Use of Styles in Graphics
Jeff Cartier, SAS Institute Inc., Cary, NC

INTRODUCTION

Today, most SAS users are taking advantage of ODS to produce documents containing output from SAS procedures. Users are aware of the existence of ODS styles and how a style can be specified to alter the fonts, colors, and other appearance aspects of their tabular output. The good news is that in SAS 9, graphical output can now be formatted in a similar fashion with an ODS style.

ODS styles have been extended to include elements that affect graphical procedure output as well as tabular output. This paper will show how to easy it is to apply any of the new supplied style definitions to SAS/GRAPH, SAS/STAT, SAS/ETS output. You will also see how SAS/GRAPH coding and supplied STATGRAPH templates interact with information supplied by a style. By adjusting source programs, you can control exactly the level of style information that contributes to final output.

ODS AND SAS/GRAPH OUTPUT

Many of the new styles offer graphical visual effects such color gradients, transparency, texture maps, shadow effects and anti-aliasing on text. To see style effects for SAS/GRAPH procedures in SAS9.1, the graphics device driver must be set to ACTIVEX, JAVA, ACTXIMG, or JAVAIMG (the first two drivers create interactive controls and the last two drivers create images). The following example illustrates how easy it is to use a style with ODS and how the style produces a coordinated visual effect on both graphical and tabular output:

```
ods html file='default.html' style=default;
  goptions reset=all border device=actximg;
  proc gchart data=sashelp.class;
      vbar3d sex / sumvar=height type=mean outside=mean;
  run; quit;
  proc means data=sashelp.class maxdec=1 nonobs mean;
      class sex;
      var height;
  run;
  ods html close;
```

```
< same program as above >
```

```
ods html file='RSVP.html' style=RSVP;
  ods html close;
```

Notice that the SAS/GRAPH coding did not include any options that specified fonts or colors to be used. This information was all derived from the style definition. If such options were present, the colors or fonts in the program would be used in place of the corresponding style values. The example program use ODS HTML destination, but it could have used any other ODS destination just as well, such as PDF, RTF, or PRINTER. All produce different output files with the same visual content.

If you have not used any of the four client drivers before, here are some other things you should know:
A client technology (ActiveX or Java) is used to render the graph, not SAS/Graph. Consequently, there may be some differences in appearance between client and non-client drivers.

- The only supported procedures are GCHART, GPLOT, GMAP, GCONTOUR, and G3D. You can also use SAS/GRAPH annotation coding with these procedures.
- Titles and footnotes appear in the output but are not part of the graph.
- The interactive client drivers (ACTIVEX and JAVA) enable you to change the graphical display via context menus.
- There are some differences in which SAS/GRAPH options are supported by Java and ActiveX technologies. See the SAS/GRAPH documentation for details.
It should be emphasized that when using SAS/GRAPH procedures with ODS, a SAS/GRAPH device driver is always in effect.

In SAS 9.1, you must use one of the “client drivers” (ACTIVEX, JAVA, ACTXIMG, or JAVAIMG) to see the effect of a style. All other drivers are “style unaware”. For example, if you were to use any of the GIF family of drivers, the ODS output would look like just like the GRSEG output, but as a GIF image. Its visual appearance is affected only by SAS/GRAPH coding and not by any ODS style.

Starting in SAS 9.2, other device drivers will use styles if you include a new GOPTION STYLE=. The following programs illustrate how this will work:

This is SAS9.2 coding showing the STYLE= goption. It produces an image named SASGRAPH.PNG in the listing destination.

```sas
ods html file='default.html' style=RSVP;

goptions reset=all border device=png style=RSVP;
proc gchart data=sashelp.class;
  vbar3d sex / sumvar=height type=mean outside=mean;
run; quit;

goptions reset=all border device=png style=RSVP;
proc gchart data=sashelp.class;
  vbar3d sex / sumvar=height type=mean outside=mean;
run; quit;
proc means data=sashelp.class maxdec=1 nonobs mean;
  class sex;
  var height;
run;

ods html close;
```

