Multiple Graphical and Tabular Reports on One Page, Multiple Ways to Do It
Niraj J Pandya, CT, USA

ABSTRACT
Creating different kind of reports for the presentation of same data sounds a normal day job, but it becomes a little tough job to generate all different kind of reports on a single page. In the world of clinical trials, sometimes it is important to have the graphical output of data sitting next to the tabular output of data on same page to better understand the information captured in the data. It is generally a multiple frames one window kind of report and it is hard to get such report through conventional SAS® programming which usually gives listing form of reports. In such cases, Output Delivery System (ODS) is greatly helpful to overcome the limitations of traditional SAS output. Various features of ODS, new powerful SAS Version 9 capabilities and functionalities of output destinations like HTML and PDF enable anyone to get multiple graphs, charts, maps, listings and tables on a single page.

Use of ODS tagsets to create HTML panels in a single page with ActiveX functionality of SAS is described in this paper in detail. Also the use of SAS/GRAPH® options which control the size of graph, in combination of ODS are explained in this paper to insert multiple graphs on a single PDF page. In the end, new ODS LAYOUT capability of SAS is also explained to insert multiple tabular and graphical outputs in a single PDF page. This paper is intended for new to intermediate level of SAS users working on Windows/UNIX platforms.

INTRODUCTION
In the pharmaceutical industry where graphical presentation is equally important as tabular presentation, it is also very much important that both kind of presentation be generated side by side on the same page. For example, statistical output of laboratory results by visit is usually generated in both tabular and in graphical form. Now by looking at tabular report, it is hard to imagine the whole picture of pattern of results at each visit. At the same time, by looking at graphical presentation of mean and standard deviation at each visit adjacent to the tabular presentation, it is lot easier to understand the pattern of results. This makes reviewer’s life little easy as the hassle of browsing through different pages is lessened. Also this makes the reports attractive yet meaningful for regulatory submissions.

As always, there are multiple ways to accomplish the same task in SAS, there are couples of methods to get multiple graphical and tabular outputs on one single page too. In the past we have used GREPLAY with the use of TEMPLATE definitions to create such reports, but it always results in code enough complex to maintain and job becomes tougher if transition has to be occurred.

But since first introduced in version 7 of SAS®, Output delivery system (ODS) has come to rescue for such complex task. Output delivery system is designed to overcome the limitations of traditional SAS output. Traditional SAS output is in listing format by default. But ODS is flexible and can be used to create custom layouts and reports. In the newer version of SAS (v9.0 and onwards), ODS TAGSETS and ODS LAYOUT with the use of HTML and PDF destination has been real boon when it comes to generate multiple graphical and tabular reports on one page. Also added functionality of SAS/GRAPH® software is another way to achieve similar results. In this presentation, we will go through various methods to get multiple reports on a single page.

DATA
For illustration and presentation purpose, I have simulated following Vital Signs Data. We will use this data as reference in every method presented in this paper.
Vitals Dataset:

<table>
<thead>
<tr>
<th>PTID</th>
<th>SEX</th>
<th>VISIT</th>
<th>SYSBP</th>
<th>DIABP</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>101</td>
<td>M</td>
<td>1</td>
<td>130</td>
</tr>
<tr>
<td>2</td>
<td>101</td>
<td>M</td>
<td>2</td>
<td>125</td>
</tr>
<tr>
<td>3</td>
<td>101</td>
<td>M</td>
<td>3</td>
<td>124</td>
</tr>
<tr>
<td>4</td>
<td>101</td>
<td>M</td>
<td>4</td>
<td>120</td>
</tr>
<tr>
<td>5</td>
<td>102</td>
<td>F</td>
<td>1</td>
<td>130</td>
</tr>
<tr>
<td>6</td>
<td>102</td>
<td>F</td>
<td>2</td>
<td>119</td>
</tr>
<tr>
<td>7</td>
<td>102</td>
<td>F</td>
<td>3</td>
<td>122</td>
</tr>
<tr>
<td>8</td>
<td>102</td>
<td>F</td>
<td>4</td>
<td>105</td>
</tr>
<tr>
<td>9</td>
<td>103</td>
<td>F</td>
<td>1</td>
<td>138</td>
</tr>
<tr>
<td>10</td>
<td>103</td>
<td>F</td>
<td>2</td>
<td>127</td>
</tr>
<tr>
<td>11</td>
<td>103</td>
<td>F</td>
<td>3</td>
<td>114</td>
</tr>
<tr>
<td>12</td>
<td>103</td>
<td>F</td>
<td>4</td>
<td>121</td>
</tr>
<tr>
<td>13</td>
<td>104</td>
<td>M</td>
<td>1</td>
<td>135</td>
</tr>
<tr>
<td>14</td>
<td>104</td>
<td>M</td>
<td>2</td>
<td>136</td>
</tr>
<tr>
<td>15</td>
<td>104</td>
<td>M</td>
<td>3</td>
<td>130</td>
</tr>
<tr>
<td>16</td>
<td>104</td>
<td>M</td>
<td>4</td>
<td>128</td>
</tr>
<tr>
<td>17</td>
<td>105</td>
<td>F</td>
<td>1</td>
<td>142</td>
</tr>
<tr>
<td>18</td>
<td>105</td>
<td>F</td>
<td>2</td>
<td>136</td>
</tr>
<tr>
<td>19</td>
<td>105</td>
<td>F</td>
<td>3</td>
<td>128</td>
</tr>
<tr>
<td>20</td>
<td>105</td>
<td>F</td>
<td>4</td>
<td>125</td>
</tr>
</tbody>
</table>

