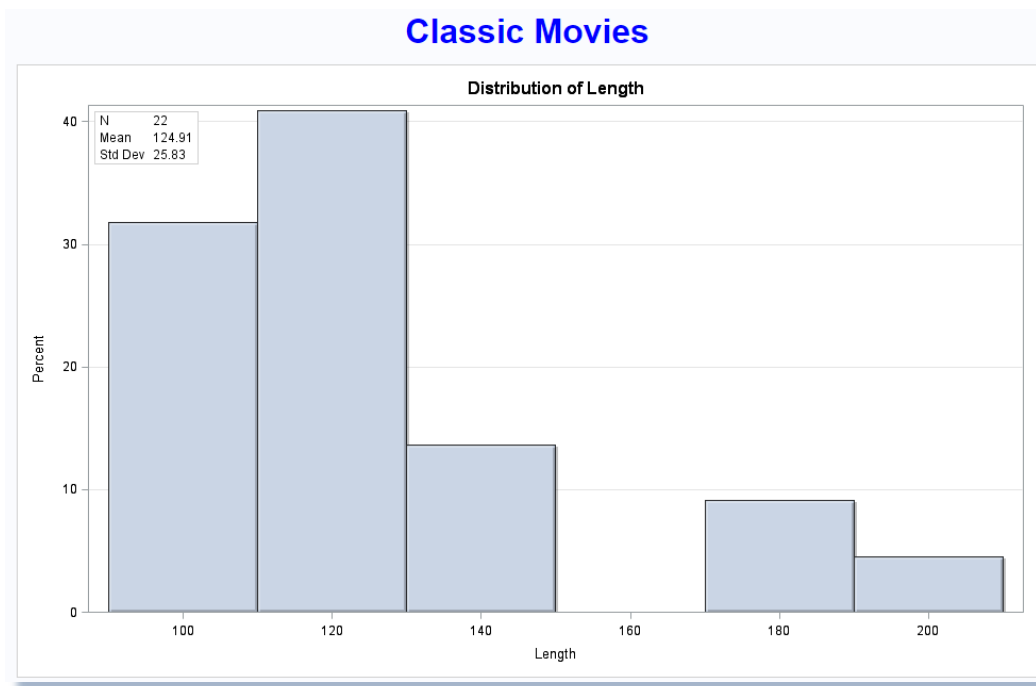
```

* Step 1 - Access Data Sources ;
libname mydata 'e:\workshops\workshop data' ;

* Step 2 - Build Example 1 Static Dashboard ;
ods html file='Example 1 Static Dashboard.html'
      path='e:\' (url=none) ;
ods graphics / reset imagemap=on width=8in height=5in
              imagename='Example 1 Static Dashboard' ;
TITLE 'Horizontal Bar Chart with SGPLOT' ;
PROC SGPLOT DATA=mydata.MOVIES ;
  FORMAT Length 5.2 ;
  HBAR RATING / RESPONSE=Length STAT=Mean DATALABEL ;
RUN ;
TITLE ;
ods html close ;
ods graphics off ;

```

2. Static Dashboard – PROC PRINT Dashboard using PROC FORMAT



Key Points about Code

1. Base SAS provides users with several procedures for creating graphical output. The three procedures available are: SGPLOT, SGSCATTER and SGPANEL.
2. The type of dashboard created in this approach is a static, or non-drill-down, dashboard.
3. A template definition, created with PROC TEMPLATE, provides details to define the chart including the chart title, the layout of an overlay scenario, the creation of a histogram chart, the specification of the “crisp” dataskin, and the specification of a true border.
4. The specification of the ODS HTML and GRAPHICS statements to define the dashboard’s dimensions.
5. PROC SGRENDER is used to produce a graph from an input SAS data set and an ODS graph template.

Base-SAS Code:

```

*****
**** Program Name: Dashboard with Histogram and PROC SGRENDER.SAS
**** Purpose.....: Create and display a simple dashboard interface using PROC
****                TEMPLATE and PROC SGRENDER.
****
**** Author.....: Kirk Paul Lafler, Software Intelligence Corporation
**** Date Written: 02/22/2015
**** SAS Version.: SAS 9.4
**** Input Files.: Workshop Data
****                Movies
**** Output Files: HTML Output (1 File)
****                Dashboard-SGRender-Histogram.htm
**** Subroutines.: None
**** User-defined Formats: None
**** Macro Variables: None
**** Includes....: None
**** Modification History:
**** 02/23/2015 KPL Added Header information.
*****

```

```
libname mydata 'e:\workshops\workshop data' ;
```

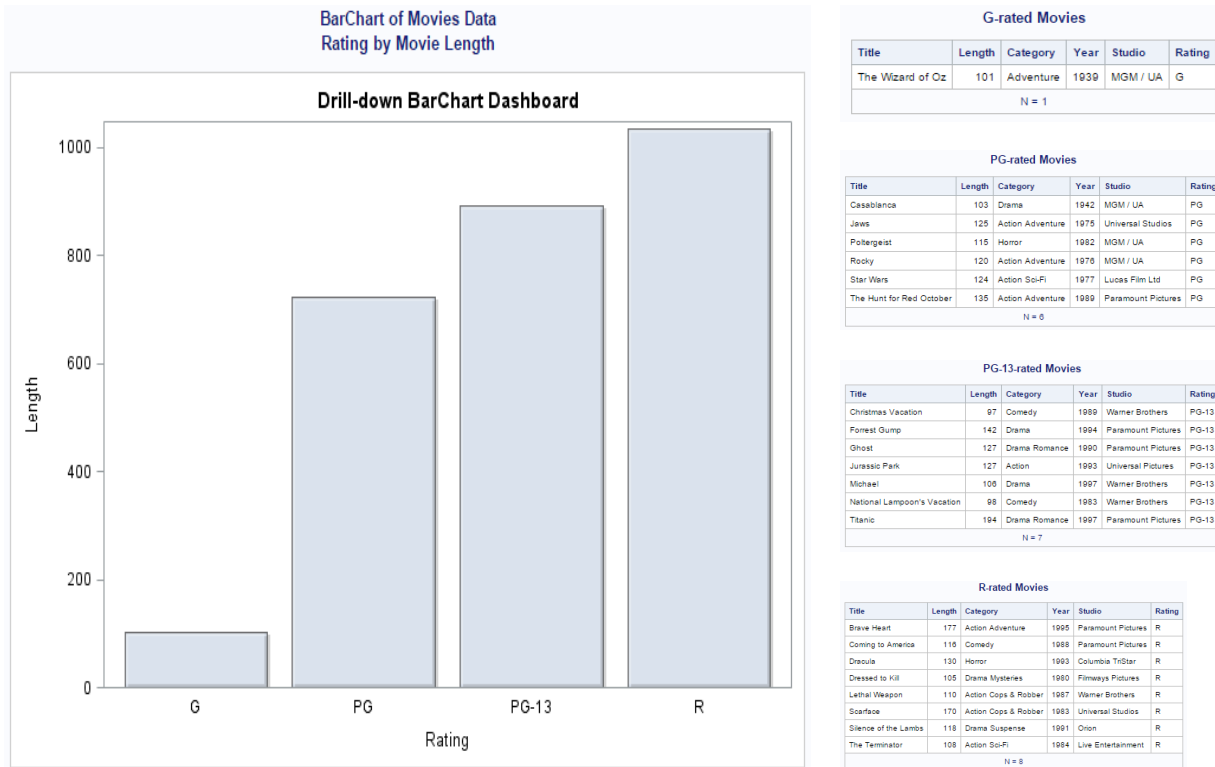
```
*STEP 1 - Create DashboardHistogram Template Overlay with PROC TEMPLATE ;
```

```
proc template ;
  define statgraph DashboardHistogram ;
    dynamic VAR ;
    begingraph ;
      entrytitle "Distribution of " VAR ;
      layout overlay / yaxisopts=(griddisplay=on) ;
        histogram VAR / scale=percent
          dataskin=crisp ;
      layout gridded / columns=2
        autoalign=(topleft topright) border=true
          opaque=true backgroundcolor=GraphWalls:color ;
      entry halign=left "N" ;
      entry halign=left eval(strip(put(n(VAR),12.0))) ;
      entry halign=left "Mean" ;
      entry halign=left eval(strip(put(mean(VAR),12.2))) ;
      entry halign=left "Std Dev" ;
      entry halign=left eval(strip(put(stddev(VAR),12.2))) ;
    endlayout ;
  endlayout ;
endgraph ;
end ;
run ;
```

```
*STEP 2 - Produce Graphical Output from DashboardHistogram Template using PROC SGRENDER ;
```

```
ods html file='Dashboard-SGRender-Histogram.htm' path='e:\' (url=none) ;
ods graphics / reset imagemap=on width=10in height=6in
  imagename='Dashboard-SGRender-Histogram' ;
title1 h=7 color=Blue "Classic Movies" ;
proc sgrender data=mydata.movies template=DashboardHistogram ;
  dynamic VAR="Length" ;
run ;
quit ;
title ;
ods html close ;
```

