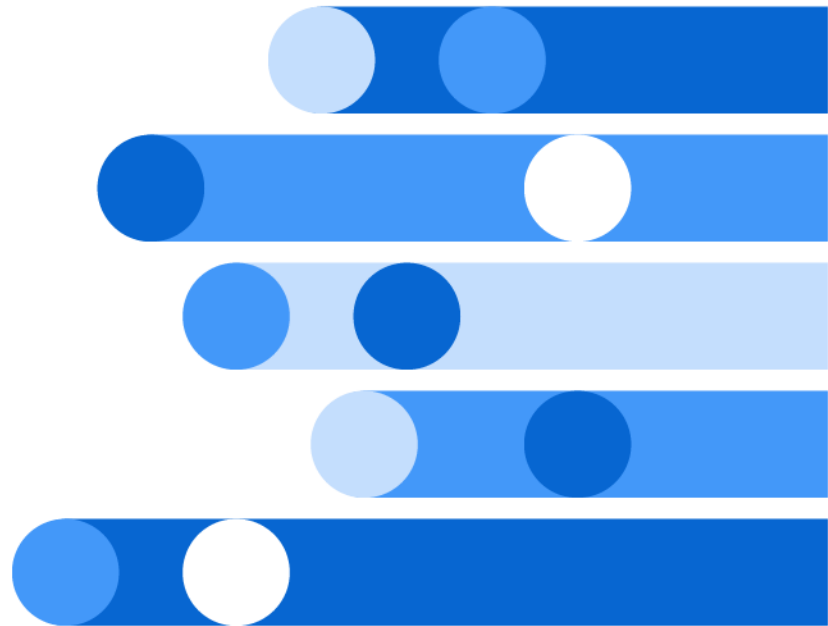




SAS[®] Viya[®] Platform Universal Printing

2020.1-2026.06*



* This document might apply to additional versions of the software. Open this document in [SAS Help Center](#) and click on the version in the banner to see all available versions.



The correct bibliographic citation for this manual is as follows: SAS Institute Inc. 2020. *SAS® Viya® Platform Universal Printing*. Cary, NC: SAS Institute Inc.

SAS® Viya® Platform Universal Printing

Copyright © 2020, SAS Institute Inc., Cary, NC, USA

All Rights Reserved. Produced in the United States of America.

For a hard copy book: No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, or otherwise, without the prior written permission of the publisher, SAS Institute Inc.

For a web download or e-book: Your use of this publication shall be governed by the terms established by the vendor at the time you acquire this publication.

The scanning, uploading, and distribution of this book via the Internet or any other means without the permission of the publisher is illegal and punishable by law. Please purchase only authorized electronic editions and do not participate in or encourage electronic piracy of copyrighted materials. Your support of others' rights is appreciated.

U.S. Government License Rights; Restricted Rights: The Software and its documentation is commercial computer software developed at private expense and is provided with RESTRICTED RIGHTS to the United States Government. Use, duplication, or disclosure of the Software by the United States Government is subject to the license terms of this Agreement pursuant to, as applicable, FAR 12.212, DFAR 227.7202-1(a), DFAR 227.7202-3(a), and DFAR 227.7202-4, and, to the extent required under U.S. federal law, the minimum restricted rights as set out in FAR 52.227-19 (DEC 2007). If FAR 52.227-19 is applicable, this provision serves as notice under clause (c) thereof and no other notice is required to be affixed to the Software or documentation. The Government's rights in Software and documentation shall be only those set forth in this Agreement.

SAS Institute Inc., SAS Campus Drive, Cary, NC 27513-2414

June 2026

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.

v_001-P1:uprint

Contents

<i>About This Book</i>	v
Chapter 1 / Overview of Universal Printing	1
What Is Universal Printing?	1
Universal Printing Output Formats	2
Universal Printing and ODS	2
Color Support for Universal Printers	3
Chapter 2 / Setting Up and Managing Universal Printing	11
Setting Up the Universal Printing Interface and the Default Printing Environment	11
Viewing Universal Printer Settings	12
Modifying Universal Printing Printer Settings	13
Managing Universal Printers Using SAS Procedures	14
Chapter 3 / Using Universal Printing	17
Viewing Universal Printers and Printer Prototypes	17
Specifying the Page Orientation for Universal Printing Documents	18
Embedding Non-Viewable Comments in Universal Printing Output	19
Chapter 4 / Managing Fonts in Universal Printing	21
Rendering Fonts	21
The FONTEMBEDDING System Option	22
ODS Styles and TrueType Fonts	23
Portability of TrueType Fonts	23
International Character Support	24
TrueType Fonts Supplied by SAS	24
Registering Fonts	26
Listing the Registered Fonts for a Device or Printer	27
Specifying Fonts	29
Chapter 5 / Creating Files Using Universal Printing	33
Creating PDF Files Using Universal Printing	34
Creating SVG (Scalable Vector Graphics) Files Using Universal Printing	37
Creating TIFF Images Using Universal Printing	69
Creating Animated GIF Images and SVG Documents	72
Chapter 6 / Dictionary of Printing-Related System Options	79
About SAS System Options	80
Summary of System Options that Control Universal Printing	81
Dictionary	83

About This Book

About Examples and Sample Data Sets

This document includes examples that illustrate some of the features of a statement or system option. Many examples process sample data contained in a SAS data set stored in the Sashelp library. The documentation for each example provides the data set name used. To see a brief description of any SAS data set in the Sashelp library, as well as output displaying the first five observations in each data set, please refer to [Sashelp Data Sets](#).

Recommended Reading

Here is the recommended reading list for this title:

- [SAS Programmer's Guide: Essentials](#)
- [Learning SAS by Example](#)
- [SAS Output Delivery System: User's Guide](#)
- [SAS System Options: Reference](#)
- [SAS Data Set Options: Reference](#)
- [SAS Functions and CALL Routines: Reference](#)
- [SAS DATA Step Statements: Reference](#)
- [Base SAS Procedures Guide](#)
- [The Little SAS Book: A Primer, Fifth Edition](#)

Overview of Universal Printing

<i>What Is Universal Printing?</i>	1
<i>Universal Printing Output Formats</i>	2
<i>Universal Printing and ODS</i>	2
<i>Color Support for Universal Printers</i>	3
Universal Printers and the Color Spaces They Support	3
CMYK Colors	4
RGB and RGBA Colors	6
Example: Static and Varying Background Color in a Table Using RGBA Colors	7

What Is Universal Printing?

Universal Printing is a system that enables you to create a variety of document and graphic output formats programmatically. For example, you can use Universal Printing to create HTML5 files or PDF and RTF documents.

Universal Printing enables you to define printers and to set options to control the printed output. In addition to creating the various document and graphic output types, you can send output to a printer.

SAS routes all printing through Universal Printing services. All Universal Printing features are controlled by system options, thereby enabling you to control many print features, even in batch mode. For more information about these system options, see [“About SAS System Options” on page 80](#).

In SAS Data and AI Studio (known as SAS Studio prior to 2026.06), printing is also available through a browser by selecting the print icon.

Universal Printing Output Formats

In addition to sending print jobs to a printer, you can also direct output to external files that are widely recognized by different types of printers and software programs. You can use Universal Printing to produce a PDF file type in SAS Data and AI Studio. SAS Data and AI Studio also supports the HTML5 and RTF ODS destinations, either of which might use a universal printer.

You set the value of the PRINTERPATH= system option to a Universal Printer or use ODS statements to create output in one of the above formats. When the PRINTERPATH= system option is set to a printer that prints to a file, the default filename is sasprt.*extension*. *extension* is the printer format type. An example is sasprt.pdf. The file is written to the current directory. In SAS Data and AI Studio, you must specify the path and the filename where the output file is to be written.

You can use the PRINTERPATH= system option to change the location and the name of the file. Here is an example:

```
options printerpath=(pdf out);  
filename out '~/myimages/graph1.pdf;
```

Universal Printing and ODS

The following table lists some of the other ODS destinations that use Universal Printing.

Table 1.1 ODS Destinations That Use Universal Printing

Output Format	ODS Statement	Description
User-specified	ODS PRINTER	Uses the printer specified in the PRINTER= option. A Universal Printer can be specified to create an output file in a specific format.
PDF	ODS PDF	Uses the Universal Printing PDF printer to generate a PDF document. By default, images are in the SVG format.
HTML	ODS HTML5	Generates an HTML document. By default, images are in the SVG format. In the HTML code, images can be inline or

Output Format	ODS Statement	Description
		inserted using an <EMBED>, , or tag.
Microsoft Word	ODS WORD	Generates a Word file. By default, images are in the SVG format.

For more information about ODS, see [SAS Output Delivery System: User's Guide](#).

Color Support for Universal Printers

Universal Printers and the Color Spaces They Support

All Universal Printers support 24-bit RGB colors. Most printers support 32-bit CMYK colors or 32-bit RGBA (transparency) colors. The following table lists the Universal Printers and their respective color support.

Table 1.2 Color Support for Universal Printers

Universal Printer	Color Support	Supports Transparency
EMF	RGBA (32-bit)	Yes
SASEMF	RGBA only for bitmap images. RGB for vector elements.	Yes, for bitmap images
GIF	RGBA (24-bit with support for transparent backgrounds)	Yes
PCL5c ¹	RGBA	Yes
PDF	CMYK and RGBA	Yes, for RGBA colors
PNG	RGBA (32-bit)	Yes
PostScript	CMYK and RGB RGBA for GIF images ²	No No

Universal Printer	Color Support	Supports Transparency
SVG	RGBA (32-bit)	Yes
TIFF	RGBA	Yes
TIFFk	CYMK	No

1 PCL4 and PCL5 Universal Printers support only monochrome printing.

2 The PostScript Universal Printer recognizes RGBA colors but transparency is not supported.

For information about CYMK, RGB, and RGBA colors, see [“CMYK Colors” on page 4](#) and [“RGB and RGBA Colors” on page 6](#).