Labels for the variables are:

PTID: Patient ID
SEX: Sex
VISIT: Visit Number
SYSBP: Systolic BP (In mmHg)
DIABP: Diastolic BP (In mmHg)

ODS TAGSETS

Use of ODS tagsets feature in SAS V9.1 is a great help for generating multiple reports in a single page. With the use of ActiveX device and Java functionality of SAS V9.1, it is easy to generate HTML output which can be internet ready output. With this method, multiple tabular and graphical outputs can be forced and set to different panels on a single HTML page. Once HTML page is created from SAS outputs, you can modify any of the output with the use of ActiveX and Java features without even modifying SAS code.

To start with, you need to divide a single page in different HTML panels. Then each panel has to be divided in number of cells by rows and columns from top left to bottom right direction. You can divide each page in multiple panels. Then you can divide each panel in multiple rows and each row can be divided in multiple cells by multiple columns in it. The beauty of this feature is that you can divide a single page in as many cells as needed based on the requirements.
In the figure below, a page is divided in two panels. Panel 1 has only one row and 3 columns and so Panel 1 is divided in 3 cells. Panel 2 is divided in 2 row and 3 columns and thus panel 2 has 6 cells.

![Diagram of two panels with cell divisions](image)

In this presentation, we will generate first panel and 3 cells in it and will see how different outputs will be placed in these 3 cells. To generate the cells, you need to use three ODS tagsets, one for panel, one for row and one for column. As with other ODS features, tagsets also have start and finish statements.

To create a panel, you need to start panel first, then a row and then a column as follow.

```sas
GOPTIONS Device = ActiveX;
GOPTIONS xpixels=360 ypixels=360;
ODS tagsets.HTMLpanel Path="/Pharmasug09"
File = "Vitals.html";
/* Start the Panel 1 */
ODS tagsets.HTMLpanel event = panel (start);
/* Start the Row 1 */
ODS tagsets.HTMLpanel event = row_panel (start);
/* Start the Column 1 */
ODS tagsets.HTMLpanel event = column_panel (start);
```

Above mentioned code will create Vitals.html file at the location mentioned in Path option using ActiveX device. Pixel size under GOPTIONS will control the size of each cell in the panel which can be changed according to desired cell size. ODS tagsets statements with event option will start panel first, then row in the panel and then first column in the panel. You can also define your own custom style using PROC TEMPLATE and then specify that style in the STYLE option along with Path and File options in the very first ODS tagsets statement.

Now you can add any SAS code which will generate some output and it will be placed in the first cell of the panel. I am just creating a simple listing of the data by following code.

```sas
Proc Print Data = Vitals;
Run;
```

Output from the above code will be placed in the first cell (Row 1, Col 1) of the first panel. Now to create and place another output in the next cell, you must close first cell and start the next cell as follow.
Above code will close the first cell and will start second cell by starting column 2.
Now I will calculate the mean of Systolic BP by visit and then plot the graph of mean by visit which will be placed in second cell (Row 1, Col 2) as follow.

```bash
/* Sort the data by visit */
Proc Sort Data = Vitals;
   by visit ptid;
Run;

/* Mean of SYSBP by Visit */
Proc Means Data = Vitals noprint;
   by visit;
   var sysbp;
   Output out = stat mean = MEAN;
Run;

/* Plot of Mean of SYSBP by Visit */
Proc Gplot Data = Stat;
   Plot MEAN * VISIT;
   Symbol1 v = dot Interpol=join;
Run; Quit;
```