3. Interactive Drill-down Dashboard – Bar Chart Dashboard



Key Points about Code

1. The first step illustrates the data preparation process. In this example, the data is sorted in ascending order by the Movie Rating variable.
2. The second step assigns an HTML link, for drill-down purposes, to a URL (hyperlinking) variable with a series of IF-THEN/ELSE statements.
3. The third step creates a template definition using PROC TEMPLATE by providing details to define the chart including the chart title, the layout of an overlay scenario, the creation of a vertical bar chart, the specification of the “crisp” dataskin, a transparency value of .3 to allow the vertical bar chart to show through the overlay, a vertical orientation, and the specification of the URL= parameter to allow drill-down capabilities.
4. The fourth step specifies the ODS HTML and GRAPHICS statements, the dimensions of the dashboard, title and footnote statements, and PROC SGRENDER to produce the graphical image from an input SAS data set and an ODS graph template.
5. The fifth step creates the output to display when the dashboard user clicks one of the vertical bar chart’s bars displayed on the dashboard.

Base-SAS Code:

```

*****
**** Program Name: Drill-down with PROC SGRENDER and BarChart.SAS *****
**** Purpose.....: Create and display a simple drill-down dashboard interface *****
****                using a DATA step, PROC TEMPLATE, PROC SGRENDER with the *****
****                ODS HTML destination. *****
**** Author.....: Kirk Paul Lafler, Software Intelligence Corporation *****
**** Date Written: 02/19/2015 *****
**** SAS Version.: SAS 9.4 *****
**** Input Files.: Movies Workshop Data *****
**** Output Files: HTML Output (6 Files) *****
****                SGRender-BarChart-Dashboard.htm *****
****                SGRender-BarChart-G-Movies.htm *****

```

```

*****          SGRender-BarChart-PG-Movies.htm          *****
*****          SGRender-BarChart-PG13-Movies.htm        *****
*****          SGRender-BarChart-PG17-Movies.htm        *****
*****          SGRender-BarChart-R-Movies.htm           *****
***** Subroutines.: None                               *****
***** User-defined Formats: None                       *****
***** Macro Variables: None                           *****
***** Includes....: None                               *****
***** Modification History:                            *****
***** 02/21/2015 KPL Added Header information.         *****
*****;
libname mydata 'e:\workshops\workshop data';

/*****
/* STEP 1 - Order Movies Dataset in Ascending Order by Rating */
/*****
proc sort data=mydata.movies out=work.sorted_movies ;
  by rating ;
run ;

/*****
/* STEP 2 - Assign HTML Link to URLLINK Variable */
/*****
data Classic_Movies;
  length URLLINK $50;
  max=200;
  set work.sorted_movies;
  if upcase(rating) = 'G'      then URLLINK='SGRender-BarChart-G-Movies.htm';
  else
  if upcase(rating) = 'PG'    then URLLINK='SGRender-BarChart-PG-Movies.htm';
  else
  if upcase(rating) = 'PG-13' then URLLINK='SGRender-BarChart-PG13-Movies.htm';
  else
  if upcase(rating) = 'PG-17' then URLLINK='SGRender-BarChart-PG17-Movies.htm';
  else
  if upcase(rating) = 'R'     then URLLINK='SGRender-BarChart-R-Movies.htm';
run;

/*****
/* STEP 3 - Create BarChart Template Overlay with PROC TEMPLATE */
/*****
proc template ;
  define statgraph BarChartDashboard ;
    begingraph ;
    entrytitle "Drill-down BarChart Dashboard" ;
    layout overlay / xaxisopts=(display=(label tickvalues line)) ;
      barchart x=Rating y=length / url=urllink
              dataskin=crisp
              datatransparency=0.3
              orient=vertical ;
    endlayout ;
  endgraph ;
end ;
run ;

/*****
/* Step 4 - Produce Graphical Output from BarChart Template using PROC SGRENDER */
/*****
ods html file='SGRender-BarChart-Dashboard.htm' path='e:\' (url=none) ;
ods graphics / reset imagemap=on width=7in height=5in
              imagename='SGRender-BarChart-Dashboard' ;
title1 "BarChart of Movies Data" ;
title2 "Rating by Movie Length" ;
proc sgrender data=Classic_Movies template=BarChartDashboard ;
run ;
title ;
ods html close ;

/*****
/* STEP 5 - Create Drill-down Output for Each Rating Group with PROC PRINT */
/*****
ODS html body="SGRender-BarChart-G-Movies.htm" path='e:\' (url=none) ;
PROC PRINT DATA=classic_movies(DROP=urllink max) NOOBS N ;
  TITLE "G-rated Movies" ;
  WHERE UPCASE(rating) = "G" ;

```



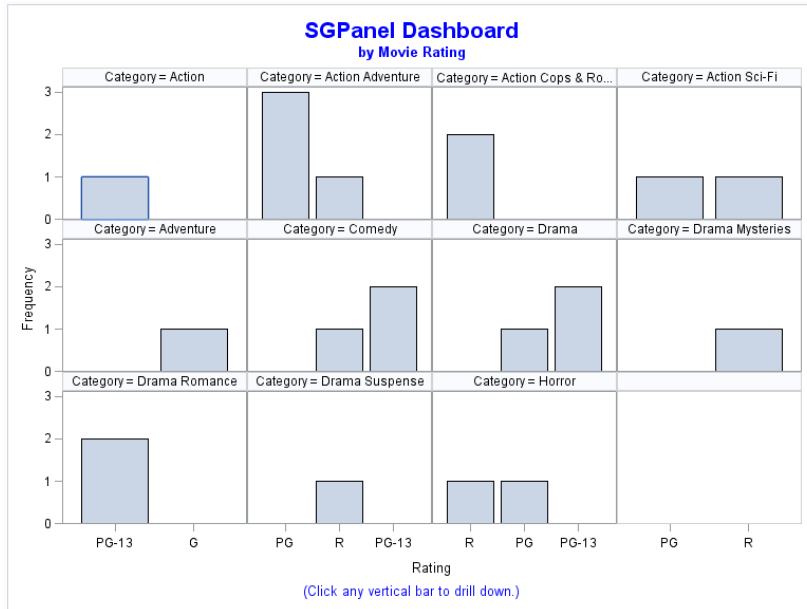
```

RUN ;
ODS html close ;

. . . . .

ODS html body="SGRender-BarChart-R-Movies.htm" path='e:\' (url=none) ;
PROC PRINT DATA=classic_movies(DROP=url link max) NOOBS N ;
  TITLE "R-rated Movies" ;
  WHERE UPCASE(rating) = "R" ;
RUN ;
ODS html close ;
    
```