CMYK Colors

CMYK colors setting specify eight hexadecimal characters with a value of 0–255 to specify the amount of cyan, magenta, yellow, and black ink. Use your printer's Pantone Color Lookup table to find the CMYK values for your printer. If you specify an unsupported color, such as a CMYK color with an EMF printer, the color is converted to a color that is supported.

You can specify CMYK colors where ever colors can be set (for example, in the PROC PRINT statement STYLE option or in the TITLE statement).

Preface the hexadecimal number with a CMYK or a K. Here are some examples of CMYK colors that you can set in SAS:

Table 1.3 Example CMYK Colors

Hexadecimal Representation	Color
cmykFF000000	cyan
k00FF0000	magenta
cmyk0000FF00	yellow
kFFFF0000	blue
cmykFF00FF00	green
k00FFFF00	red
cmykFFFFFF00	process black, using cyan, magenta, and yellow

Hexadecimal Representation	Color
k000000FF	black

The first byte of the hexadecimal number represents cyan. The second byte represents magenta. The third byte represents yellow. The fourth byte represents black.

This example uses the STYLE option to set the column background color to magenta and sets the foreground color to white. The TITLE statement sets the output title to blue.

```
options obs=5 nodate;
ods html5 close;
ods pdf;
proc print data=sashelp.demographics label
      style(header)={background=cmyk00ff0000
foreground=k000000000} noobs;
  var name pop;
  label name=Country Name pop=Population;
  title color=kffff0000 'Demographics 2005';
run;
ods pdf close;
ods html5;
```

Figure 1.1 CMYK Color Specified in the STYLE Option

Demographics 2005	
Country Name	Population
BAHAMAS	323,063
BELIZE	269,736
CANADA	32,268,243
COSTA RICA	4,327,228
CUBA	11,269,400

RGB and RGBA Colors

RGB and RGBA colors combines red, green, and blue colors in different ratios to create colors. The A is the alpha channel, which represents a percentage of opacity.

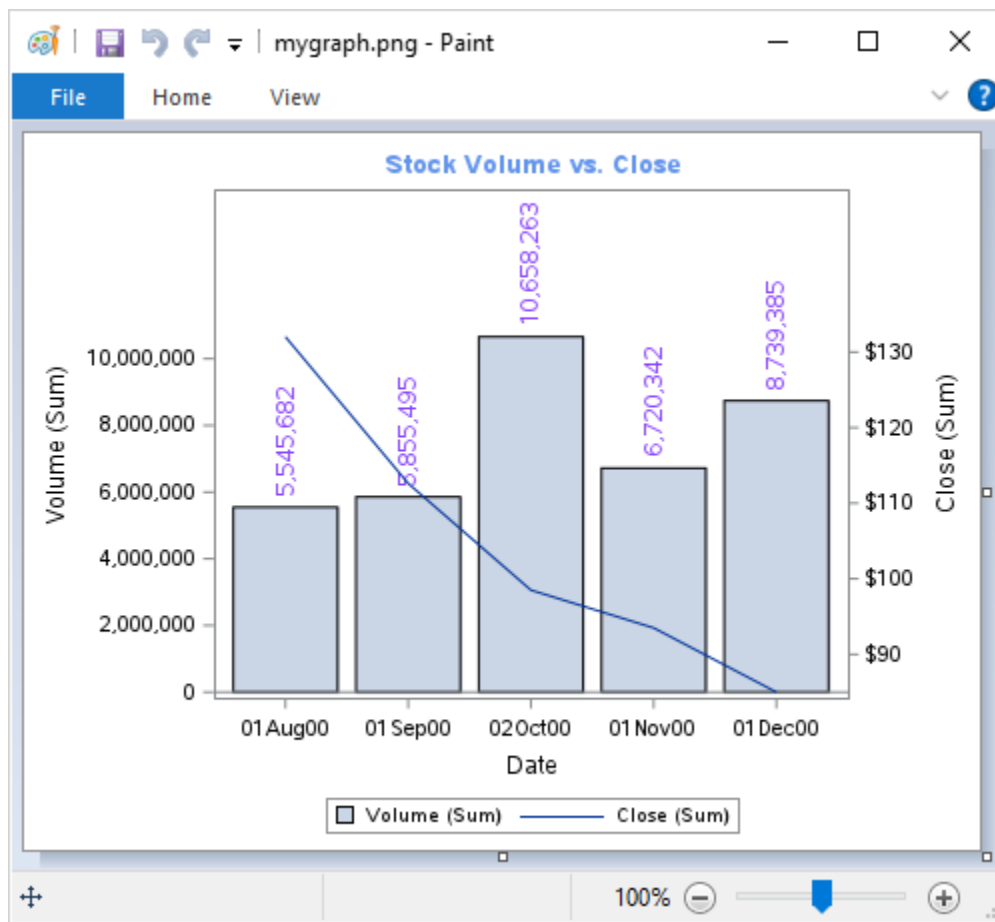
You specify RGB colors as a triple of hexadecimal numbers, ranging from 00–FF. Each hexadecimal number indicates how much of the red, green, or blue is included in the color. RGBA color includes an additional hexadecimal number for the alpha channel that indicates how transparent the color is. FF is opaque and 00 is transparent. In both RGB and RGBA color specifications, the first hexadecimal number is red, the second, is green, and the third is blue. In RGBA colors, the fourth hexadecimal number is the alpha channel specification.

You can specify RGB and RGBA colors where ever colors can be set. For example, set colors as an option in the VBAR statement in the SGPLOT procedure or in the TITLE statement. For RGB colors, preface the hexadecimal number with a CX. For RGBA colors, preface the hexadecimal number with RGBA or A.

The following SGPLOT procedure uses an RGBA color to create the bar labels:

```
ods html5 close;
ods printer printer=png;
proc sgplot data=sashelp.stocks (where=(date >= "01jan2000"d
                                     and date <= "01jan2001"d
                                     and stock = "IBM"));
  title color=a6495edff "Stock Volume vs. Close";
  vbar date / response=volume
        datalabel
        datalabelattrs=(color=a8a44ff8a size=10);
  vline date / response=close y2axis;
run;
title;
ods printer close;
ods html5;
```

Here is the PNG file with bar labels:

Figure 1.2 RGBA Color Specified for the Bar Labels

Example: Static and Varying Background Color in a Table Using RGBA Colors

This example program does the following:

- Creates the format PCT. using a DATA_NULL_ statement. The DATA step defines salary ranges of \$3,000.00 and calculates an RGBA color value for each salary range. The CALL EXECUTE statement is used to display the FORMAT procedure code as it is generated.
- Creates a data set.
- The PRINT procedure uses an RGBA color value for the background of the table header and formats the salary variable using the PCT. format.

```
options nodate;
```

```
/* Create the PCT format. */
/* The color variable is a concatenation of calculated */
/* hexadecimal values. */
```

```

data _null_ ;
  call execute('proc format fmtlib ; value pct');
  max=10000;
  maxloop=255;
  do i=1 to maxloop by 10;
    color='RGBA' || put((maxloop)/(maxloop+i)*200),hex2.)
              || put((maxloop)/(maxloop+i)*235),hex2.)
              || put((maxloop)/(maxloop+i)*255),hex2.) || '95';
    from=max;
    to=(max+3000);
    max=max+3000;
    /* Create salary ranges of $3000.00 equal to the calculated RGBA
color value.*/
    call execute(put(from,best.) || '-' || put(to,best.) || '=' ||
quote(color));
    end;

    /* Create RGBA values for missing values and values outside the
salary ranges. */
    call execute('.="RGBAF7F5F0480" other="RGBAFF2A2A88"; run;');
  run;

data staff;
  infile datalines dlm='#';
  input Name $16. IdNumber $ Salary
        Site $ HireDate date7.;
  format hiredate date7.;
  datalines;
Capalleti, Jimmy# 2355# 21163# BR1# 30JAN09
Chen, Len#      5889# 20976# BR1# 18JUN06
Davis, Brad#    3878# 19571# BR2# 20MAR84
Leung, Brenda#  4409# 34321# BR2# 18SEP94
Martinez, Maria# 3985# 49056# US2# 10JAN93
Orfali, Philip#  0740# 50092# US2# 16FEB03
Patel, Mary#    2398# 35182# BR3# 02FEB90
Smith, Robert#  5162# 40100# BR5# 15APR66
Sorrell, Joseph# 4421# 38760# US1# 19JUN11
Zook, Carla#    7385# 22988# BR3# 18DEC10
;
run;
ods html close;
ods pdf file='outpdf.pdf';
proc print data=staff noobs label
  style(HEADER)={background=rgbac7eafe95 fontstyle=italic}
  style(DATA)={foreground=black};
  var name IdNumber ;
  var salary /style(DATA)={background=pct.};
  label IdNumber='Employee Number' salary='Salary in U.S. Dollars';
  format salary dollar7.;
  title 'Generated Colors for the Variable Salary';
run;
ods pdf close;

```

Here is the result of the FORMAT procedure execution in the first DATA step:

FORMAT NAME: PCT LENGTH: 13 NUMBER OF VALUES: 28		
MIN LENGTH: 1 MAX LENGTH: 40 DEFAULT LENGTH: 13 FUZZ: STD		
START	END	LABEL (VER. V7 V8 15JUN2022:11:29:12)
.	.	RGBAF7F5F0480
10000	13000	RGBAC7EAFE95
13000<	16000	RGBABFE1F495
16000<	19000	RGBAB8D9EB95
19000<	22000	RGBAB2D1E395
22000<	25000	RGBAAACADB95
25000<	28000	RGBAA6C3D495
28000<	31000	RGBAA1BD9D95
31000<	34000	RGBAA9CB7C795
34000<	37000	RGBAA97B2C195
37000<	40000	RGBAA93ADB95
40000<	43000	RGBAA8FA8B695
43000<	46000	RGBAA8BA3B195
46000<	49000	RGBAA879FAC95
49000<	52000	RGBAA849BA895
52000<	55000	RGBAA8097A495
55000<	58000	RGBAA7D93A095
58000<	61000	RGBAA7A909C95
61000<	64000	RGBAA778C9895
64000<	67000	RGBAA74899595
67000<	70000	RGBAA72869195
70000<	73000	RGBAA6F838E95
73000<	76000	RGBAA6D808B95
76000<	79000	RGBAA6B7D8895
79000<	82000	RGBAA687B8595
82000<	85000	RGBAA66788395
85000<	88000	RGBAA64768095
OTHER	**OTHER**	RGBAAFF2A2A88

Here is the formatted PDF output:

Output 1.1 PDF Formatted Using RGBA Color Values**Generated Colors for the Variable Salary**

<i>Name</i>	<i>Employee Number</i>	<i>Salary in U.S. Dollars</i>
Capalleti, Jimmy	2355	\$21,163
Chen, Len#	5889	\$20,976
Davis, Brad#	3878	\$19,571
Leung, Brenda#	4409	\$34,321
Martinez, Maria#	3985	\$49,056
Orfali, Philip#	0740	\$50,092
Patel, Mary#	2398	\$35,182
Smith, Robert#	5162	\$40,100
Sorrell, Joseph#	4421	\$38,760
Zook, Carla#	7385	\$22,988

Setting Up and Managing Universal Printing

<i>Setting Up the Universal Printing Interface and the Default Printing Environment</i>	11
Universal Printing in UNIX	11
Return to the Default Printer	12
<i>Viewing Universal Printer Settings</i>	12
<i>Modifying Universal Printing Printer Settings</i>	13
<i>Managing Universal Printers Using SAS Procedures</i>	14
Using the PRTDEF Procedure	14
Using the FONTREG Procedure	14
Using the OPTIONS Procedure	14
Using the QDEVICE Procedure	14
Using the REGISTRY Procedure	15

Setting Up the Universal Printing Interface and the Default Printing Environment

Universal Printing in UNIX

Universal Printing is enabled when SAS starts in the Linux operating environment. No further action is required.

Return to the Default Printer

When you use the `PRINTERPATH=` system option to specify a printer, the print job is controlled by Universal Printing. In SAS Data and AI Studio, the `PRINTERPATH=` system option value is PDF.

```
options printerpath="";
```

Viewing Universal Printer Settings

You can use the `QDEVICE` procedure to view the settings of a Universal Printer. To view printer settings using the `QDEVICE` procedure, submit this code:

```
proc qdevice;  
  printer printer-name;  
run;
```

Here are the printer settings for the GIF printer:

```

19  proc qdevice;
20  printer gif;
21  run;

      Name: GIF
      Description: Graphics Interchange Format RGB Color/Alpha Blending
      Module: SASPDGIF
      Type: Universal Printer
      Registry: SASHELP
      Prototype: GIF
      Default Typeface: Cumberland AMT
      Typeface Alias: Courier
      Font Style: Regular
      Font Weight: Normal
      Font Height: 8 points
      Font Version: Version 1.03
      Maximum Colors: 16777216
      Visual Color: True Color
      Color Support: RGBA
      Destination: sasprt.gif
      I/O Type: DISK
      Data Format: GIF
      Height: 6.25 inches
      Width: 8.33 inches
      Ypixels: 600
      Xpixels: 800
      Rows(vpos): 50
      Columns(hpos): 114
      Left Margin: 0 inches
      Minimum Left Margin: 0 inches
      Right Margin: 0 inches
      Minimum Right Margin: 0 inches
      Bottom Margin: 0 inches
      Minimum Bottom Margin: 0 inches
      Top Margin: 0 inches
      Minimum Top Margin: 0 inches
      XxY Resolution: 96x96 pixels per inch
      Compression Enabled: Always
      Compression Method: LZW
      Font Embedding: Never
      Animation: Enabled

```

The QDEVICE procedure does not report all printer settings. For a description of the printer settings that can be reported, see [“QDEVICE Procedure” in Base SAS Procedures Guide](#).

Modifying Universal Printing Printer Settings

You can modify printer settings by setting SAS system options or by using the PRTDEF procedure. See the following topics:

- To modify printer settings using SAS system options, see [“About SAS System Options” on page 80](#).

- To modify printer settings using the PRDEF procedure, see [“Using the PRTDEF Procedure” on page 14](#).

Managing Universal Printers Using SAS Procedures

Using the PRTDEF Procedure

This procedure is used to create and modify printer definitions. It is especially useful if you use SAS in batch mode. A system administrator can use it with the USESASHELP option to create printer definitions in the Sashelp library for all SAS users at your site. Individual users can use it without the USESASHELP option to create and modify personal printer definitions in their Sasuser library. In either case, a SAS data set that contains printer attribute data is required. For details and examples, see [“PRTDEF Procedure” in Base SAS Procedures Guide](#).

Using the FONTREG Procedure

Register fonts in the SAS registry with the FONTREG procedure, to augment those TrueType and Type1 fonts that come installed with SAS. These system fonts can be used in SAS output. For more information, see [“Registering Fonts” on page 26](#).

Using the OPTIONS Procedure

The OPTIONS procedure is a discovery tool that lists the current settings of SAS system options. The SAS log displays the results. For details about the SAS procedure that discovers SAS system option settings, see the [“OPTIONS Procedure” in SAS System Options: Reference](#).

Using the QDEVICE Procedure

This procedure is useful for viewing universal printer definitions for a specific printer, such as PDF. You can also use the QDEVICE procedure to view the list of fonts installed for a device or a printer. Or use it to determine whether the printer

that you are using supports font embedding. For more information, see [“QDEVICE Procedure” in *Base SAS Procedures Guide*](#).

Using the REGISTRY Procedure

The REGISTRY procedure is useful in maintaining the SAS registry. For example, you can import, validate, diagnose, or uninstall registry files. Use this procedure to view the list of available Universal Printers and printer prototypes that SAS provides. For more information, see [“REGISTRY Procedure” in *Base SAS Procedures Guide*](#).

Using Universal Printing

<i>Viewing Universal Printers and Printer Prototypes</i>	17
<i>Specifying the Page Orientation for Universal Printing Documents</i>	18
<i>Embedding Non-Viewable Comments in Universal Printing Output</i>	19

Viewing Universal Printers and Printer Prototypes

SAS provides Universal Printers and printer prototypes that you can use to create your own printers. You can use the QDEVICE procedure to create a data set of printers and then print the printer information using the PRINT procedure.

To create a table of printers and print the list with a description of each printer, submit this code:

```
proc qdevice out=printers;
    printer _all_;
run;

proc print data=printers;
    var name desc;
run;
```

For more information, see [“QDEVICE Procedure”](#) in *Base SAS Procedures Guide*.

To print a list of printer prototypes to the SAS log, submit this SAS program:

```
filename registry temp;
proc printto log=registry;
run;

proc registry list keyonly levels=1 startat="core\printing
\prototypes";
proc printto;
run;
```

```

data prototypes;
  keep prototype;
  infile registry lrecl=300 pad;
  length line $300;
  input line $300.;
  if substr(line,1,1) = "["
  then do;
    prototype = strip(substr(line,2,length(line)-2));
    if index(prototype,'core\printing\prototypes') ne 0
    then delete;
  else
    output;
  end;

run;

proc print label;
  label prototype = "Prototype";
run;

```

For more information, see [“REGISTRY Procedure” in Base SAS Procedures Guide](#).

Specifying the Page Orientation for Universal Printing Documents

You can specify the page orientation for each page of a multiple-page document that is created by a Universal Printer. You can also use page orientation for documents that are created for the ODS PDF destination.

The ORIENTATION= system option has four values: PORTRAIT, LANDSCAPE, REVERSEPORTRAIT, and REVERSELANDSCAPE. To change the orientation of a document page, specify the OPTIONS statement, using the ORIENTATION= system option, between the steps that create output to change the page orientation.

```

options nodate nonumber;
ods printer printer=svgview file='orientation.svg' style=Ocean;
title 'Demonstration of Page Orientation Changes in a Document';
footnote 'PROC SGPLOT in Landscape Orientation';
options orientation=landscape;
proc sgplot data=sashelp.class;
  vbar age;
run;

```

```

options orientation=portrait;
footnote 'PROC PRINT in Portrait Orientation';
proc print data=sashelp.class;
run;

```

```

options orientation=landscape;
footnote 'PROC SGSCATTER in Landscape Orientation';

```

```
proc sgscatter data=sashelp.cars;
matrix mpg_city enginesize horsepower /
      diagonal=(histogram kernel);
run;

options orientation=portrait;
footnote 'PROC MEANS in Portrait Orientation';
proc means data=sashelp.cars n mean;
      var mpg_city enginesize horsepower;
run;

ods printer close;
```

For more information, see [“ORIENTATION= System Option” on page 96](#).

Embedding Non-Viewable Comments in Universal Printing Output

You can embed a comment in Universal Printer output that does not appear in the output when the file is displayed or printed. The comment can be a text string up to 4,000 characters that you specify using the COLOPHON= system option. You might want to use the comment as a digital signature or to identify the image, vector graphic, or PDF file. You can use a text editor or a third-party application to view the text string in the file.