**ODS STATISTICAL GRAPHICS OUTPUT**

In Version 9, SAS/STAT and SAS/ETS procedures can produce graphs when used with ODS. Here is an example of using PROC LIFETEST to produce a survival plot showing the Hall-Wellner band (HWB).

```
ods html style=analysis;

ods graphics on;
proc lifetest data=mydata;
  time surv*censor(1);
  survival plots=(hwb);
  strata treatment;
run;
ods graphics off;

ods html close;
```

(Tabular output not shown.)
Here are some facts about the ODS Statistical Graphs produced by SAS/STAT and SAS/ETS:

- Graphs are produced by entirely Java technology. They do not support any form of SAS/GRA PHS coding, including device drivers.
- Graphs are not produced by default. You must enable / disable graphics with the ODS GRAPHICS statement.
- Statistical procedures supply one or more ODS STATGRAPH templates that specify a predefined graph. You simply instruct the procedure which graphs to produce.
- The supplied STATGRAPH templates use ODS styles to set colors, fonts, and as well as other appearance features such as markers and line patterns.

SUPPLIED STYLES
To view the supplied ODS styles, issue the ODSTEMPLATE command from your Display Manager session. If you have not created any of your own styles, you will see a single node for SASHELP.TEMPLMST under the TEMPLATES tree. Expand this node to see all supplied template folders. Select STYLES to display the contents of this folder. In addition to the styles provided in Version 8, there are many new styles in Version 9, including:

Analysis  Astronomy  Banker  BlockPrint  Curve  Gears  Education  Electronics
Magnify  Money  RSVP  Science  Sketch  Statistical  Tom  Watercolor

DEFINING COLORS AND FONTS IN A STYLE
A major part of any style definition establishes colors and fonts for specific areas of the output, Titles, BY lines, Tables, etc. If you examine the style, you will see lists of colors assigned to “abstract” names. These names are referenced in other style elements. Here are examples of such lists:

Fonts and Color for Tables, Titles, BY Lines, etc.

```
style fonts "Fonts used in the default style" /
  'TitleFont' = ("Arial",5,Bold Italic)
  'TitleFont2' = ("Arial",4,Bold Italic)
< other fonts >
  'headingFont' = ("Arial",4,Bold)
  'docFont' = ("Arial",3);
```

```
style color_list "Colors for default style" /
  'fgA1' = cx000000 /*foreground */
  'bgA1' = cxF0F0F0 /* background */
< other colors >
  'fgA' = cx002288
  'bgA' = cxE0E0E0;
```

```
style colors "Abstract colors" /
  'tableborder' = color_list('fgA1')
  'tablebg' = color_list('bgA1')
  'docfg' = color_list('fgA')
  'docbg' = color_list('bgA');
```

```
style container /
  font = Fonts('DocFont')
  foreground = colors('docfg')
  background = colors('docbg');
```

```
style Table from output /
  background = colors('tablebg');
```

Fonts and Colors for Graphs

```
style GraphFonts "Fonts used in graph styles"/
  'GraphDataFont' = ("Arial",8pt)
  'GraphValueFont' = ("Arial",10pt)
  'GraphLabelFont' = ("Arial",12pt,Bold)
  'GraphFootnoteFont' = ("Arial",12pt,Bold)
  'GraphTitleFont' = ("Arial",14pt,Bold);
```

```
style GraphColors "Abstract graph colors"/
  'glabel' = cx000000
  'gaxis' = cx000000
< other colors >
  'gdata1' = cx6173A9
  'gdata2' = cx8DA642
  'gdata3' = cx98341C
  'gdata4' = cxFDC861;
```

```
style GraphBackground /
  background = colors('docbg');
```

```
style GraphAxisLines /
  foreground = GraphColors('gaxis');
```

```
style GraphLabelText /
  font = GraphFonts('GraphLabelFont');
```

Notice that various style elements may reference the same color or font. If you want to change fonts or colors in a style, it is recommended that you change only the font or color values (but not their abstract names) in elements Fonts, GraphFonts, Color_List, and GraphColors. This ensures a consistent effect is created across tables and graphs. Color values can be specified in many ways including SAS color names, RGB or HLS. Consult the ODS documentation for examples. When testing the appearance of modified colors and fonts, you should include both graphs and tables to assure that you get the desired consistency for both forms of output.
GRAPHICAL STYLE ELEMENTS IN VERSION 9

All styles incorporate a large number of graphically-related style elements that better coordinate the appearance of graphical and tabular output. There are tables at the end of the paper that summarize graphical style elements and style attributes.