Now you must close second cell to start third cell as follow.

```bash
/* Close the Column 2 */
ODS tagsets.HTMLpanel event = column_panel (finish);
/* Start the Column 3 */
ODS tagsets.HTMLpanel event = column_panel (start);
```

Now I will calculate the mean of Diastolic BP by visit and then plot the graph of mean by visit in the similar way as I did for Systolic BP which will be placed in third cell as follow.

```bash
/* Mean of DIABP by Visit */
Proc Means Data = Vitals noprint;
   by visit;
   var diabp;
   Output out = stat1 mean = MEAN;
Run;

/* Plot of Mean of DIABP by Visit */
Proc Gplot Data = Stat1;
   Plot MEAN * VISIT;
   Symbol1 v = dot Interpol=join;
Run; Quit;
```

Output generated by above code will be placed in third cell (Row 1, Col 3) of the first panel. Now that all three cells of the first panel have been filled, you can close this panel and proceed to the next panel and create different outputs in the similar way to place in different cells of the second panel. To close the first panel, you need to follow the same steps as we did for starting the panel, but it has to be in the reverse order. First close the row, then last column of the row and then close the panel as follow.
Above code will close the first panel. After this you can start second panel and process each cell of second panel in the similar way as we did for first panel. In this presentation, we created a combination of tabular and graphical outputs on the same HTML page, but you can create multiple tabular outputs or multiple graphical outputs alone and that too of different sizes on a single HTML page. You can also create a single HTML panel which has cells of different sizes in it based on your need of reporting.

The output of the first panel from above mentioned codes looks like following figure.

<table>
<thead>
<tr>
<th>Obs</th>
<th>PTID</th>
<th>SEX</th>
<th>VISIT</th>
<th>SYSBP</th>
<th>DIABP</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>101</td>
<td>M</td>
<td>1</td>
<td>130</td>
<td>84</td>
</tr>
<tr>
<td>2</td>
<td>101</td>
<td>M</td>
<td>2</td>
<td>125</td>
<td>82</td>
</tr>
<tr>
<td>3</td>
<td>101</td>
<td>M</td>
<td>3</td>
<td>124</td>
<td>52</td>
</tr>
<tr>
<td>4</td>
<td>101</td>
<td>M</td>
<td>4</td>
<td>120</td>
<td>80</td>
</tr>
<tr>
<td>5</td>
<td>102</td>
<td>F</td>
<td>1</td>
<td>130</td>
<td>85</td>
</tr>
<tr>
<td>6</td>
<td>102</td>
<td>F</td>
<td>2</td>
<td>119</td>
<td>80</td>
</tr>
<tr>
<td>7</td>
<td>102</td>
<td>F</td>
<td>3</td>
<td>122</td>
<td>81</td>
</tr>
<tr>
<td>8</td>
<td>102</td>
<td>F</td>
<td>4</td>
<td>105</td>
<td>72</td>
</tr>
<tr>
<td>9</td>
<td>103</td>
<td>F</td>
<td>1</td>
<td>138</td>
<td>88</td>
</tr>
<tr>
<td>10</td>
<td>103</td>
<td>F</td>
<td>2</td>
<td>127</td>
<td>84</td>
</tr>
<tr>
<td>11</td>
<td>103</td>
<td>F</td>
<td>3</td>
<td>114</td>
<td>75</td>
</tr>
<tr>
<td>12</td>
<td>103</td>
<td>F</td>
<td>4</td>
<td>121</td>
<td>80</td>
</tr>
<tr>
<td>13</td>
<td>104</td>
<td>M</td>
<td>1</td>
<td>135</td>
<td>80</td>
</tr>
<tr>
<td>14</td>
<td>104</td>
<td>M</td>
<td>2</td>
<td>136</td>
<td>86</td>
</tr>
<tr>
<td>15</td>
<td>104</td>
<td>M</td>
<td>3</td>
<td>130</td>
<td>64</td>
</tr>
<tr>
<td>16</td>
<td>104</td>
<td>M</td>
<td>4</td>
<td>126</td>
<td>62</td>
</tr>
<tr>
<td>17</td>
<td>105</td>
<td>F</td>
<td>1</td>
<td>142</td>
<td>91</td>
</tr>
<tr>
<td>18</td>
<td>105</td>
<td>F</td>
<td>2</td>
<td>136</td>
<td>86</td>
</tr>
<tr>
<td>19</td>
<td>105</td>
<td>F</td>
<td>3</td>
<td>128</td>
<td>83</td>
</tr>
<tr>
<td>20</td>
<td>105</td>
<td>F</td>
<td>4</td>
<td>125</td>
<td>83</td>
</tr>
</tbody>
</table>