This example adds text to an SVG document using the COLOPHON= option:

```
options printerpath=svg colophon='Colophon text: SVG SGPLOT for
sashelp.class';
ods _all_ close;
ods printer file="output-path/MyGraph.svg";
proc sgplot data=sashelp.class;
      reg x=height y=weight / CLM CLI;
run;
ods printer close;
```

Here is the comment in the SVG document (highlighted):

```
<svg xmlns="http://www.w3.org/2000/svg" xmlns:xlink="http://www.w3.org/1999/xlink"
  xml:space="preserve" baseProfile="full" version="1.1" id="SVGMain"
  onload='SVGMain_Init("SVGMain")' viewBox="-1 -1 801 601">
<desc></desc>
<!-- Colophon text: SVG SGPLOT for sashelp.class -->
<defs>
...

```

For more information, see [“COLOPHON= System Option” on page 89](#).

Managing Fonts in Universal Printing

<i>Rendering Fonts</i>	21
<i>The FONTEMBEDDING System Option</i>	22
<i>ODS Styles and TrueType Fonts</i>	23
<i>Portability of TrueType Fonts</i>	23
<i>International Character Support</i>	24
<i>TrueType Fonts Supplied by SAS</i>	24
<i>Registering Fonts</i>	26
<i>Listing the Registered Fonts for a Device or Printer</i>	27
<i>Specifying Fonts</i>	29
Specifying Fonts with ODS Styles	29
Specifying Fonts with SAS Program Statements	30
Specifying Fonts that Support International Characters	30

Rendering Fonts

Universal printing uses the following two methods to generate and display fonts in SAS output.

- the FreeType library
- the font-rendering capabilities of the host

Universal printing supports the following font formats:

- TrueType fonts
- Type1 fonts

Note: Universal Printing does not support double-byte Type1 fonts.

Noto Sans TrueType fonts are included to support languages for harmonious web display.

The output methods in the following table are recommended because they use the FreeType library to render fonts. This means that they can render fonts in all of the operating environments that SAS supports. ¹

Table 4.1 *Devices That Use the FreeType Library to Render Fonts*

Output Method	Device
ODS printing and Universal Printing	PDF, PDF/A
ODS RTF	PNG, SASEMF ¹ EMF
ODS HTML	PNG, PNGT, PNG300, GIF, JPEG, SVG, SVG ^T

¹ If the NOFONTRENDERING option is set, the device driver uses only the FreeType library for measuring the text.

Linux Specifics: With devices that use host rendering in a Linux operating environment, the TrueType fonts must be installed on the X server that is being used. This is usually specified by the DISPLAY environment variable. For more information, see the [Configuration Guide for SAS 9.4 Foundation for UNIX Environments](#).

Run the OPTIONS procedure with the FONTSLOC option specified to determine the location of font files at your site:

```
proc options option=fontsloc value;
run;
```

You can also contact your system administrator to discover the font files location. To get a list of the fonts installed, use the QDEVICE procedure. See [“Listing the Registered Fonts for a Device or Printer” on page 27](#).

The FONTEMBEDDING System Option

Use a system option to set up font embedding. It allows the fonts used in the creation of output to accompany that output, ensuring that it is displayed or printed exactly as you intended. See the [“FONTEMBEDDING System Option” on](#)

1. The FreeType library is used to perform two distinct operations in SAS: measuring the text and rendering the font. Depending on the output devices specified, the FreeType library can perform one or both of these operations to render fonts.

[page 93](#) for more information. Here are some important points to know about font embedding:

- Fonts are included in the output files that are created by the Universal Printer, and, if installed, SAS/GRAPH.
- Output files with embedded fonts do not rely on fonts being installed on the computer that is used to view or print the output file.
- When NOFONTEMBEDDING is set, the output files rely on the fonts being installed on the computer that is used to view or print the font.
- File size is increased for vector output for printers such as PDF.
- Not all printers support font embedding. To determine whether the printer that you are using supports font embedding, use the [QDEVICE procedure](#). If Font Embedding is listed in the SAS log with a value of `Option` or `Always`, then the printer supports font embedding.

```
proc qdevice report=general;
    printer pdf;
run;
```

- Not all browsers support font embedding.

ODS Styles and TrueType Fonts

By default, many SAS/GRAPH device drivers and all Universal Printers generate output by using ODS styles, and these ODS styles use TrueType fonts. If no style is specified, the default style is used.

Note: A licensed installation of SAS/GRAPH is optional.

Portability of TrueType Fonts

TrueType fonts are portable across operating environments and are always available in Microsoft Windows environments. A few TrueType fonts are included with some versions of Linux X Windows.

International Character Support

TrueType fonts support a wide range of international characters. These fonts support Unicode characters. For examples, see [“Specifying Fonts that Support International Characters” on page 30.](#)

TrueType Fonts Supplied by SAS

When you install SAS, a number of TrueType fonts are available. TrueType fonts that are supplied by SAS can be categorized into four groups: Windows Glyph List (WGL) Pan-European character set fonts; graphic symbol; multilingual; and monolingual Asian. The following tables show the fonts supplied by SAS.

Note: The fonts that are supplied by SAS are automatically registered in the SAS registry when you install SAS. To register other TrueType Fonts, see [“Registering Fonts” on page 26.](#)

Table 4.2 *Multilingual TrueType Fonts*

Font Name	Language Supported	Font Description
Arial Unicode MS ¹	Arabic, Armenian, Basic Latin, Bengali, Bopomofo, Cyrillic, Devanagari, Georgian, Greek and Coptic, Gujarati, Gurmukhi, hangul jamo, Hebrew, hiragana, Kanbun, Kannada, katakana, Lao, Malayalam, Oriya, Tamil, Telugu, Oriya, Tamil, Telugu, Thai Tibetan.	sans-serif
Times New Roman Uni ¹	Arabic, Basic Latin, Bopomofo, Cyrillic, Devanagari, Georgian, Greek and Coptic, Gujarati, hangul jamo,	serif

Font Name	Language Supported	Font Description
	Hebrew, hiragana, Kanbun, katakana, Lao, Mongolian, Tamil, Telugu, Thai, Tibetan.	

¹ The Arial Unicode MS and Times New Roman fonts replace the Monotype Sans WT and Thorndale Duospace WT fonts, respectively.

Table 4.3 *Monolingual Asian TrueType Fonts*

Language Supported	Font Name	Character Set
Korean	Gulim, Dotum	KSC5601
	Batang, Gungsuh	KSC5601
Japanese	MS Gothic, MS UI Gothic, MS PGothic	Shift JIS
	MS Mincho, MS PMincho	Shift JIS
Simplified Chinese	CSongGB18030C-Light	GB18030 and
	CSongGB18030-LightHWL	GB2312
	MYingHei_18030_C-Medium	
	MYingHei_18030_C-MediumHWL	
Traditional Chinese ¹	Hei	Big5
	MingLiU, PMingLiU	Big5

¹ Hei, MingLiU, and PMingLiU support HKSCS2004 (Hong Kong Supplemental Character Set) characters.

Table 4.4 *Anova UI and HelveticaNeueforSAS TrueType Fonts Distributed by SAS*

Font Name
Anova UI Italic ¹
Anova UI Roman ¹
Anova UI Bold Italic ¹
Anova UI Bold Roman ¹
Anova UI Light Italic ¹

Font Name
Anova UI Light Roman ¹
HelveticaNeueforSAS
HelveticaNeueforSAS Bold Italic
HelveticaNeueforSAS Bold
HelveticaNeueforSAS Italic
HelveticaNeueforSAS Light
HelveticaNeueforSAS Light Italic

¹ The Anova UI fonts are available starting with 2025.06.

Registering Fonts

In addition to the TrueType fonts that come installed with SAS, SAS supports PostScript Type1 fonts. The following table shows the font prefix and file extension for TrueType and Type1 fonts:

Table 4.5 *Font Types Supported by SAS*

Type	Tag	File Extension
TrueType	<ttf>	.ttf
Type1	<at1>	.pfb ¹

¹ Data from a .pfm file is used to generate output using the WMF device and the EMF universal printer. This file is not required to register Type1 fonts using PROC FONTREG. If you do not register a .pfm file, you might not have the desired results.

The fonts that are supplied by SAS are automatically registered in the SAS registry when you install SAS. Custom fonts such as fonts that are installed after you install SAS or system fonts that are not available in SAS must be registered manually in the SAS registry by using the FONTREG procedure.

Note: Registering a font in the SAS registry means that SAS knows where to find the font file. The font file is not actually used until the font is called for in a SAS program.

The FONTREG procedure enables you to update the SAS registry to include custom fonts, which can then be used in SAS output. PROC FONTREG uses FreeType font-rendering to recognize and incorporate various types of font definitions. Fonts of any type that can be incorporated and used by SAS are known collectively in this documentation as fonts in the FreeType library.

To register custom fonts in SAS, submit the following SAS code.

```
proc fontreg;
    fontpath 'pathname';
run;
```

The font files must be stored in persistent storage that is accessible to the SAS Computer server. The FONTPATH statement specifies the location of the font files, and *pathname* is the directory path to the files.

Note: Do not move or delete the font files after you have registered your fonts in the SAS registry.

In SAS Data and AI Studio, when you run the FONTREG procedure in a Program Editor window, the registered fonts are made available for the duration of the session. When you sign out of SAS Data and AI Studio, your custom font registration is lost. You must register your custom fonts each time you start a new SAS Data and AI Studio session. You can add the FONTREG procedure step code to your SAS Data and AI Studio autoexec file to register your fonts automatically each time you start SAS Data and AI Studio. For more information, see [SAS Data and AI Studio: User's Guide](#).