This figure shows the names of some of the graphical style elements and indicates the aspects of a graph affected by each.

Most of the style element names are self-explanatory.

The elements GraphData1 – GraphData12 are used to associate style attributes with sets of data values. The properties for each level of the subgroup variable are obtained from the GraphData elements. These elements can specify not only colors, but also line and marker properties for plots.

The remainder of this paper shows how customize the appearance of graphs in your ODS output by adapting supplied styles. We will modify the supplied STYLES.CURVE as our starting point (parent) and name our style STYLES.MYCURVE:

```solidus
proc template;
define style Styles.myCurve;
parent = styles.Curve;
/* style statements defined below */
end;
run;
```

Changing graph size

Adding OUTPUTWIDTH= and OUTPUTHEIGHT= to the Graph element forces all graphs produced with this style to be of a given size.

```solidus
/* add to mycurve style definition */
style Graph from Graph /
outputwidth = 400px
outputheight = 400px;
```

You can use other units of measurement such as IN or CM. The default size of all graphs is OUTPUTWIDTH=640px and OUTPUTHEIGHT=480.

Using Transparency

One of the more interesting style attributes is **transparency**. This affects how much you can “see through” portions of a chart to the graph background (see above).

The CURVE style employs transparency with two elements:

```solidus
style GraphCharts from
GraphCharts /
transparency = 0.1;

style GraphWalls from
GraphWalls /
transparency = 1.0;
```

transparency=0 for chart and wall

transparency=1 for chart and wall
Changing Graph Background

The CURVE style uses this definition for the GraphBackground element:

```plaintext
replace GraphBackground / 
    background = colors('docbg') 
    image = "Curve.jpg" 
    just = Right 
    vjust = Bottom;
```

CURVE.JPG is one of several image files supplied with base SAS that are used with style definitions. The location of these files is defined by the system option TEXTURELOC=. You can add your own images to the TEXTURELOC path, and refer to them without path information or you can include the fully-qualified name (or URL) to your own image. Filetypes are not restricted to JPG.

This output shows a corporate logo used for the IMAGE attribute. The image is displayed using its actual size. The JUST attribute (LEFT, CENTER, RIGHT) and VJUST attribute (TOP, MIDDLE, BOTTOM) control its position.

A related attribute is BACKGROUNDIMAGE. This differs from IMAGE in that it specifies an image to be stretched to fit the entire background. VJUST and JUST do not apply to BACKGROUNDIMAGE.

ADJUSTING SAS/GRAPH PROGRAMS FOR STYLES

Recall the SAS/GRAPH coding of our original program:

```plaintext
goptions reset=all border dev=actximg; 
proc gchart data=sashelp.class;
    vbar3d sex / sumvar=height type=mean 
        outside=mean; 
    run; quit;
```

Notice that this program does NOT contain any of the numerous SAS/GRAPH options that change colors or fonts of the output. If any of these options were to appear in the program, they would have precedence over any style attribute that may address the same feature.

In general, a style does not enable a SAS/GRAPH feature – you must do this in your SAS/GRAPH program. Examples of this include GOPTIONS BORDER | NOBORDER to enable or disable a border around the graph. If you enable the border, the Output and Graph styles elements control its visual characteristics (which are coordinated with the table border in the supplied styles).

Another example is the FRAME | NOFRAME option used by GCHART and GPLOT action statements. In general, you only need to enable or disable this feature. If you use CFRAME to turn on the frame you will also override the color defined in the style. Here is a list of some common SAS/GRAPH options that affect the same graph features that graphical style elements do:

<table>
<thead>
<tr>
<th>GOPTIONS</th>
<th>AXIS</th>
<th>LEGEND</th>
<th>GPLOT – PLOT / BUBBLE</th>
<th>GCHART – VBAR/HBAR/ VBAR3D/HBAR3D</th>
</tr>
</thead>
<tbody>
<tr>
<td>COLORS</td>
<td>AXST</td>
<td>LEGEND</td>
<td>GAXIS CFRAME</td>
<td>GAXIS CFRAME</td>
</tr>
<tr>
<td>HSIZE</td>
<td>AXST</td>
<td>LEGEND</td>
<td>CBORDER</td>
<td>CTEXT IFRAME</td>
</tr>
<tr>
<td>VSIZE</td>
<td>AXST</td>
<td>LEGEND</td>
<td>CSHADOW</td>
<td>CAUTOHREF</td>
</tr>
<tr>
<td>XPIXELS</td>
<td>AXST</td>
<td>LEGEND</td>
<td>FWIDTH</td>
<td>CAUTOVREF</td>
</tr>
<tr>
<td>YPIXELS</td>
<td>AXST</td>
<td>LEGEND</td>
<td>LABEL=(COLOR</td>
<td>LAUTOVREF</td>
</tr>
<tr>
<td>IBACK</td>
<td>AXST</td>
<td>LEGEND</td>
<td>FONT HEIGHT)</td>
<td>COUTLINE</td>
</tr>
<tr>
<td>CTEXT</td>
<td>AXST</td>
<td>LEGEND</td>
<td>VALUE=(COLOR</td>
<td>CTEXT IFRAME</td>
</tr>
<tr>
<td>CTITLE</td>
<td>AXST</td>
<td>LEGEND</td>
<td>FONT HEIGHT)</td>
<td>CTEXT IFRAME</td>
</tr>
<tr>
<td>CBACK</td>
<td>AXST</td>
<td>LEGEND</td>
<td>VALUE=(COLOR</td>
<td>LAUTOHREF</td>
</tr>
<tr>
<td>CSYMBOL</td>
<td>AXST</td>
<td>LEGEND</td>
<td>FONT HEIGHT)</td>
<td>LAUTOHREF</td>
</tr>
<tr>
<td>CPATTERN</td>
<td>AXST</td>
<td>LEGEND</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FTEXT</td>
<td>AXST</td>
<td>LEGEND</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FTITLE</td>
<td>AXST</td>
<td>LEGEND</td>
<td></td>
<td></td>
</tr>
<tr>
<td>HTEXT</td>
<td>AXST</td>
<td>LEGEND</td>
<td></td>
<td></td>
</tr>
<tr>
<td>HTITLE</td>
<td>AXST</td>
<td>LEGEND</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

5
ODS STATISTICAL GRAPHICS
As mentioned earlier, you can create one or more graphs for statistical procedures, independent of SAS/GRAPH.

```
ods graphics on;
ods html style=statistical;
proc loess data=ENSO;
   model Pressure=Month/ clm alpha=0.01 ;
run;
ods html close;
ods graphics off;
```

(Only output from the FIT template shown)

ODS Statistical Graphics output uses mostly the same graphical style elements and attributes that SAS/GRAPH does. There are a few style attributes that STATGRAPH templates do not support (such as those for image gradient backgrounds). But there are also several additional style elements that apply only to STATGRAPH templates. If you look in the DEFAULT style, you will see all of these elements. For example, these style elements control the appearance of fit lines and confidence bands / lines:

```
style StatGraphFitLine from GraphComponent /
   transparency = 0.00
   linethickness = 2px
   linestyle = 1
   contrastcolor = GraphColors('gcfit');
```

```
style StatGraphConfidence from GraphComponent
   "Foreground for band fill"
   "ContrastColor for lines" /
   transparency = 0.50
   linethickness = 1px
   linestyle = 34
   contrastcolor = GraphColors('gcconfidence')
   foreground = GraphColors('gcconfidence');
```

If you were to change the colors associated with these styles elements in some style (say, Statistical) you would change the appearance of all statistical plots with fit lines and / or confidence bands, regardless of the procedure that produced them.

CONCLUSION
In Version 9, you will be able to control the appearance of graphs as well as tables in your ODS output. ODS will provide many new styles. You can define your own styles to create many interesting effects.

The two tables that follow document the Version 9 style elements and attributes. These tables also relate style elements and attributes to SAS/GRAPH syntax features so you can more easily adjust your programs to use more (or less) of the style definition in any particular program.