4. Interactive Drill-down Dashboard – SGPanel Dashboard using PROC FORMAT



G-rated Movies

Title	Length	Category	Year	Studio	Rating
The Wizard of Oz	101	Adventure	1939	MGM / UA	G

N = 1

PG-rated Movies

Title	Length	Category	Year	Studio	Rating
Casablanca	103	Drama	1942	MGM / UA	PG
Jaws	125	Action Adventure	1975	Universal Studios	PG
Pottergeist	115	Horror	1982	MGM / UA	PG
R Rocky	120	Action Adventure	1976	MGM / UA	PG
Star Wars	124	Action Sci-Fi	1977	Lucas Film Ltd	PG
The Hunt for Red October	135	Action Adventure	1989	Paramount Pictures	PG

N = 6

PG-13-rated Movies

Title	Length	Category	Year	Studio	Rating
Christmas Vacation	97	Comedy	1989	Warner Brothers	PG-13
Forrest Gump	142	Drama	1994	Paramount Pictures	PG-13
Ghost	127	Drama Romance	1990	Paramount Pictures	PG-13
Jurassic Park	127	Action	1993	Universal Pictures	PG-13
Michael	106	Drama	1967	Warner Brothers	PG-13
National Lampoon's Vacation	98	Comedy	1983	Warner Brothers	PG-13
Titanic	194	Drama Romance	1997	Paramount Pictures	PG-13

N = 7

R-rated Movies

Title	Length	Category	Year	Studio	Rating
Brave Heart	177	Action Adventure	1995	Paramount Pictures	R
Coming to America	116	Comedy	1988	Paramount Pictures	R
Dracula	130	Horror	1993	Columbia TriStar	R
Dressed to Kill	105	Drama Mysteries	1980	Filmways Pictures	R
Lethal Weapon	110	Action Cops & Robber	1987	Warner Brothers	R
Scarface	170	Action Cops & Robber	1983	Universal Studios	R
Silence of the Lambs	118	Drama Suspense	1991	Orion	R
The Terminator	108	Action Sci-Fi	1984	Live Entertainment	R

N = 8

Key Points about Code

1. The first step illustrates any data preparation process including data manipulation techniques. In this example, a PROC FORMAT is specified to create a URL variable to be associated with the Movie Rating variable.
2. The second step assigns the HTML link, for drill-down purposes, to a URLLINK variable with the user-defined format created in step 1, above.
3. The third step specifies the ODS HTML and GRAPHICS statements, the dimensions of the dashboard, title and footnote statements, PROC SGPanel to produce the graphical vertical bar chart from an input SAS data set, and the specification of the URL= parameter to allow drill-down capabilities.
4. The fourth step creates the output to display when the user clicks one of the vertical bar chart's bars displayed on the dashboard.

Base-SAS Code:

```

*****
**** Program Name: SGPANEL-Dashboard-1.SAS ****
**** Purpose.....: Create and display a simple drill-down dashboard interface ****
****                using PROC SGPANEL and PROC FORMAT with the ODS HTML. ****
****                ****
**** Author.....: Kirk Paul Lafler, Software Intelligence Corporation ****
**** Date Written: 03/19/2016 ****
**** SAS Version.: SAS 9.4 ****
**** Input Files.: Workshop Data ****
****                Movies ****
**** Output Files: VBAR HTML Output (6 Files) ****
****                SGPanel-Dashboard-1.htm ****
****                SGPanel-1-Drill-down-G-Movies.htm ****
****                SGPanel-1-Drill-down-PG-Movies.htm ****
****                SGPanel-1-Drill-down-PG13-Movies.htm ****
****                SGPanel-1-Drill-down-PG17-Movies.htm ****
****                SGPanel-1-Drill-down-R-Movies.htm ****
**** Image Files: SGPanel-Dashboard-1.png ****
**** Subroutines.: None ****
**** User-defined Formats: $RATFMT. ****
**** Macro Variables: None ****
**** Includes....: None ****
**** Modification History: ****
**** 03/19/2016 KPL Added Header information. ****
*****
LIBNAME mydata 'e:\workshops\workshop data' ;

*STEP 1 - Create User-defined Rating Format ;
PROC FORMAT ;
  VALUE $RATFMT
    'G'   = 'SGPanel-1-Drill-down-G-Movies.htm'
    'PG'  = 'SGPanel-1-Drill-down-PG-Movies.htm'
    'PG-13' = 'SGPanel-1-Drill-down-PG13-Movies.htm'
    'PG-17' = 'SGPanel-1-Drill-down-PG17-Movies.htm'
    'R'   = 'SGPanel-1-Drill-down-R-Movies.htm' ;
RUN ;

*STEP 2 - Assign HTML Link to URLLINK Variable ;
data Classic_Movies ;
  set mydata.movies ;
  urllink = put(rating,$RATFMT.) ;
run ;

*STEP 3 - Create Drill-down HTML File with PROC SGPANEL and User-defined Format ;
ODS html file="SGPanel-Dashboard-1.htm" path='e:\' (url=none) ;
ods graphics / reset imagemap=on width=8in height=6in imagename='SGPanel-Dashboard-1' ;
TITLE1 h=.2in color=blue "SGPanel Dashboard" ;
TITLE2 color=blue "by Movie Rating" ;
FOOTNOTE COLOR=blue "(Click any vertical bar to drill down.)" ;
OPTIONS NODATE ;
proc sgppanel data=Classic_Movies ;
  panelby category / onepanel uniscale=row ;
  vbar rating / url=urllink ;
run ;
ODS html close ;

*STEP 4 - Create Drill-down Output for Each Rating Group with PROC PRINT ;
TITLE ;
FOOTNOTE ;
ODS html body="SGPanel-1-Drill-down-G-Movies.htm" (url=none) ;
PROC PRINT DATA=Classic_Movies(DROP=urllink) NOOBS N ;
  TITLE "G-rated Movies" ;
  WHERE UPCASE(rating) = "G" ;
RUN ;
ODS html close ;

ODS html body="SGPanel-1-Drill-down-PG-Movies.htm" (url=none) ;
PROC PRINT DATA=Classic_Movies(DROP=urllink) NOOBS N ;
  TITLE "PG-rated Movies" ;
  WHERE UPCASE(rating) = "PG" ;
RUN ;
ODS html close ;