Listing the Registered Fonts for a Device or Printer

You can use the QDEVICE procedure to view the list of fonts that have been registered in the SAS registry, including fonts that you registered with the FONTREG procedure. The following code creates macro %FONTLIST, which you can use to list the available fonts for a SAS/GRAPH device or a Universal Printer.

```
%macro fontlist(type, name);
    proc qdevice report=font out=fonts;
        &type "&name";
        var font ftype fstyle fweight;
    run;

    data fontdata;
        set fonts;
        drop ftype;
        length type $16;
        nullChar = index(font, '00'x);
        if (nullChar) then do;
```

```

        font = substr(font,1,nullChar);
    end;
    if ftype = "System"
    then do;
        if substr(font,2,3) = "ttf" then type = "TrueType";
        else if substr(font,2,3) = "at1" then type = "Adobe Type1";
        else if substr(font,2,3) = "cff" then type = "Adobe CFF/
Type2";
        else if substr(font,2,3) = "pfr" then type = "Bitstream
PFR";
        else type = "System";
        if type ^= "System" then font = substr(font,7,length(font)-6);
        else if substr(font,1,1) = "@"
        then font = substr(font, 2,length(font)-1);
        end;
    else type = "Printer Resident";
    drop nullChar;
run;

proc sort;
    by font;
run;

title "Fonts Supported by the %upcase(&name)
%sysfunc(propcase(&type))";
proc print label;
    label fstyle="Style" fweight="Weight" font="Font" type="Type";
run;
%mend fontlist;

```

Macro argument *type* can be DEVICE for a SAS/GRAPH device or PRINTER for a Universal Printer. *Name* specifies the name of the device or printer. The following statement lists the fonts that are available with the SVG Universal Printer:

```
%fontlist(printer, svg)
```

Here is an example of the output:

Fonts Supported by the SVG Printer

Obs	Font	Style	Weight	Type
1	Albany AMT	Italic	Bold	TrueType
2	Albany AMT	Italic	Normal	TrueType
3	Albany AMT	Roman	Bold	TrueType
4	Albany AMT	Roman	Normal	TrueType
5	Anova	Italic	Normal	TrueType
6	Anova	Roman	Normal	TrueType
7	Anova Bold	Italic	Normal	TrueType
8	Anova Bold	Roman	Normal	TrueType
9	Anova Light	Italic	Normal	TrueType
10	Anova Light	Roman	Normal	TrueType
11	Anova UI	Italic	Normal	TrueType
12	Anova UI	Roman	Normal	TrueType
13	Anova UI Bold	Italic	Normal	TrueType
14	Anova UI Bold	Roman	Normal	TrueType
15	Anova UI Light	Italic	Normal	TrueType
16	Anova UI Light	Roman	Normal	TrueType
17	Arial Symbol	Italic	Bold	TrueType
18	Arial Symbol	Italic	Normal	TrueType
19	Arial Symbol	Roman	Bold	TrueType
20	Arial Symbol	Roman	Normal	TrueType

For more information, see “QDEVICE Procedure” in *Base SAS Procedures Guide*.

Specifying Fonts

Specifying Fonts with ODS Styles

Each ODS style defines attributes of ODS output according to a specific theme. The attributes include fonts, colors, and so on. You can use the STYLE= option in your ODS statement to select a different ODS style for your output. To see examples of the ODS styles that are available, see “ODS Styles Gallery” in *SAS Output Delivery System: Procedures Guide*.

You can use the TEMPLATE procedure create your own custom ODS style that defines the fonts that you want to use in your output. For information, see “TEMPLATE Procedure: Creating a Style Template” in *SAS Output Delivery System: Procedures Guide*.

Specifying Fonts with SAS Program Statements

Many statements in SAS provide options that enable you to specify a font. For example, if you want to use the TrueType font Albany AMT for your title, include the FONT= option in the TITLE statement as shown in the following example:

```
title1 font="Albany AMT" "Text in Albany AMT";
```

You can also specify attributes such as style or weight in the TITLE statement by using the forward slash (/) as a delimiter.

```
title1 font="Albany AMT/Italic/Bold" "Text in Bold Italic Albany AMT";
```

In ODS style templates, the text attributes are specified after the text size parameter.

Note: You should use the <ttf> tag only when it is necessary (for example, to distinguish between a TrueType font and another type of font with the same name).

Specifying Fonts that Support International Characters

To print international characters, specify a font that supports Unicode characters and use the Unicode value for each character. TrueType fonts support a wide range of international characters. Here is an example that includes the Euro currency symbol in a title:

```
title1 font="Arial Unicode MS" "Total Sales in " '20AC'x;
```

Here is an example uses the PRINT statement to print data that contains world currency symbols:

```
%macro addobs(name, char);                                /* 1 */
  name="&name";
  code=upcase("&char");
  char=kcvrt("&char"x, 'ucs2b', 'utf8');
  output;
%mend addobs;

data utf8char;                                           /* 2 */
  length name $40;
  label name="Name" code="Code" char="Character";
  %addobs(Cent Sign, 00A2);
  %addobs(Pound Sign, 00A3);
  %addobs(Currency Sign, 00A4);
  %addobs(Yen Sign, 00A5);
  %addobs(Rupee Sign, 20A8);
  %addobs(Euro Sign, 20AC);
```

```

%addobs(Dong Sign, 20AB);
%addobs(Euro-currency Sign, 20A0);
%addobs(Colon Sign, 20A1);
%addobs(Cruzeiro Sign, 20A2);
%addobs(French Franc Sign, 20A3);
%addobs(Lira Sign, 20A4);
run;
title "Currency Symbols";
proc print data=utf8char label /* 2 */
    style(data)={fontsize=16pt fontfamily='Arial Unicode MS'};
    id name code;
run;

```

- 1 Macro %ADDOBS generates an observation for a Unicode character code. The KCVT function converts the Unicode code to a UTF-8 character, which is stored in variable Char. See “KCVT Function” in *SAS National Language Support (NLS): Reference Guide*.
- 2 DATA step UTF8CHAR uses macro %ADDOBS to create a data set that contains several international characters.
- 3 The PRINT procedure prints the data. Option STYLE specifies font Arial Unicode MS, which supports Unicode characters.

Here is the output.

Currency Symbols		
Name	Code	Character
Cent Sign	00A2	¢
Pound Sign	00A3	£
Currency Sign	00A4	¤
Yen Sign	00A5	¥
Rupee Sign	20A8	₹
Euro Sign	20AC	€
Dong Sign	20AB	₫
Euro-currency Sign	20A0	ؑ
Colon Sign	20A1	ؑ
Cruzeiro Sign	20A2	ؑ
French Franc Sign	20A3	₣
Lira Sign	20A4	₺

Creating Files Using Universal Printing

<i>Creating PDF Files Using Universal Printing</i>	34
PDF Files in SAS	34
Creating a PDF File	34
Example of Creating a PDF Using the ODS PDF Statement	35
System Options That Affect PDF Output	36
<i>Creating SVG (Scalable Vector Graphics) Files Using Universal Printing</i>	37
Overview of Scalable Vector Graphics in SAS	37
Web Server Content Type for SVG Documents	39
The SVG Universal Printers and the Output That They Create	39
How to Create SVG Files	40
Browser Support for Viewing SVG Documents	43
Images in SVG Documents	44
Setting the Environment to Create Stand-alone SVG Documents	46
Creating Stand-alone SVG Documents Using the ODS PRINTER Destination	53
Inserting SVG Images into HTML Documents	62
<i>Creating TIFF Images Using Universal Printing</i>	69
TIFF Images in SAS	69
The TIFF Universal Printers	70
Creating a TIFF Image	70
Example of Creating a TIFF Image Using the ODS PRINTER Statement	71
<i>Creating Animated GIF Images and SVG Documents</i>	72
About Animated GIF Images and SVG Documents	72
Animation System Options	74
Example: Creating an Animated SVG Image	75

Creating PDF Files Using Universal Printing

PDF Files in SAS

PDF files can be read by the Adobe Acrobat Reader and other applications. In SAS, you create PDF files using the Output Delivery System (ODS). ODS uses the PDF Universal Printing printer to create a PDF. ODS provides styles and templates that you can apply to a document, or you can create your own styles and templates to customize a document. For more information, see [“ODS PDF Statement” in SAS Output Delivery System: User’s Guide](#).

For a description of the PDF printer, you can either view the printer in the SAS registry or submit the following QDEVICE procedure and view the output in the SAS log:

```
proc qdevice;  
  printer pdf;  
run;
```

Note: If you have SAS/GRAPH installed, your PDF output can contain links and pop-up text boxes. For more information, see [“Enhancing Web Presentations with Chart Descriptions, Data Tips, and Drill-Down Functionality” in SAS/GRAPH: Reference](#).

Creating a PDF File

You can create a PDF file using the ODS PDF or ODS PRINTER statements. You specify the PDF Universal Printer either as the value of the PRINTERPATH= system option or as the value of the PRINTER= option in the ODS PRINTER statement. The ODS PDF statement creates output using the PDF Universal Printer. Therefore, you do not need to explicitly specify the PDF Universal Printer when you use the ODS PDF statement.

Here is some sample code to create a PDF file. In the first sample, the PDF Universal Printer does not need to be specified because the ODS PDF statement uses the PDF Universal Printer to create a PDF. In the second sample, the PDF Universal Printer is specified as the value of the PRINTERPATH= system option and the ODS PRINTER statement creates the PDF:

```

■ ods html close;
  ods pdf;

  ...more SAS code...

ods pdf close;
ods html;

■ options printerpath=pdf;
  ods html close;
  ods printer;

  ...more SAS code...

ods printer close;
ods html;

```

Example of Creating a PDF Using the ODS PDF Statement

This example creates a PDF file that contains the first five observations of the data set `Sashelp.Class`:

```

options obs=5 nodate pageno=1;
ods html close;
ods pdf;

proc print data=sashelp.class;
run;

ods pdf close;
ods html;

```

Here is the PDF output:

Figure 5.1 *Sashelp.Class in a PDF File*

Obs	Name	Sex	Age	Height	Weight
1	Alfred	M	14	69.0	112.5
2	Alice	F	13	56.5	84.0
3	Barbara	F	13	65.3	98.0
4	Carol	F	14	62.8	102.5
5	Henry	M	14	63.5	102.5

System Options That Affect PDF Output

Before you create PDF output, you can use SAS system options to set document security restrictions. The document security restrictions specify what can be done to the document, as well as the security method, the printing resolution, and the encryption level.