CONTACT INFORMATION
Your comments and questions are valued and encouraged. Contact the author at:
   Jeff Cartier
   SAS Institute Inc.
   Cary NC 27513
   Work Phone:
   Email: Jeff.Cartier@sas.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.
<table>
<thead>
<tr>
<th>Style Element</th>
<th>Affects</th>
<th>Style Attributes</th>
<th>SAS/Graph Override</th>
</tr>
</thead>
<tbody>
<tr>
<td>Graph</td>
<td>Graph size, border around graph</td>
<td>OutputWidth, OutputHeight, BorderWidth, BorderColor, CellSpacing, CellPadding</td>
<td>GOPTIONS XPIXELS=, YPIXELS=; GOPTIONS BORDER= must be in effect to enable the border effects</td>
</tr>
<tr>
<td>GraphCharts</td>
<td>all charts in graphics area</td>
<td>Transparency</td>
<td></td>
</tr>
<tr>
<td>GraphBackground</td>
<td>background color or image of the graph</td>
<td>Gradient_Direction, StartColor, EndColor, Background, BackgroundImage, Image, Vjust, Just</td>
<td>GOPTIONS CBACK=; IBACK= IMAGESTYLE=</td>
</tr>
<tr>
<td>GraphLegendBackground</td>
<td>background color or image of the legend</td>
<td>Gradient_Direction, StartColor, EndColor, Background, BackgroundImage, Image, Vjust, Just</td>
<td>LEGEND statement CFRAME=; CBLOCK=</td>
</tr>
<tr>
<td>DropShadowStyle</td>
<td>drop shadow color for text</td>
<td>DropShadow, ForeGround</td>
<td></td>
</tr>
</tbody>
</table>
| GraphLabelText         | text for axis labels and legend title        | ForeGround, Font_Face, Font_Size, Font_Weight, Font_Style, DropShadow              | GOPTIONS FTEXT=; CTXT=; GOPTIONS FRAME=; AXIISTMENT LABEL=() options COLOR=, FONT= HEIGHT=; /
| GraphValueText         | text for axis tick marks values and legend entries | ForeGround, Font_Face, Font_Size, Font_Weight, Font_Style, DropShadow              | GOPTIONS FTEXT=; CTXT=; GOPTIONS FRAME=; AXIISTMENT LABEL=() options COLOR=, FONT= HEIGHT=; /
| GraphGridLines         | grid / reference lines                       | ForeGround, LineStyle, OutputWidth                                               | AXIISTMENT COLOR= , STYLE=, WIDTH= options                                           |
| GraphAxisLines         | axis lines and tick marks                   | ForeGround, LineStyle, OutputWidth                                               | Procedure CAXIS=; AXIISTMENT COLOR= , STYLE=, WIDTH=                                 |
| GraphBorderLines       | frame around axis area and legend            | ForeGround, LineStyle, OutputWidth                                               | Chart FRAME option, LEGEND statement CBORDER= FWIDTH=                                |
| GraphOutlines          | lines that outline bars, map regions, etc.   | ForeGround, LineStyle, OutputWidth                                               | Pattern statement                                                                  |
| GraphWalls             | wall color or image                          | Transparency, StartColor, EndColor, Gradient_Direction, Background, BackgroundImage, Image | Procedure action statement IFRAME= IMAGESTYLE= CFRAME= options                      |
| GraphFloor             | floor color or image                         | Transparency, StartColor, EndColor, Gradient_Direction, Background, BackgroundImage, Image |                                                                                     |
| TwoColorRamp           | maps with continuous response                | StartColor, EndColor                                                              |                                                                                     |
| GraphData1             | graphics primitives related to data items:  | Foreground, ContrastColor, MarkerSymbol, MarkerSize, LineStyle, LineThickness    | GOPTIONS COLORS=( ); SYMBOL statement; Pattern statement                            |
| GraphData12            | color, fill, marker                          |                                                                                  |                                                                                     |