```

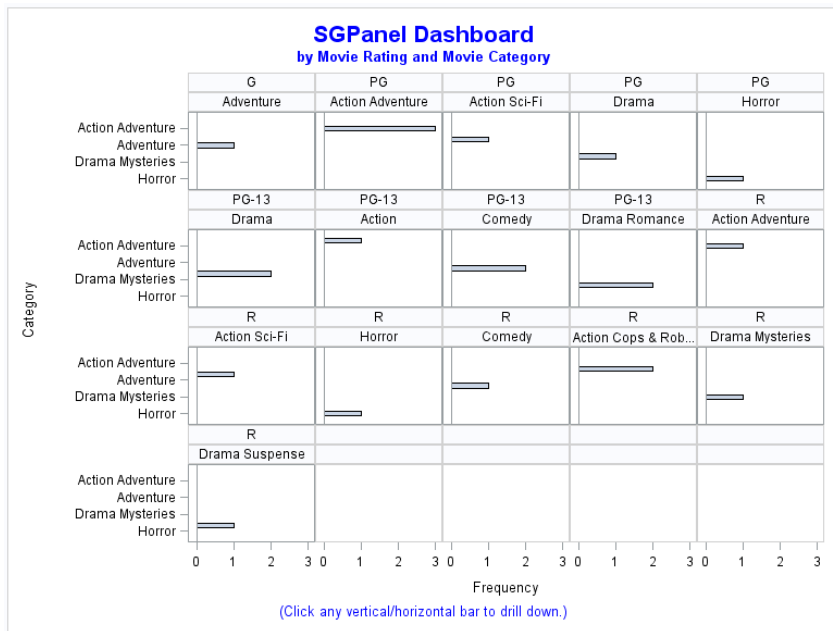
```

ODS html body="SGPanel-1-Drill-down-PG13-Movies.htm" (url=none) ;
PROC PRINT DATA=Classic_Movies(DROP=urllink) NOOBS N ;
  TITLE "PG-13-rated Movies" ;
  WHERE UPCASE(rating) = "PG-13" ;
RUN ;
ODS html close ;

ODS html body="GPanel-1-Drill-down-PG17-Movies.htm" (url=none) ;
PROC PRINT DATA=Classic_Movies(DROP=urllink) NOOBS N ;
  TITLE "PG-17-rated Movies" ;
  WHERE UPCASE(rating) = "PG-17" ;
RUN ;
ODS html close ;

ODS html body="SGPanel-1-Drill-down-R-Movies.htm" (url=none) ;
PROC PRINT DATA=Classic_Movies(DROP=urllink) NOOBS N ;
  TITLE "R-rated Movies" ;
  WHERE UPCASE(rating) = "R" ;
RUN ;
ODS html close ;
    
```

5. Interactive Drill-down Dashboard – SGPanel with HBAR using PROC FORMAT



G-rated Movies

Title	Length	Category	Year	Studio	Rating
The Wizard of Oz	101	Adventure	1939	MGM / UA	G

N = 1

PG-rated Movies

Title	Length	Category	Year	Studio	Rating
Casablanca	103	Drama	1942	MGM / UA	PG
Jaws	125	Action Adventure	1975	Universal Studios	PG
Pottergeist	115	Horror	1982	MGM / UA	PG
Rocky	120	Action Adventure	1976	MGM / UA	PG
Star Wars	124	Action Sci-Fi	1977	Lucas Film Ltd	PG
The Hunt for Red October	135	Action Adventure	1989	Paramount Pictures	PG

N = 6

PG-13-rated Movies

Title	Length	Category	Year	Studio	Rating
Christmas Vacation	97	Comedy	1989	Warner Brothers	PG-13
Forest Gump	142	Drama	1994	Paramount Pictures	PG-13
Ghost	127	Drama Romance	1990	Paramount Pictures	PG-13
Jurassic Park	127	Action	1993	Universal Pictures	PG-13
Michael	106	Drama	1997	Warner Brothers	PG-13
National Lampoon's Vacation	98	Comedy	1983	Warner Brothers	PG-13
Titanic	194	Drama Romance	1997	Paramount Pictures	PG-13

N = 7

R-rated Movies

Title	Length	Category	Year	Studio	Rating
Brave Heart	177	Action Adventure	1995	Paramount Pictures	R
Coming to America	118	Comedy	1989	Paramount Pictures	R
Dracula	130	Horror	1993	Columbia TriStar	R
Dressed to Kill	105	Drama Mysteries	1980	Filmways Pictures	R
Lethal Weapon	110	Action Cops & Robber	1987	Warner Brothers	R
Scarface	170	Action Cops & Robber	1983	Universal Studios	R
Silence of the Lambs	118	Drama Suspense	1991	Orion	R
The Terminator	108	Action Sci-Fi	1984	Live Entertainment	R

N = 8

Key Points about Code

1. The first step illustrates any data preparation process including data manipulation techniques. In this example, a PROC FORMAT is specified to create a URL variable to be associated with the Movie Rating variable. Another PROC FORMAT is specified to create a URL (hyperlinking) variable to be associated with the Movie Category variable.
2. The second step assigns the HTML hyperlink values, for drill-down purposes, to the URL (hyperlinking) variables created with the user-defined formats in step 1, above.

- The third step specifies the ODS HTML and GRAPHICS statements, the dimensions of the dashboard, title and footnote statements, PROC SGPanel to produce the graphical horizontal bar chart from an input SAS data set, and the specification of the URL= parameters to allow drill-down capabilities.
- The fourth step specifies the type of output to produce when the user clicks the horizontal bar chart's bars displayed on the dashboard.