Also, another beauty here is the ActiveX capability. As mentioned earlier, these graphical outputs are generated using ActiveX device which gives another powerful tool to you right in your hand without any extra efforts. Once a graphical output is generated using SAS code, ActiveX gives wide range of choices for the modification of graphical output without changing a single line of SAS code. You will be amazed to see how easily you can modify a graphical output with just a click of mouse as described in following figures.
Above plot of mean of Systolic BP by visit is changed in to vertical bar chart as follow by selecting Vertical Bar as Chart Type from right click menu.

You can change many other countless attributes of graphical output with this capability.

Thus, a comparison of long list of graphical and tabular outputs, which might take hours of efforts easily, is greatly simplified when HTML panels are used. This method enables you to present enormous amount of information in different formats on the same page which makes comparison of even a large size data an easy task.

**SAS/GRAPH® with ODS PDF**

Another method to generate multiple graphical outputs on the same page using SAS/GRAPH software along with ODS PDF destination is pretty straightforward and simple. ODS PDF destination file can be generated either by PDF or PDFC device which can be specified in device option of GOPTIONS.
SAS/GRAPH has options available which can control the size of the graphical output and also the position of graphical output on the page. SAS/GRAPH options like HSIZE and VSIZE control the size of graphical output. SAS/GRAPH options like HORIGIN and VORIGIN control the position of the graphical output on a page. By default, ODS PDF will go to new page each time new graphical output is generated. But STARTPAGE=never option in ODS PDF tells SAS not to go to new page even if new graphical output is started. Thus, in conjunction with ODS PDF, SAS/GRAPH software can easily generate multiple charts and graphs on a single page.

For presentation, we will use the same vitals data and we will generate four different plots on a same page. First we will generate plots of Mean of Systolic BP and Diastolic BP by Sex and then plots of Mean of Systolic BP and Diastolic BP by Visit. We have already calculated means by visit in previous section and so in this section we will only calculate means by sex and then we will plot all four graphical outputs on a single page. Following code illustrates how to generate these four plots on a single page.

```sas
OPTIONS orientation=landscape;
GOPTIONS reset = all Device = PDFC;
GOPTIONS hsize = 4in vsize = 3in;
ODS PDF File = "C:\Documents and Settings\Pharmasug09\vitals.pdf"
    Startpage = never;
/* Sort the data by sex */
Proc Sort Data = Vitals;
    by sex ptid;
Run;
/* Mean of SYSBP by Sex */
Proc Means Data = Vitals noprint;
    by sex;
    var sysbp;
    Output out = stat2 mean = MEAN;
Run;
/* Mean of DIABP by Sex */
Proc Means Data = Vitals noprint;
    by sex;
    var diabp;
    Output out = stat3 mean = MEAN;
Run;
/* Plot of Mean of SYSBP by Sex */
Proc Gplot Data = Stat2;
    GOPTIONS horigin = 0in vorigin = 1in;
    Plot MEAN * SEX;
    Symbol1 v = dot Interpol=join;
    Title 'Plot of Mean of SYS BP by Sex';
Run; Quit;
/* Plot of Mean of DIABP by Sex */
Proc Gplot Data = Stat3;
    GOPTIONS horigin = 4.5in vorigin = 1in;
    Plot MEAN * SEX;
    Symbol1 v = dot Interpol=join;
    Title 'Plot of Mean of DIA BP by Sex';
Run; Quit;
```
/* Plot of Mean of SYSBP by Visit */
Proc Gplot Data = Stat;
GOPTIONS horigin = 0in vorigin = 5in;
   Plot MEAN * VISIT;
   Symbol1 v = dot Interpol=join;
   Title 'Plot of Mean of SYS BP by Visit';
Run; Quit;

/* Plot of Mean of DIABP by Visit */
Proc Gplot Data = Stat1;
GOPTIONS horigin = 4.5in;
   Plot MEAN * VISIT;
   Symbol v = dot Interpol=join;
   Title 'Plot of Mean of DIA BP by Visit';
Run; Quit;

ODS PDF close;

Above code is generating four plots of the size 4in. by 3in. and their respective position on the page is specified in terms of HORIGIN and VORIGIN in respective PROC GPLOT. Output of above code looks like following figure.