Table 1  Version 9 Graphical Style Elements

Note: Style elements include all recognized attributes. Style elements do not have to define all attributes.
<table>
<thead>
<tr>
<th>Style Attribute</th>
<th>Type</th>
<th>Affects</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>OutputWidth</td>
<td>dimension</td>
<td>width of graph; line thickness</td>
<td>OutputWidth=400px OutputWidth=2in</td>
</tr>
<tr>
<td>OutputHeight</td>
<td>dimension</td>
<td>height of graph</td>
<td>OutputHeight=300px OutputHeight=4cm</td>
</tr>
<tr>
<td>Transparency</td>
<td>number: 0.0=opaque 1.0=transparent</td>
<td>Chart, walls, floor and legend backgrounds</td>
<td>Transparency=0.2</td>
</tr>
<tr>
<td>Background</td>
<td>color</td>
<td>background color of the graph, walls, or floor</td>
<td>Background=colors('docbg')</td>
</tr>
<tr>
<td>Foreground</td>
<td>color</td>
<td>color of text, data fill item</td>
<td>Foreground=colors('docfg')</td>
</tr>
<tr>
<td>ContrastColor</td>
<td>color</td>
<td>alternate color for maps; marker color</td>
<td>ContrastColor=red</td>
</tr>
<tr>
<td>LineStyle</td>
<td>integer: 1 = solid line 2-46= dashed line</td>
<td>borders, axis lines, grid, reference, model, confidence lines</td>
<td>LineStyle=2</td>
</tr>
<tr>
<td>LineThickness</td>
<td>dimension</td>
<td>thickness of line</td>
<td>LineThickness=2px</td>
</tr>
<tr>
<td>DropShadow</td>
<td>boolean: On or Off</td>
<td>drop shadow color for text</td>
<td>DropShadow=on DropShadow=off</td>
</tr>
<tr>
<td>BackGroundImage</td>
<td>string: image file (including path)</td>
<td>image that can be stretched, but not positioned in graph, chart, walls, floor</td>
<td>Image=://server/images/myimage.gif</td>
</tr>
<tr>
<td>Image</td>
<td>string: image file (including path or URL)</td>
<td>image that can be positioned, but not stretched in graph, chart, walls, floor</td>
<td>Image= &quot;<a href="http://www.a.com/~images/pic.gif">http://www.a.com/~images/pic.gif</a>&quot;</td>
</tr>
<tr>
<td>Just</td>
<td>justification: center, left, or right</td>
<td>image horizontal positioning</td>
<td>Just=left</td>
</tr>
<tr>
<td>Vjust</td>
<td>justification: top, middle, bottom</td>
<td>image vertical positioning</td>
<td>Vjust=bottom</td>
</tr>
<tr>
<td>Gradient_Direction</td>
<td>string: use “Xaxis” for left-to-right; “Yaxis” for top-to-bottom</td>
<td>graph background, legend background, charts, walls, floors</td>
<td>Gradient_Direction=&quot;Xaxis&quot;</td>
</tr>
<tr>
<td>StartColor</td>
<td>color: initial color used with gradient</td>
<td>graph background, legend background, charts, walls, floors</td>
<td>StartColor=yellow</td>
</tr>
<tr>
<td>EndColor</td>
<td>color: final color used with gradient</td>
<td>graph background, legend background, charts, walls, floors</td>
<td>StartColor=red</td>
</tr>
<tr>
<td>MarkerSymbol</td>
<td>string</td>
<td>markers related to data values</td>
<td>MarkerSymbol=&quot;circle&quot; MarkerSymbol=&quot;square&quot;</td>
</tr>
<tr>
<td>MarkerSize</td>
<td>dimension</td>
<td>marker size related to data values</td>
<td>MarkerSize=5px MarkerSize=3%</td>
</tr>
<tr>
<td>Font_Face</td>
<td>string</td>
<td>value text, label text</td>
<td>Font_Face=&quot;Helvetica&quot;</td>
</tr>
<tr>
<td>Font_Size</td>
<td>fontsize: 1 to 7 or dimension</td>
<td>value text, label text</td>
<td>Font_Size=3 Font_Size=10pt</td>
</tr>
<tr>
<td>Font_Weight</td>
<td>fontweight: light, medium, bold, etc.</td>
<td>value text, label text</td>
<td>Font_Weight=bold</td>
</tr>
<tr>
<td>Font_Style</td>
<td>fontstyle: italic, roman, slant</td>
<td>value text, label text</td>
<td>Font_Style=Italic</td>
</tr>
<tr>
<td>Font</td>
<td>Aggregate definition in parentheses</td>
<td>value text, label text</td>
<td>Font=(&quot;arial, helvetica&quot;, 4, medium roman)</td>
</tr>
</tbody>
</table>

Table 2  Version 9 Graphical Style Attributes