The following table lists the system options that can be used to set the PDF document security restrictions:

System Option	Description
PDFACCESS on page 100	Specifies whether the PDF document can be edited.
PDFASSEMBLY on page 102	Specifies whether PDF documents can be assembled.
PDFCOMMENT on page 103	Specifies whether PDF document comments can be modified.
PDFCONTENT on page 105	Specifies whether the contents of a PDF document can be changed.
PDFCOPY on page 106	Specifies whether text and graphics from a PDF document can be copied.
PDFFILLIN on page 107	Specifies whether PDF forms can be filled in.
PDFPAGELAYOUT on page 109	Specifies the page layout for PDF documents.
PDFPAGEVIEW on page 110	Specifies the page viewing mode for PDF documents.
PDFPASSWORD on page 111	Specifies the password to use to open a PDF document and the password used by a PDF document owner.
PDFPRINT on page 113	Specifies the resolution to print PDF documents.
PDFSECURITY on page 115	Specifies the level of encryption for PDF documents.

Creating SVG (Scalable Vector Graphics) Files Using Universal Printing

Overview of Scalable Vector Graphics in SAS

Scalable Vector Graphics in SAS

Scalable Vector Graphics (SVG) is an XML language for describing two-dimensional vector graphics. SAS creates SVG documents based on the W3C recommendation for SVG documents. SAS SVG documents are created using the UNICODE standard encoding UTF-8.

SAS can create SVG documents by using Universal Printers and SAS/GRAPH device drivers. Because SAS/GRAPH SVG device drivers use the SVG Universal Printers, this section contains some information about creating SVG documents using SAS/GRAPH.

Most often in SAS, the SVG Universal Printers and device drivers are used to create graphs. Graphs can be created by using ODS Graphics or SAS/GRAPH. You can also use the SVG Universal Printers to show tables or reports that you create as SVG documents.

Several ODS destinations (EPUB, HTML, HTML5, LISTING, and PRINTER destinations) can be used to create SVG documents. SVG is the default Universal Printer and device driver for the ODS HTML5 destination.

SVG documents can be stand-alone files or integrated within an HTML5 or EPUB file. A stand-alone SVG document can be referenced as a link target, referenced as an embedded file in an HTML document, or referenced as a CSS2 or XSL property. For information about embedding SVG documents in web pages, see the topic on using SVG documents in web pages in the SVG 1.1 specification on the W3 SVG website <http://www.w3.org/TR/SVG>.

Multi-page SVG documents can be animated in Base SAS and SAS/GRAPH. When you create animated SVG documents in Base SAS using Universal Printing without specifying any ODS Graphics procedures, the animated SVG documents appear as a slide show or an animated PowerPoint presentation. For more information, see [“Creating Animated GIF Images and SVG Documents” on page 72](#).

If you have SAS/GRAPH installed, your SVG documents can contain links and pop-up text boxes.

The information provided here is limited to creating SVG documents using Universal Printers in Base SAS and ODS Graphics. For more information about

creating SVG files in SAS/GRAPH, see “[Enhancing Web Presentations with Chart Descriptions, Data Tips, and Drill-Down Functionality](#)” in *SAS/GRAPH: Reference*.

For detailed information about the SVG standard, see the W3 documentation at <http://www.w3.org/TR/SVG>.

SVG Terminology

SVG canvas

the space upon which the SVG document is rendered.

viewBox

specifies the coordinate system and the area of the SVG document that is visible in the viewport.

viewport

a finite rectangular space within the SVG canvas where an SVG document is rendered. In SAS, the viewport is determined by the value of the PAPER SIZE= system option for a scalable viewport and by the SVGWIDTH= and SVGHEIGHT= system options for a static viewport.

viewport coordinate system or viewport space

the starting X and Y coordinates and the width and height values of the viewport.

user coordinate system or user space

the starting X and Y coordinates and the width and height values of the area of the document to display in the viewport.

user units

is equal to one unit of measurement that is defined in your environment's coordinate system. In many cases, the coordinate system uses pixels. Check with your system administrator to determine the unit of measure that is used in your environment.

Why Create SVG Documents?

SVG documents are displayed clearly at any size in any viewer or browser that supports SVG. SVG documents are ideal for producing documents to display on a computer monitor, PDA, or cell phone; or documents to be printed. Because it is a vector graphic, a single SVG document can be transformed to any screen resolution without compromising the clarity of the document. In comparison, a multiple raster graphic image might require using different screen resolutions in order to display the image at various screen resolutions and sizes.

An SVG document might also be smaller in file size than the same image created by a raster graphic Universal Printer, such as GIF or PNG.

Web Server Content Type for SVG Documents

If the mime content type setting for your web server does not have the correct setting for SVG documents, your web browser might render SVG documents as text files, or SVG documents might be unreadable.

To ensure that SVG documents are rendered correctly, configure your web server to use this mime content type:

```
image/svg+xml
```

The SVG Universal Printers and the Output That They Create

The following table lists the SAS SVG Universal Printers.

Table 5.1 *SVG Universal Printers*

Printer Name	Description
SVG ¹	produces SVG 1.1 documents.
SVGt ¹	produces SVG 1.1 documents that are transparent (no background).
SVGnotip	produces SVG 1.1 documents without tooltips.
SVGZ ¹	produces compressed SVG 1.1 documents.
SVGView ¹	produces SVG1.1 documents with navigational controls when the SVG file contains multiple pages.

¹ When you use this printer in SAS/GRAPH, you can create pop-up data tips. For more information, see ["Data Tips for Web Presentations" in SAS/GRAPH: Reference](#).

SVG prototypes for creating printers are available in the SAS Registry under CORE \PRINTING\PROTOTYPES. You can define your own SVG printer using the PRTDEF procedure. For more information, see ["PRTDEF Procedure" in Base SAS Procedures Guide](#) and ["Managing Universal Printers Using SAS Procedures" on page 14](#).

For a description of an SVG printer, you can either view the printer in the SAS registry or submit the following QDEVICE procedure and view the output in the SAS log:

```
proc qdevice;
  printer svg-printer-name;
run;
```

See Also

- [“Color Support for Universal Printers” on page 3](#)
- [“Creating Animated GIF Images and SVG Documents” on page 72](#)

How to Create SVG Files

Fundamentals of Creating SVG Files Using Universal Printers

To create SVG files, specify an SVG Universal Printer as the value of the PRINTERPATH= system option. Also, specify an ODS destination, such as the ODS PRINTER statement, as shown below.

```
options printerpath=svg;
ods _all_ close;
ods printer;
```

...more SAS code...

```
ods printer close;
ods html;
```

Alternatively, you can specify the SVG printer in the ODS PRINTER statement and eliminate the OPTIONS statement, as shown below.

```
ods _all_ close;
ods printer printer=svg;
```

...more SAS code...

```
ods printer close;
ods html;
```

You can create SVG graphs for ODS Graphics using these statements:

```
ods _all_ close;
ods html5 options (svg_mode="inline");
ods graphics /imagefmt=svg;
```

```

...more SAS code...

ods html5 close;
ods html;
■ ods _all_ close;
  options printerpath=svg;
ods html;

...more SAS code...

ods html close;
ods html;
■ ods _all_ close;
  ods listing;
ods graphics /imagefmt=svg;

...more SAS code...

ods listing close;
ods html;

```

To create SVG graphs using SAS/GRAPH, you can use the ODS LISTING statement:

```

ods _all_ close;
ods listing;
goptions dev=SVG;

...more SAS code...

ods listing close;
ods html;

```

SAS has several system options that enable you to modify various aspects of your SVG document. Here are some SVG document traits:

- a specific SVG Universal Printer
- the height and width of the SVG document
- the size of the viewBox
- whether a multi-page SVG document contains navigational controls

By using the NEWFILE option in the ODS PRINTER statement, you can create an SVG document for the output from each procedure or DATA step.

For more information, see the following language elements:

- [“PRINTERPATH= System Option” on page 117](#)
- [ODS PRINTER Statement](#) in *SAS Output Delivery System: User's Guide*
- [System Options for SVG Documents on page 46](#)

A Summary of ODS Destinations that Create Documents with SVG Images

The following table lists ODS destinations that you can use to create documents that include SVG images.

Destination ¹	Type of SVG Document	Output Created
ODS EPUB	graphs created by ODS Graphics and SAS/GRAPH	an EPUB file with integrated SVG images
ODS EXCEL	graphs created by ODS Graphics and SAS/GRAPH	an Excel file with integrated SVG images
ODS HTML5 svg_mode='embed'	graphs created by ODS Graphics and SAS/GRAPH	an SVG file for each graph and an HTML file that inserts each graph with an <EMBED> tag
ODS HTML5 svg_mode='img'	graphs created by ODS Graphics and SAS/GRAPH	an SVG file for each graph and an HTML file that inserts each graph with an tag.
ODS HTML5 svg_mode='inline'	graphs created by ODS Graphics and SAS/GRAPH	an HTML file with inline SVG images
ODS HTML5 svg_mode='link'	graphs created by ODS Graphics and SAS/GRAPH	an SVG file for each graph and an HTML file that inserts a link to each graph with an tag.
ODS PDF	graphs created by ODS Graphics and SAS/GRAPH	a PDF file with integrated SVG images
ODS POWERPOINT	graphs created by ODS Graphics and SAS/GRAPH	a PowerPoint file with integrated SVG images

¹ SVG is the default image format for all of the destinations listed.