Base-SAS Code:

```
*****
**** Program Name: SGPanel-Dashboard-2.SAS ****
**** Purpose.....: Create and display a simple drill-down dashboard interface ****
****                using PROC SGPanel and PROC FORMAT with the ODS HTML. ****
****                ****
**** Author.....: Kirk Paul Lafler, Software Intelligence Corporation ****
**** Date Written: 03/19/2016 ****
**** SAS Version.: SAS 9.4 ****
**** Input Files.: Workshop Data ****
****                Movies ****
**** Output Files: VBAR HTML Output (6 Files) ****
****                SGPanel-Dashboard-1.htm ****
****                SGPanel-2-Drill-down-G-Movies.htm ****
****                SGPanel-2-Drill-down-PG-Movies.htm ****
****                SGPanel-2-Drill-down-PG13-Movies.htm ****
****                SGPanel-2-Drill-down-PG17-Movies.htm ****
****                SGPanel-2-Drill-down-R-Movies.htm ****
**** Image Files: SGPanel-Dashboard-2.png ****
**** Subroutines.: None ****
**** User-defined Formats: $RATFMT. and $CATFMT. ****
**** Macro Variables: None ****
**** Includes....: None ****
**** Modification History: ****
**** 03/19/2016 KPL Added Header information. ****
*****
```

```
LIBNAME mydata 'e:\workshops\workshop data' ;
```

```
*STEP 1a - Create User-defined Rating Format ;
```

```
PROC FORMAT ;
  VALUE $RATFMT
    'G'      = 'SGPanel-2-Drill-down-G-Movies.htm'
    'PG'     = 'SGPanel-2-Drill-down-PG-Movies.htm'
    'PG-13'  = 'SGPanel-2-Drill-down-PG13-Movies.htm'
    'PG-17'  = 'SGPanel-2-Drill-down-PG17-Movies.htm'
    'R'      = 'SGPanel-2-Drill-down-R-Movies.htm' ;
RUN ;
```

```
*STEP 1b - Create User-defined Category Format ;
```

```
PROC FORMAT ;
  VALUE $CATFMT
    'Action'           = 'SGPanel-2-Drill-down-Act-Movies.htm'
    'Action Adventure' = 'SGPanel-2-Drill-down-AA-Movies.htm'
    'Action Cops & Robber' = 'SGPanel-2-Drill-down-ACR-Movies.htm'
    'Action Sci-Fi'    = 'SGPanel-2-Drill-down-ASF-Movies.htm'
    'Adventure'        = 'SGPanel-2-Drill-down-Adv-Movies.htm'
    'Comedy'           = 'SGPanel-2-Drill-down-C-Movies.htm'
    'Drama'            = 'SGPanel-2-Drill-down-D-Movies.htm'
    'Drama Mysteries'  = 'SGPanel-2-Drill-down-DM-Movies.htm'
    'Drama Romance'    = 'SGPanel-2-Drill-down-DR-Movies.htm'
    'Drama Suspense'   = 'SGPanel-2-Drill-down-DS-Movies.htm'
    'Horror'           = 'SGPanel-2-Drill-down-H-Movies.htm' ;
RUN ;
```

```
*STEP 2 - Assign HTML Link to URLLINK Variable ;
```

```
data Classic_Movies ;
  set mydata.movies ;
  urllink1 = put(rating,$RATFMT.) ;
  urllink2 = put(category,$CATFMT.) ;
run ;
```

```
*STEP 3 - Create Drill-down HTML File with PROC SGPanel and User-defined Format ;
```

```
ods html file="SGPanel-Dashboard-2.htm" path='e:\' (url=None) ;
ods graphics / reset imagemap=on width=8in height=6in imagename='SGPanel-Dashboard-2' ;
TITLE1 h=2in color=blue "SGPanel Dashboard" ;
```

```

TITLE2 color=blue "by Movie Rating and Movie Category" ;
FOOTNOTE COLOR=blue "(Click any vertical/horizontal bar to drill down.)" ;
OPTIONS NODATE ;
proc sgpanel data=Classic_Movies ;
  panelby rating category / layout=panel onepanel novarname columns=5 ;
  hbar category / url=urllink2 ;
run ;
ODS html close ;

*STEP 4 - Create Drill-down Output for Each Rating Group with PROC PRINT ;
TITLE ;
FOOTNOTE ;
ODS html body="SGPanel-2-Drill-down-G-Movies.htm" (url=none) ;
PROC PRINT DATA=Classic_Movies(DROP=urllink1) NOOBS N ;
  TITLE "G-rated Movies" ;
  WHERE UPCASE(rating) = "G" ;
RUN ;
ODS html close ;

ODS html body="SGPanel-2-Drill-down-PG-Movies.htm" (url=none) ;
PROC PRINT DATA=Classic_Movies(DROP=urllink1) NOOBS N ;
  TITLE "PG-rated Movies" ;
  WHERE UPCASE(rating) = "PG" ;
RUN ;
ODS html close ;

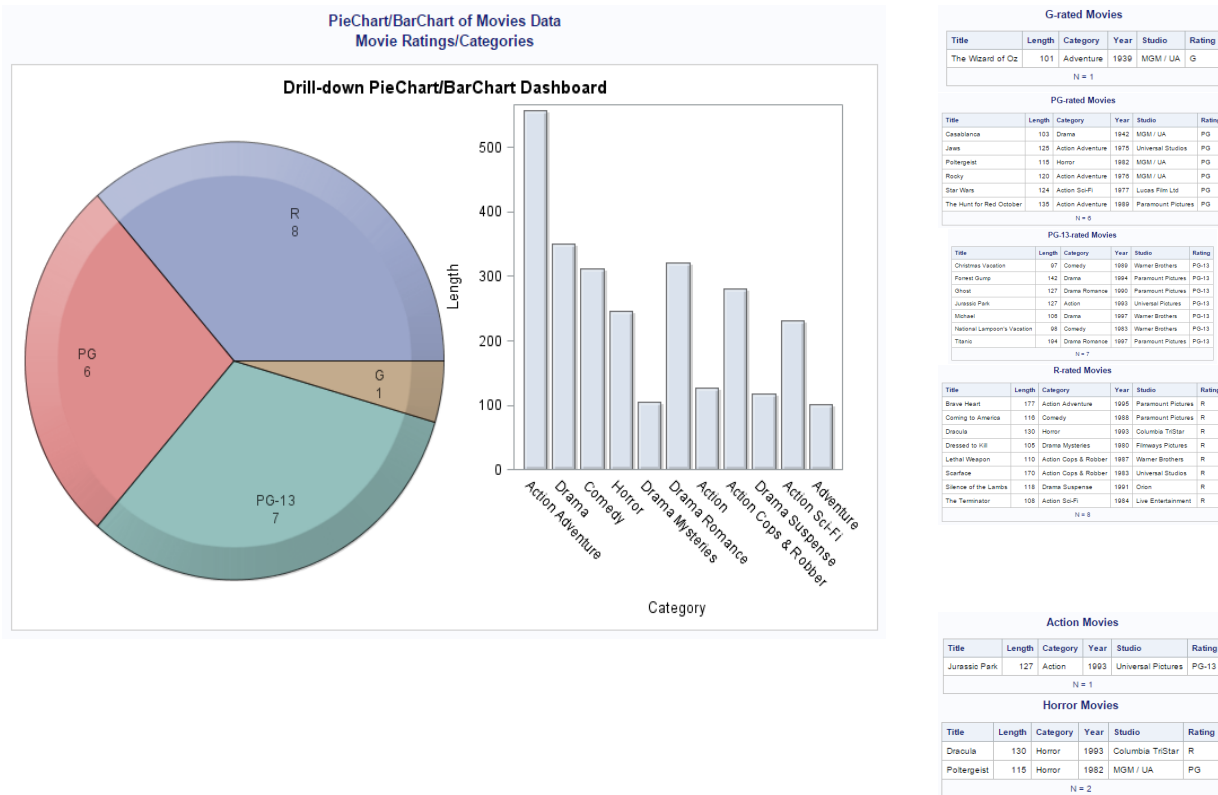
ODS html body="SGPanel-2-Drill-down-PG13-Movies.htm" (url=none) ;
PROC PRINT DATA=Classic_Movies(DROP=urllink1) NOOBS N ;
  TITLE "PG-13-rated Movies" ;
  WHERE UPCASE(rating) = "PG-13" ;
RUN ;
ODS html close ;

ODS html body="SGPanel-2-Drill-down-PG17-Movies.htm" (url=none) ;
PROC PRINT DATA=Classic_Movies(DROP=urllink1) NOOBS N ;
  TITLE "PG-17-rated Movies" ;
  WHERE UPCASE(rating) = "PG-17" ;
RUN ;
ODS html close ;

ODS html body="SGPanel-2-Drill-down-R-Movies.htm" (url=none) ;
PROC PRINT DATA=Classic_Movies(DROP=urllink1) NOOBS N ;
  TITLE "R-rated Movies" ;
  WHERE UPCASE(rating) = "R" ;
RUN ;
ODS html close ;

```

6. Interactive Drill-down Dashboard – Pie / Bar Chart with GTL



Key Points about Code

1. The first step specifies a series of IF-THEN/ELSE statements to create a URL (hyperlinking) variable to be associated with the Movie Rating variable and another URL (hyperlinking) variable to be associated with the Movie Category variable.
1. The second step uses a PROC TEMPLATE to create a template overlay to be used in step 3, below.
2. The third step assigns the HTML links, for drill-down purposes, to the URL (hyperlinking) variables created in step 1, above, and the user-defined template in step 2, above. It also specifies the ODS HTML and GRAPHICS statements, the dimensions of the dashboard, title and footnote statements, PROC SGRENDER to produce the graphical pie chart and vertical bar chart from the input SAS data set, and the specification of the URL= parameters to allow drill-down capabilities.
3. The fourth step specifies the type of output to produce when the user clicks on one the pieces of the pie chart displayed on the dashboard.
4. The fifth step specifies the type of output to produce when the user clicks on the bars of the vertical bar chart displayed on the dashboard.