You can always change the number of graphical outputs on a page, size of graphical output and position of graphical output by playing with graphic options like HSIZE, VSIZE, HORIGIN and VORIGIN.
ODS LAYOUT

One another fine feature of SAS V9.1 is ODS LAYOUT. This feature allows you to generate multiple graphical and tabular outputs on a single page in different regions.

One limitation is that this method works best with PDF format only. ODS LAYOUT has two different ways: Absolute Layout and Gridded (relative) Layout. Gridded layout should work fine with HTML format also, but absolute layout will not support HTML format at all. Both of these methods divide the layout in regions and then put different outputs in these regions.

Absolute Layout

In absolute layout, you need to specify the horizontal and vertical space available on a page in the ODS LAYOUT statement with the use of WIDTH and HEIGHT options. Then you need to specify where to start the region in the form of x and y coordinates in the ODS REGION statement. You also need to specify the size of region in the same statement with the use of WIDTH and HEIGHT options. Size of layout and region can be specified in inches, centimeters or in the form of percentages. Same way position of region can also be specified in inches, centimeters or percentages of total space available on horizontal and vertical axes. All these measurements calculate the distance in specified unit from top left corner of the page. As any other ODS feature, layout also has start and end statements.

For presentation purpose, we will use the same data that we have used in previous two sections. Following code will generate a tabular output of vitals data and plot of mean of Systolic BP by visit on the same page.

```odslayout
ODS PDF File = "C:\Documents and Settings\Pharmasug09\layout.pdf"
    Startpage = no;

    /* Start the Layout */
    ODS Layout start width = 6in height = 6in;

    /* Start first region */
    ODS Region width = 3in height = 6in x = 0in y = 0in;

    /* Tabular output of Vitals Data */
    Proc Print Data = Vitals;
    Run;

    /* Start second region */
    ODS Region width = 3in height = 3in x = 3in y = -6in;

    /* Plot of Mean of SYSBP by Visit */
    Proc Gplot Data = Stat;
        Plot MEAN * visit;
        Symbol1 v = dot Interpol=join;
    Run; Quit;

    /* Close the Layout */
    ODS layout end;

    ODS pdf close;
```

The output of above code will look like following figure.

The drawback of absolute layout is that one output can be overlapped by another output if region specifications are not specified accurately. Also if output is big and specified region is not big enough, it will simply truncate the output.
Gridded Layout

With gridded layout method, drawbacks of absolute method can be overcome. In this method you simply tell ODS how many columns and rows you want and it will make each grid big enough to accommodate the whole output. In this method, if you don’t specify region, it will put output of each procedure in next available region from left to right and top to bottom direction. If you want specific output in a specific region then you can simply specify the region by COLUMN and ROW option when you start the region. Also you can control the size of each grid by COLUMN_WIDTHS and ROW_HEIGHTS options in ODS LAYOUT statement itself. Output similar to above figure can be generated with gridded layout and code can be reduced to the below code.

```sas
/* Start the Layout */
ODS Layout start columns = 2;

/* Tabular output of Vitals Data */
Proc Print Data = Vitals;
Run;

/* Plot of Mean of SYSBP by Visit */
Proc Gplot Data = Stat;
   Plot MEAN * visit;
   Symbol1 v = dot Interpol=join;
Run; Quit;

/* Close the Layout */
ODS layout end;
```
There are many other options available in ODS Layout and ODS Region statements which can take care of many other aspects of the multiple outputs on a single PDF page. All different options can be used either in absolute or in gridded layout and detail discussion on the use of all these options is out of the scope of this paper.

CONCLUSION

As said before, SAS always offers multiple solutions to any problem and there are multiple ways to get multiple outputs on a single page. ODS has been a great help to programmers of all levels who want to enhance and customize their reports generated in day to day activity. This compilation is just an effort to put together different ways and basic concepts to generate multiple reports on a single page in your choice of destination. Some methods explained in this paper are in the experimental phase and they have to pass through more testing. Some features might be subject to failure or unexpected result if stressed out, but you can use them as building blocks and play around at least to start with.

REFERENCES

- SAS V9.1 Online documentation

CONTACT INFORMATION

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

Name: NIRAJ J PANDYA
Phone: 201-936-5826
E-mail: niraj_mech@yahoo.com

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.