Browser Support for Viewing SVG Documents

Browsers That Support SVG Documents

In order to view SVG documents, you need a viewer or browser that supports Scalable Vector Graphics. Some browsers, such as Mozilla Firefox, have built-in support for SVG documents.

The following table lists some browsers and viewers that support SVG documents.

Table 5.2 SVG Browser Support

Browser or Viewer	Company
Batik SVG Toolkit	Apache Software Foundation
Google Chrome ^{1, 2}	Google Inc.
Microsoft Edge ¹	Microsoft
Mozilla Firefox ¹	Mozilla Foundation
Opera	Opera Software ASA
Safari, including iPad ^{1, 2, 3}	Apple, Inc.

¹ This browser is supported by SAS.

² See ["Sizing and Scaling SVG Images" in SAS/GRAPH: Reference](#).

³ When an animated SVG file is opened in Safari, the **Refresh** button does not work. You need to refresh the browser.

Notes on Using Mozilla Firefox

- Compressed SVG documents using the SVGZ Universal Printer are not supported.
- Zooming and panning features are not currently implemented.
- If you select **View** ⇒ **Page Style** ⇒ **No Style**, all graphs appear as a black rectangle.
- Firefox does not support font embedding. To avoid font mapping problems in your SVG document, you can set the NOFONTEMBEDDING system option. The FONTEMBEDDING option can be set when an SVG document is created. If so, and the SVG document is subsequently viewed in Firefox, Firefox uses the

default font setting. This setting is defined on the **Contents** tab in the Firefox Options window.

Notes on SVG Documents in HTML5 Output

In HTML5 output, SVG document height and width attributes are set to the size of the SVG because many browsers do not scale an SVG file to 100% of the container. The Google Chrome and Safari browsers do scale the SVG file to 100% of the container.

To enable scaling of SVG files that can be scaled to the size of the container, you set the SVGHEIGHT= and SVGWIDTH= system options to 100%:

```
options svgheight="100%" svgwidth="100%";
```

SVGZ documents that you create for ODS HTML5 output can be viewed only with the Google Chrome or Opera web browsers.

Images in SVG Documents

When your SAS program creates an SVG document that contains images, SAS does the following for each image in the document:

- converts the specified image to a PNG format
- encodes the PNG image using base64 encoding
- embeds the base64 encoded PNG image into the SVG document

In the SVG document, the <image> element has an xlink attribute that begins as follows:

```
xlink:href="data:image/png;base64,image-data..."
```

The base64 encoded image follows after base64,.

By default, the SVG universal printer encodes PNG files and embeds them in the SVG document. In the following example, you see the <image> element at the bottom. The encoded image begins after the /png;base64, element attribute. Here is an example of an <image> element in an SVG file. Due to space limitations, only the first part of the element is shown.

```
<image id="SVGMain_Image1" width="95" height="50" xlink:href="data:
image/png;base64,iVBORw0K..."
```

The encoded image data begins with iVBORw0K. The image is the SAS logo in the following SVG document:

Output 5.1 SVG Document Viewed in a Web Browser

Five Observations of sashelp.class					
Obs	Name	Sex	Age	Height	Weight
1	Alfred	M	14	69.0	112.5
2	Alice	F	13	56.5	84.0
3	Barbara	F	13	65.3	98.0
4	Carol	F	14	62.8	102.5
5	Henry	M	14	63.5	102.5

The following program created the embedded image:

```
proc template;
  define style logo;
    parent = Styles.barrettsblue;
    style Body from Body /
      backgroundimage = 'c:\saslogo.png';
  end;
run;

ods html close;
options nodate nonumber orientation=landscape obs=5;

ods printer style=logo printer=svgview file='logo.svg';

proc print data=sashelp.class;
  title color="black" 'Five Observations of sashelp.class';
run;
ods printer close;
ods html;
```

An alternative to embedding encoded PNG files in an SVG document, the SVG Universal Printer can create separate PNG files and link to them from within the SVG document. Here is an example of an <image> element in an SVG document:

```
<image id="Image3" width="200" height="150" xlink:href="I3svgimg.png">
</image>
```

The SVG Universal Printer creates separate PNG files when the SVG printer in use has set the **Images Embedded** registry setting to 0. Use the REGISTRY procedure to see this setting for an SVG printer:

```
proc registry listhelp startat='core\printing\printers\printer-name
\advanced';
run;
```

Use the REGISTRY procedure to change this setting. See [“REGISTRY Procedure” in Base SAS Procedures Guide](#).

The PNG file name has the form *counterPrinterDestinationFilename.png*. *counter* is an integer that is incremented each time a new image is created.

PrinterDestinationFilename is the output destination file name for the printer. For

example, using the default printer destination file name, sasprt, the first three images would be named l1sasprt.png, l2sasprt.png, and l3sasprt.png.

SAS writes a note in the SAS log with the path to the images.

Setting the Environment to Create Stand-alone SVG Documents

Overview of Setting the Environment to Create Stand-alone SVG Documents

As shown in [“Fundamentals of Creating SVG Files Using Universal Printers” on page 40](#), an SVG Universal Printer must be specified either as the printer value using the PRINTERPATH= system option or the ODS Printer statement. You can set any of the SVG system options when SAS is invoked in a SAS program by using the OPTIONS statement, or by using the SAS System Options window.

SAS SVG documents can be created easily by using default values for SVG system options (except for the PRINTERPATH= system option) that establish the SVG environment. This section explains the SVG system options and how they affect stand-alone SVG documents.

SAS System Options That Affect Stand-alone SVG Documents

You can use the following system options to set the environment for creating SVG documents:

Table 5.3 SAS System Options That Affect SVG Documents

Task	System Option
Specify the name of an SVG printer to create a stand-alone SVG document.	PRINTERPATH= on page 117
Embed a comment in the SVG document	COLOPHON on page 89
Options to set the SVG document size	
Set the paper size to use for Universal Printing.	PAPERSIZE= on page 98

Task	System Option
Set the height of the SVG document. If the SVG document contains embedded SVG documents, the height value affects only the outermost SVG document.	SVGHEIGHT=
Set the width of the SVG document. If the SVG document contains embedded SVG documents, the width value affects only the outermost SVG document.	SVGWIDTH=
Set the x-axis coordinate for the lower left corner of an embedded SVG document.	SVGX=
Set the y-axis coordinate for lower left corner of an embedded SVG document.	SVGY=
Specify the X and Y coordinates, and the width and height that are used to set the viewBox for the outermost SVG document. Specify the coordinates of the area of the document that is displayed in the viewport.	SVGVIEWBOX=
Specify whether to force uniform scaling of an SVG document.	SVGPRESERVEASPECTRATIO=
Options that modify the SVG document appearance	
Set the title that appears in the title bar of the SVG document.	SVGTITLE=
Specify whether to display navigational controls in a multi-page SVG document.	SVGCONTROLBUTTONS
Specify whether to display the magnify tool in SVG documents.	SVGMAGNIFYBUTTON
Options for animating SVG documents	
Start or stop creating an animation file.	ANIMATION on page 83
Set the amount of time that each frame of an animated document is held in view.	ANIMDURATION on page 84
Specifies whether to loop through the animation continuously or to play it one time.	ANIMLOOP on page 85

Task	System Option
Specifies whether to overlay frames in the animation or to play them sequentially.	ANIMOVERLAY on page 86
Specifies whether to immediately start an animation when an SVG document appears in the web browser.	SVGAUTOPLAY
Sets the number of seconds for a frame to fade into view.	SVGFADEIN
Specifies whether a frame in the animation overlaps the previous frame or if each frame is played sequentially when a frame is specified to fade in and out.	SVGFADEMODE
Sets the number of seconds for a frame to fade out of view.	SVGFADEOUT

For more information, see [SAS System Options: Reference](#).

Setting the SVG Universal Printer

You set the SVG Universal Printer by setting the `PRINTERPATH=` system option to one of the SVG Universal Printers. You can set the `PRINTERPATH=` system option at any time. The following `OPTIONS` statement sets the Universal Printer to create compressed SVG documents:

```
options printerpath=svgz;
```

For more information, see the following topics:

- [“The SVG Universal Printers and the Output That They Create”](#) on page 39
- [“PRINTERPATH= System Option”](#) on page 117
- [“Fundamentals of Creating SVG Files Using Universal Printers”](#) on page 40

Scaling an SVG Document to the Viewport

To scale an SVG document to the viewport, you can use the default value of null for the `SVGHEIGHT=` and `SVGWIDTH=` system options. A null value equates to the value of 100%, which scales the SVG document to the size of the viewport. In addition, the value of the `SVGVIEWBOX=` system option must be null.

For more information, see the following system options in [SAS System Options: Reference](#):

- [SVGHEIGHT= System Option](#)
- [SVGWIDTH= System Option](#)
- [SVGVIEWBOX= System Option](#)

Setting the ViewBox

The `viewBox` attribute on the `<svg>` element is a set of four numbers: the starting X coordinate, the starting Y coordinate, the height of the SVG document, and the width of the SVG document. SAS sets the `viewBox` attribute value from the value of the `SVGVIEWBOX=` system option. If that option has no value, SAS uses the value of the `PAPERSIZE=` system option to set the height and the width arguments of the `viewBox` attribute. The starting coordinate values are set to 0.

When the `SVGVIEWBOX=`, `SVGHEIGHT=`, and `SVGWIDTH=` system options have a null value (the default value for each of these system options), the SVG document scales to the size of the viewport. If you specify a value for the `SVGVIEWBOX=` system option, the SVG document is scaled to the dimensions specified in the `SVGVIEWBOX=` option.

If you specify the `SVGHEIGHT=` option and the `SVGWIDTH=` option using percentage units, the SVG document scales to the size of the browser window whenever the browser window changes size. If these options are specified using units other than percentage, such as in, cm, or px, the SVG document is a static size and does not scale to the browser window when the window changes size.