Base-SAS Code:

```

*****
**** Program Name: 2-Cell Drill-down, PROC SGRENDER, Pie and Bar Chart #2.SAS
**** Purpose.....: Create and display a simple drill-down dashboard with two
****                categorical variables using a DATA step, PROC TEMPLATE, Pie,
****                Bar Chart, and PROC SGRENDER with the ODS HTML destination.
****
**** Author.....: Kirk Paul Lafler, Software Intelligence Corporation
**** Date Written: 03/28/2015
**** SAS Version.: SAS 9.4
**** Input Files.: Workshop Data
****                Movies
**** Output Files: HTML Output (17 Files)
****                PieChart-BarChart-Dashboard.htm (Qty 1)
****                PieChart-Rating-Movies.htm (Qty 5)
****                BarChart-Category-Movies.htm (Qty 11)
**** Subroutines.: None
**** User-defined Formats: None
**** Macro Variables: None
**** Includes....: None
**** Modification History:
**** 03/28/2015 KPL Added Header information.
*****
libname mydata 'e:\workshops\workshop data';

/*****
/* Step 1a - Assign PieChart URL= to URLRATING Variable */
/*****
data Classic_Movies;
  length URLRATING URLCATEGORY $60;
  max=200;
  set mydata.movies;
  if upcase(rating) = 'G' then URLRATING='PieChart-G-Movies.htm' (url=none) ;
  else
  if upcase(rating) = 'PG' then URLRATING='PieChart-PG-Movies.htm' (url=none) ;
  else
  if upcase(rating) = 'PG-13' then URLRATING='PieChart-PG13-Movies.htm' (url=none) ;
  else
  if upcase(rating) = 'R' then URLRATING='PieChart-R-Movies.htm' (url=none) ;

/*****
/* Step 1b - Assign BarChart URL= to URLCATEGORY Variable */
/*****
  if upcase(category) = 'ACTION' then
    URLCATEGORY='BarChart-Action-Movies.htm' (url=none) ;
  else
    . . . . .

  else
  if upcase(category) = 'HORROR' then
    URLCATEGORY='BarChart-Horror-Movies.htm' (url=none) ;
run;

/*****
/* Step 2 - Create PieChart / BarChart Template Overlay with PROC TEMPLATE */
/*****
proc template ;
  define statgraph PieChartBarChartDashboard ;
    begingraph ;
    entrytitle "Drill-down PieChart/BarChart Dashboard" ;
    layout gridded / columns=1 ;
    layout lattice / columns=2 ;
    cell ;
    layout region ;
    piechart category=Rating / url=URLRATING
              dataskin=crisp
              datatransparency=0.3 ;
    endlayout ;
  endcell ;
  cell ;
  layout overlay / width=250px xaxisopts=(display=(label tickvalues line)) ;
  barchart x=Category y=length / url=URLCATEGORY
           dataskin=crisp

```

```

                                datatransparency=0.3
                                orient=vertical
                                barwidth=0.8 ;
                                endlayout ;
                                endcell ;
                                endlayout ;
                                endlayout ;
                                endgraph ;
                                end ;
run ;
quit ;

/*****
/* Step 3 - Produce Graphical Output from PieChart BarChart */
/*      Template using PROC SGRENDER      */
*****/
ods html file='SGRender-PieChart-BarChart-Dashboard.htm'
      path='e:\' (url=none) ;
ods graphics / reset imagemap=on width=8in height=5in
      imagename='PieChartBarChart-Dashboard' ;
title1 "PieChart/BarChart of Movies Data" ;
title2 "Movie Ratings/Categories" ;
proc sgrender data=Classic_Movies
      template=PieChartBarChartDashboard ;
run ;
quit ;
title ;
ods html close ;

/*****
/* Step 4 - Create Pie Chart Drill-down Output for Each Rating Group with */
/*      PROC PRINT      */
*****/
ODS html body="PieChart-G-Movies.htm" path='e:\' (url=none) ;
PROC PRINT DATA=classic_movies(DROP=URLRATING URLCATEGORY MAX) NOOBS N ;
  TITLE "G-rated Movies" ;
  WHERE UPCASE(rating) = "G" ;
RUN ;
ODS html close ;

. . . . .

ODS html body="PieChart-R-Movies.htm" path='e:\' (url=none) ;
PROC PRINT DATA=classic_movies(DROP=URLRATING URLCATEGORY MAX) NOOBS N ;
  TITLE "R-rated Movies" ;
  WHERE UPCASE(rating) = "R" ;
RUN ;
ODS html close ;

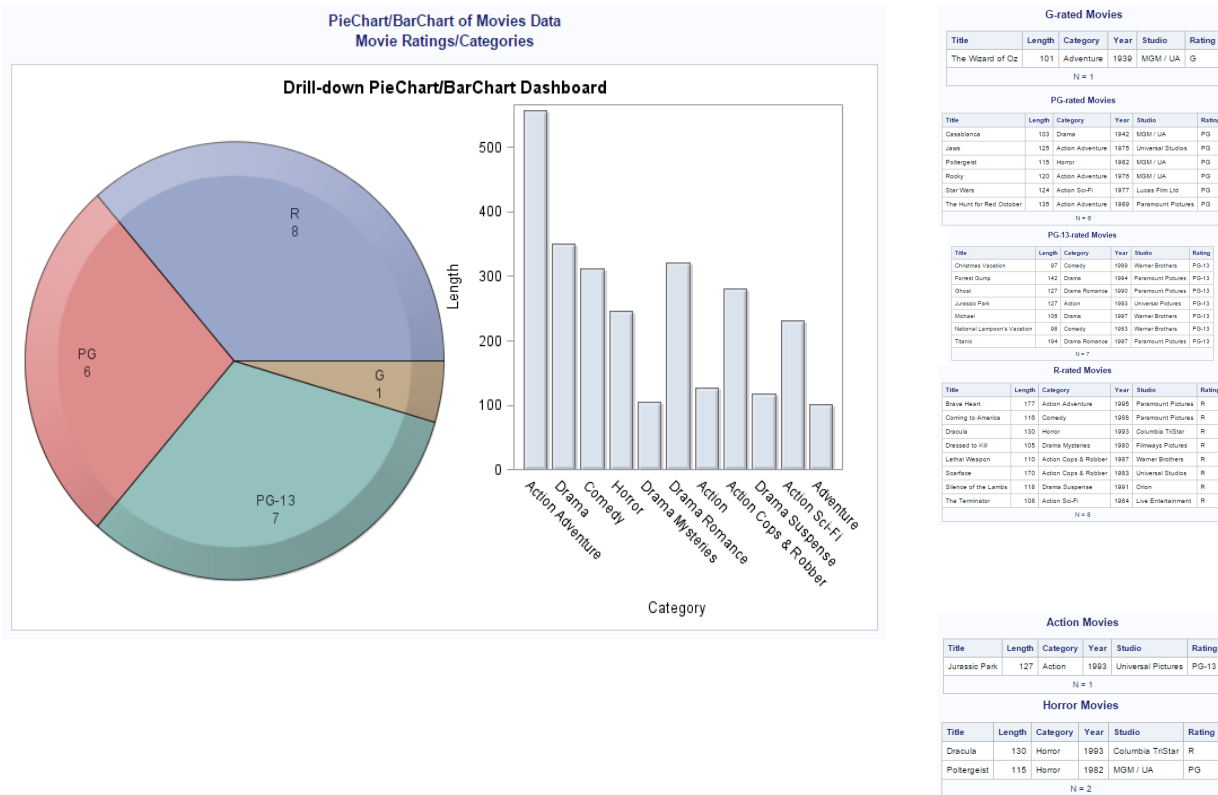
/*****
/* Step 5 - Create Bar Chart Drill-down Output for Each Category Group with */
/*      PROC PRINT      */
*****/
ODS html body="BarChart-Action-Movies.htm" path='e:\' (url=none) ;
PROC PRINT DATA=classic_movies(DROP=URLRATING URLCATEGORY MAX) NOOBS N ;
  TITLE "Action Movies" ;
  WHERE UPCASE(category) = "ACTION" ;
RUN ;
ODS html close ;

. . . . .

ODS html body="BarChart-Horror-Movies.htm" path='e:\' (url=none) ;
PROC PRINT DATA=classic_movies(DROP=URLRATING URLCATEGORY MAX) NOOBS N ;
  TITLE "Horror Movies" ;
  WHERE UPCASE(category) = "HORROR" ;
RUN ;
ODS html close ;

```


7. Interactive Drill-down Dashboard – Pie / Bar Chart with GTL and Macro Language



Key Points about Code

1. The first step specifies two user-defined PROC FORMATS to create the URL (hyperlinking) variables that will be associated with the Movie Rating variable and the Movie Category variable.
2. The second step uses a PROC TEMPLATE to create a template overlay to be used in step 3, below.
3. The third step assigns the HTML hyperlink values, for drill-down purposes, to the URL (hyperlinking) variables created in step 1, above, and the user-defined template in step 2, above. It also specifies the ODS HTML and GRAPHICS statements, the dimensions of the dashboard, title and footnote statements, PROC SGRENDER to produce the graphical pie chart and vertical bar chart from the input SAS data set, and the specification of the URL= parameters to allow drill-down capabilities.
4. The fourth step creates the output to display when the user clicks on one the pieces of the pie chart displayed on the dashboard.
5. The fifth step creates the output to display when the user clicks on the bars of the vertical bar chart displayed on the dashboard.

Base-SAS Code:

```

*****
**** Program Name: 2-panel Dashboard.SAS
**** Purpose.....: Create and display a 2-panel interactive dashboard with two
****                categorical variables using a DATA step, PROC TEMPLATE, Pie,
****                Bar Chart, and PROC SGRENDER with the ODS HTML destination.
****
**** Author.....: Kirk Paul Lafler, Joshua Horstman and Roger Muller
**** Date Written: 03/28/2015
**** SAS Version.: SAS 9.4
**** Input Files.: Workshop Data
****                Movies
**** Output Files: HTML Output (17 Files)
****                2-panel-Dashboard.htm (Qty 1)
****                2-panel-Drilldown-PieChart.htm (Qty 5)
****                2-panel-Drilldown-BarChart.htm (Qty 11)
**** Subroutines.: None
**** User-defined Formats: None
**** Macro Variables: None
**** Includes....: None
**** Modification History:
**** 06/20/2017 KPL Added Macro logic.
**** 03/28/2015 KPL Added Header information.
*****
%let dsn          = mydata.movies ;
%let myresults = /folders/myfolders/ ;
libname myresult "&myresults" ;
libname mydata  '/folders/myfolders/';

*STEP 1a - Create User-defined Rating Format ;
PROC FORMAT ;
  VALUE $RATFMT
    'G'      = '2-panel-Drilldown-PieChart-G-Movies.htm'
    'PG'     = '2-panel-Drilldown-PieChart-PG-Movies.htm'
    'PG-13' = '2-panel-Drilldown-PieChart-PG13-Movies.htm'
    'R'      = '2-panel-Drilldown-PieChart-R-Movies.htm' ;
RUN ;

*STEP 1b - Create User-defined Category Format ;
PROC FORMAT ;
  VALUE $CATFMT
    'Action'           = '2-panel-Drilldown-BarChart-Action-Movies.htm'
    'Action Adventure' = '2-panel-Drilldown-BarChart-Action Adventure-Movies.htm'
    'Action Cops & Robber' = '2-panel-Drilldown-BarChart-Action Cops & Robber-Movies.htm'
    'Action Sci-Fi'     = '2-panel-Drilldown-BarChart-Action Sci-Fi-Movies.htm'
    'Adventure'         = '2-panel-Drilldown-BarChart-Adventure-Movies.htm'
    'Comedy'            = '2-panel-Drilldown-BarChart-Comedy-Movies.htm'
    'Drama'             = '2-panel-Drilldown-BarChart-Drama-Movies.htm'
    'Drama Mysteries'   = '2-panel-Drilldown-BarChart-Drama Mysteries-Movies.htm'
    'Drama Romance'    = '2-panel-Drilldown-BarChart-Drama Romance-Movies.htm'
    'Drama Suspense'   = '2-panel-Drilldown-BarChart-Drama Suspense-Movies.htm'
    'Horror'            = '2-panel-Drilldown-BarChart-Horror-Movies.htm' ;
RUN ;

* STEP 2 - Assign HTML Link to URL (Hyperlink) Variable ;
data work.Movies_with_Hyperlinks ;
  set &dsn ;
  URLRatingLink =
    cats("2-panel-Drilldown-Piechart-",rating,"-Movies.htm") ;
  URLCategoryLink =
    cats("2-panel-Drilldown-BarChart-",category,"-Movies.htm") ;
run ;

/*****
/* Step 3 - Create PieChart / BarChart Template Overlay with PROC TEMPLATE */
/*****
proc template ;
  define statgraph TwoPanelDashboard ;
    begingraph ;
    entrytitle "2-Panel Interactive Dashboard" ;
    layout gridded / columns=1 ;
    layout lattice / columns=2 ;

```

```

cell ;
  layout region ;
    piechart category=Rating / url=URLRatingLink
                                dataskin=crisp
                                datatransparency=0.3 ;
  endlayout ;
endcell ;
cell ;
  layout overlay / width=250px xaxisopts=(display=(label tickvalues line)) ;
    barchart x=Category y=length / url=URLCategoryLink
                                dataskin=crisp
                                datatransparency=0.3
                                orient=vertical
                                barwidth=0.8 ;
  endlayout ;
endcell ;
endlayout ;
endgraph ;
end ;
run ;
quit ;

/*****
/* Step 4 - Produce Graphical Output from PieChart BarChart */
/*   Template using PROC SGRENDER                               */
*****/
ods html file='2-panel-Dashboard.htm'
  path='/folders/myfolders/' (url=none) ;
ods graphics / reset imagemap=on width=8in height=5in
  imagename='2-panel-Dashboard' ;
title1 "2-panel Dashboard" ;
title2 "Movie Ratings/Categories" ;
proc sgrender data=work.Movies_with_Hyperlinks
  template=TwoPanelDashboard ;
run ;
quit ;
title ;
ods html close ;
ods graphics off ;

/*****
/* Step 5 - Create Pie Chart Drill-down Output for Each Rating Group with */
/*   PROC PRINT                                                    */
*****/
proc sql noprint ;
  select distinct rating into :ratinglist separated by '~'
    from mydata.movies
    order by rating ;
  select count(distinct rating) into
    :numrating separated by ' '
    from mydata.movies ;
quit ;

%macro create_drilldowns_by_rating ;
  %do i = 1 %to &numrating ;
    ODS html
      body="2-panel-Drilldown-PieChart-%SCAN(&ratinglist,&i,~)-Movies.htm"
      path="/folders/myfolders/" (url=none) ;
    PROC PRINT DATA=mydata.movies NOOBS N ;
      TITLE "Detail Listing of %scan(&ratinglist,&i,~) Rated Movies" ;
      WHERE rating = "%SCAN(&ratinglist,&i,~)" ;
      var title rating category length ;
    RUN ;
    ODS html close ;
  %end ;
%mend create_drilldowns_by_rating ;
%create_drilldowns_by_rating ;

/*****
/* Step 6 - Create Bar Chart Drill-down Output for Each Category Group with */
/*   PROC PRINT                                                    */
*****/

```

```

proc sql noprint ;
  select distinct category into :categorylist separated by '~'
    from mydata.movies
    order by category ;
  select count(distinct category) into
    :numcategory separated by ' '
    from mydata.movies ;
quit ;

%macro create_drilldowns_by_category ;
  %do i = 1 %to &numcategory ;
    ODS html
      body="2-panel-Drilldown-BarChart-%SCAN(&categorylist,&i,~)-Movies.htm"
      path="/folders/myfolders/" (url=none) ;
    PROC PRINT DATA=mydata.movies NOOBS N ;
      TITLE "Detail Listing of %scan(&categorylist,&i,~) Movies" ;
      WHERE category = "%SCAN(&categorylist,&i,~)" ;
      var title rating category length ;
    RUN ;
    ODS html close ;
  %end ;
%mend create_drilldowns_by_category ;
%create_drilldowns_by_category ;

```

References

- Few, Stephen (2006), "[Common Pitfalls in Dashboard Design](#)," Copyright 2006, ProClarity Corporation, Boise, ID, USA.
- Lafler, Kirk Paul; Joshua M. Horstman and Roger D. Muller (2017), "[Building a Better Dashboard Using SAS® Base Software](#)," Proceedings of the 2017 Pharmaceutical SAS Users Group (PharmaSUG) Conference, The Trinomium Group, USA.
- Lafler, Kirk Paul; Joshua M. Horstman and Roger D. Muller (2016), "[Building a Better Dashboard Using SAS® Base Software](#)," Proceedings of the 2016 SouthCentral SAS Users Group (SCSUG) Conference, The Trinomium Group, USA.
- Lafler, Kirk Paul; Joshua M. Horstman and Roger D. Muller (2016), "[Building a Better Dashboard Using SAS® Base Software](#)," Proceedings of the 2016 SouthEast SAS Users Group (SESUG) Conference, The Trinomium Group, USA.
- Lafler, Kirk Paul; Joshua M. Horstman and Roger D. Muller (2016), "[Building a Better Dashboard Using SAS® Base Software](#)," Proceedings of the 2016 Pharmaceutical SAS Users Group (PharmaSUG) Conference, The Trinomium Group, USA.
- Lafler, Kirk Paul (2016), "[Dynamic Dashboards Using Base SAS® Software](#)," Proceedings of the 2016 SAS Global Forum (SGF) Conference, Software Intelligence Corporation, Spring Valley, CA, USA.
- Lafler, Kirk Paul (2015), "[Dynamic Dashboards Using Base SAS® Software](#)," Proceedings of the 2015 South Central SAS Users Group (SCSUG) Conference, Software Intelligence Corporation, Spring Valley, CA, USA.
- Lafler, Kirk Paul (2015), "[Dynamic Dashboards Using SAS®](#)," Proceedings of the 2015 SAS Global Forum (SGF) Conference, Software Intelligence Corporation, Spring Valley, CA, USA.
- Lafler, Kirk Paul (2013), [PROC SQL: Beyond the Basics Using SAS, Second Edition](#), SAS Institute Inc., Cary, NC, USA.
- Malik, Shadan (2007), "Elements for an Enterprise Dashboard," idashboards.com.
<http://www.dashboardinsight.com/articles/digital-dashboards/fundamentals/elements-for-an-enterprise-dashboard.aspx>
- Matange, Sanjay and Dan Heath (2011), Statistical Graphics Procedures by Example, SAS Institute Inc., Cary, NC, USA. [Click to view the book at the SAS Book store.](#)
- Nelson, Gregory S. (2009), "[Building Your First Dashboard Using the SAS® 9 Business Intelligence Platform: A Tutorial](#)," Proceedings of the 2009 SAS Global Forum (SGF) Conference, ThotWave Technologies, Cary, NC, USA.
- Overton, Stephen (2012), "[Lost in Wonderland? Methodology for a Guided Drill-Through Analysis Out of the Rabbit Hole](#)," Proceedings of the 2012 SAS Global Forum (SGF) Conference, Zencos Consulting, Cary, NC, USA.
- Sams, Scott (2013), "[SAS® BI Dashboard: Interactive, Data-Driven Dashboard Applications Made Easy](#)," Proceedings of the 2013 SAS Global Forum (SGF) Conference, SAS Institute Inc, Cary, NC, USA.
- Slaughter, Susan J. and Lora D. Delwiche (2010), "[Using PROC SGPLOT for Quick High-Quality Graphs](#)," Proceedings of the 2010 SAS Global Forum (SGF) Conference, SAS Institute Inc, Cary, NC, USA.
- Zdeb, Mike (2004), "[Pop-Ups, Drill-Downs, and Animation](#)," Proceedings of the 2004 SAS Users Group International (SUGI) Conference, University at Albany School of Public Health, Rensselaer, NY, USA.

Acknowledgments

The authors thank Clarence Jackson and Greg Gengo, SouthCentral SAS Users Group (SCSUG) Conference Co-Chairs for accepting our abstract and paper; the SouthCentral SAS Users Group (SCSUG) Executive Board; and SAS Institute for organizing and supporting a great conference!

Trademarks Citations

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.

Author Information

Kirk Paul Lafler is an entrepreneur, consultant and founder of Software Intelligence Corporation, and has been using SAS since 1979. Kirk is a SAS application developer, programmer, certified professional, provider of IT consulting services, mentor, advisor, professor at UC San Diego Extension, educator to SAS users around the world, and emeritus sasCommunity.org Advisory Board member. As the author of six books including Google® Search Complete (Odyssey Press. 2014) and PROC SQL: Beyond the Basics Using SAS, Second Edition (SAS Press. 2013); Kirk has written hundreds of papers and articles; been an Invited speaker and trainer at hundreds of SAS International, regional, special-interest, local, and in-house user group conferences and meetings; and is the recipient of 25 “Best” contributed paper, hands-on workshop (HOW), and poster awards.

Joshua M. Horstman is an independent statistical programmer with 18 years’ experience using SAS in the pharmaceutical industry. He has experience working for GlaxoSmithKline, Eli Lilly and Company, inVentiv Health, and others. Josh has presented numerous papers at PharmaSUG, SAS Global Forum, and other SAS User Group meetings. He holds a bachelor's degree in mathematics and computer science, and a master's degree in statistics from Colorado State University.

Roger D. Muller, Ph.D. has been as SAS user for over 40 years, primarily in the life sciences, business marketing, and educational areas. His current interests are in creative display of textual and graphical information using a wide variety of SAS products.

Comments and suggestions can be sent to:

Kirk Paul Lafler
SAS® Consultant, Application Developer, Programmer, Data Analyst, Educator and Author
E-mail: KirkLafler@cs.com
LinkedIn: <http://www.linkedin.com/in/KirkPaulLafler>
Twitter: @sasNerd

Joshua M. Horstman
Senior SAS® Consultant, Application Developer and Author
E-mail: jmhorstman@gmail.com
LinkedIn: <https://www.linkedin.com/in/joshuahorstman>

Roger D. Muller
Senior SAS® Consultant, Application Developer and Author
E-mail: rdmuller@hotmail.com
LinkedIn: <https://www.linkedin.com/in/roger-muller-b42a9929>