For more information, see the following topics:

- [ODS PRINTER Statement](#) in *SAS Output Delivery System: User's Guide*
- ["SVGVIEWBOX= System Option"](#) in *SAS System Options: Reference*
- ["PAPERSIZE= System Option"](#) on page 98
- ["Creating a Static viewBox"](#) on page 50

Interaction between SAS SVG System Options and the SVG Element Attributes

SAS uses the values of the `SVGHEIGHT=`, `SVGWIDTH=`, `SVGVIEWBOX=`, `SVGPRESERVEASPECTRATIO=`, `SVGX=`, and `SVGY=` system options as values for their respective attributes on the outermost `<svg>` element: `height`, `width`, `viewBox`, and `preserveAspectRatio`. For example, if you specify `SVGWIDTH="400"` and `SVGHEIGHT="300"`, SAS creates the `<svg>` element with the attributes `width="400"` and `height="300"`. The values of the `SVGX=` and `SVGY=` system options are used only on embedded `<svg>` elements for the `x` and `y` attributes.

All of these system options have a null default value. When the `SVGVIEWBOX=` system option is null, SAS determines the `viewBox` size based on the value of the `PAPERSIZE=` system option. Therefore, if you do not specify a value for any of

these system options, the only <svg> attribute that SAS sets is the viewBox attribute using the SAS SVG system options. Other <svg> attributes, such as version and xmlns are set by SAS and not by using system options.

SAS creates the following <svg> element when all of the SAS SVG system options are set to their default values:

```
<svg xmlns="http://www.w3.org/2000/svg"
      xmlns:xlink="http://www.w3.org/1999/xlink"
      xml:space="preserve" baseProfile="full" version="1.1"
      id="SVGMain" onload='SVGMain_Init("SVGMain")'
      viewBox="-1 -1 801 601">
```

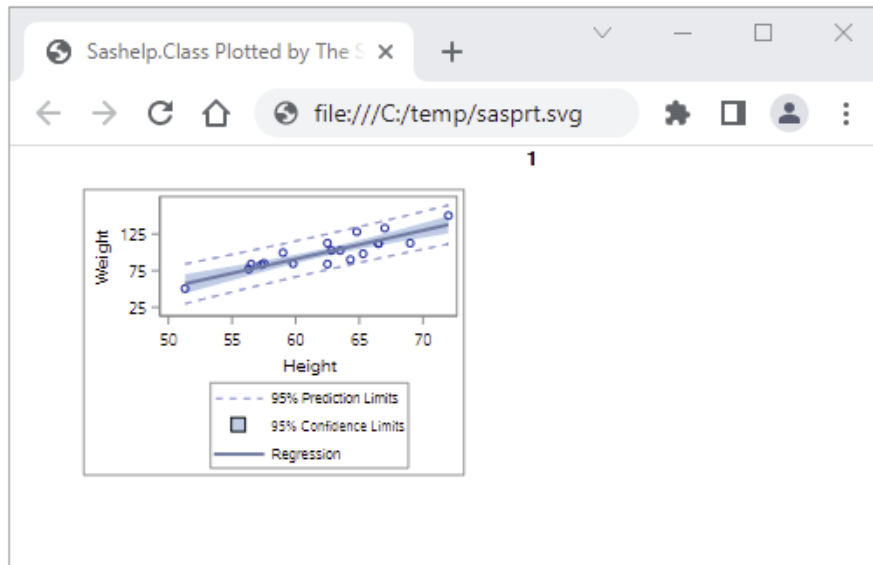
The SVGPRESERVEASPECTRATIO= system option is used to set the preserveAspectRatio attribute in the <svg> element and has an effect only when the viewBox attribute has also been specified in an SVG document.

Negative values can be specified for the values of SVG options. However, if a negative value is specified for the SVGHEIGHT= option or the SVGWIDTH= option, or the height or width arguments in the SVGVIEWBOX= option, the SVG document is not rendered by the browser. It can be useful to specify negative values for the x and y arguments of the SVGVIEWBOX= option to place the origin of the SVG document. A negative argument in the SVGVIEWBOX= option shifts the output to the right. A negative value of the SVGVIEWBOX= option shifts the placement of the document downward.

Creating a Static viewBox

A static viewBox is a viewBox that cannot be changed. When the viewport changes, such as when you resize your browser window, the viewBox remains the same size. To create a static viewBox, you specify the same width and height values for the PAPERSIZE=, SVGWIDTH=, and SVGHEIGHT= system options. The PAPERSIZE= system option sets the viewBox. The SVGWIDTH= and SVGHEIGHT= system options set the size of the SVG document. If the SVGHEIGHT= and SVGWIDTH= options are specified using percentage unit, the SVG document scales to the size of the browser window when the browser window changes size. [Figure 5.2 on page 51](#) shows a static viewBox created by using the following statements:

```
options nodate printerpath=svg papersize=("8cm" "5cm") svgwidth="8cm"
svgheight="5cm"
  svgtitle="Sashelp.Class Plotted by The SGPLOT Procedure";
ods _all_ close;
ods printer
proc sgplot data=sashelp.class;
  reg x=height y=weight / CLM CLI;
run;
ods printerclose;
options papersize=("","") svgwidth="" svgheight="";
ods html;
```

Figure 5.2 A Static Viewbox

To reset the `SVGWIDTH=`, `SVGHEIGHT=`, and `SVGPRESERVEASPECTRATIO=` system options to null, specify two single quotation marks or two double quotation marks with no space between them:

```
options printerpath=svg svgwidth="" svgheight="" svgpreserveaspectratio="";
```

Preserving the Aspect Ratio

When you change the size of the `viewBox`, you can use the `SVGPRESERVEASPECTRATIO=` system option to specify whether you want to preserve the aspect ratio of the SVG document and how to place the SVG document in the viewport. Set this option by using one of the following assignments:

```
SVGPRESERVEASPECTRATIO=align | meetOrSlice | NONE | ""
```

```
SVGPRESERVEASPECTRATIO="align meetOrSlice"
```

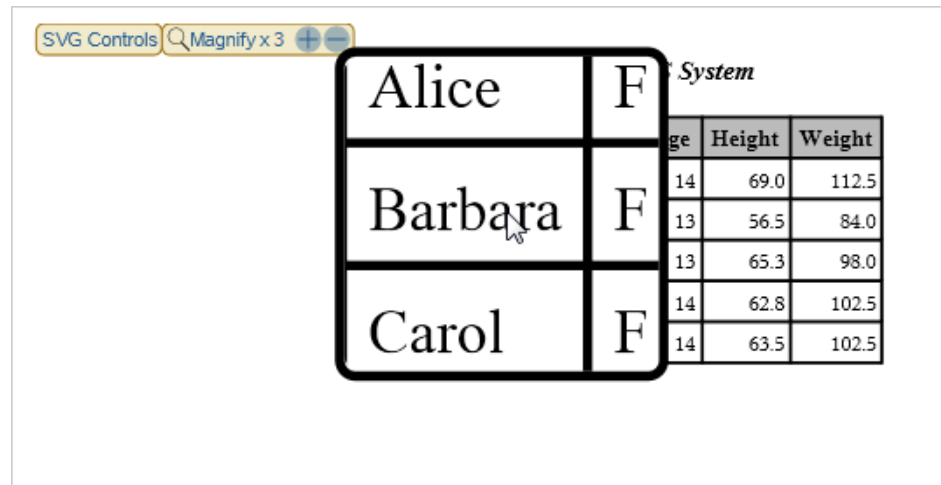
The first argument, *align*, specifies whether to force uniform scaling by specifying the alignment method to use. For example, you can use the `xMidyMid` value to align the midpoint X value of the `viewBox` to the midpoint X value of the viewport, which centers the document horizontally.

The second argument, *meetOrSlice*, specifies how to scale the SVG document to the `viewBox`. The value for this argument can be either `meet` or `slice`. If you specify `meet`, the SVG document is scaled up as much as possible while meeting other criteria. The viewport displays some unused space. If you specify `slice`, the SVG document is scaled down as much as possible while meeting other criteria. In the latter case, some of the SVG document appears to be cut off. The SVG document is still complete, but you cannot see all of it in the viewport. You can use your browser controls to move the SVG document around in the viewport.

For more information, see “[SVGPRESERVEASPECTRATIO= System Option](#)” in [SAS System Options: Reference](#).

Including the Magnify Tool in SVG Documents

You can include a magnify tool in SVG documents by setting the SVGMAGNIFYBUTTON system option. When the tool is enabled, a magnifying glass is available to enlarge a portion of an SVG document. The size of the magnification area cannot be changed.



By default, the magnify tool is not included in the SVG document. You must explicitly set the SVGMAGNIFYBUTTON system option. You can use this `OPTIONS` statement:

```
options svgmagnifybutton;
```

To disable the magnify tool, use the NOSVGMAGNIFYBUTTON system option.

When the magnify tool is enabled, the **Magnify** button, , appears at the top of the SVG document. To make the magnifying glass appear, click **Magnify**. Using the mouse, move the magnifying glass over the SVG document to enlarge the area under the glass. To turn the magnifying glass off, click **Magnify** again. By default, the magnifying glass enlarges the area using a magnification level of three. Click on **+** to increase the magnification level and **-** to decrease the magnification level. When the SVG document is viewed on an iPad, the first tap of the **Magnify** button displays the tooltip. The second and subsequent taps change the magnification and closes the magnifying glass.

There are some restrictions for using the magnify tool:

- The magnify tool is not supported for the SVGT printer and animated SVG documents.
- When you use the SVGnotip printer, no tooltip is displayed to tell you to enable or disable the magnifying glass.
- When the magnify tool is enabled, the magnify tool is turned off on the Index page of a multi-page document.

The magnify tool is more useful when the SVG document is viewed in browsers that expect an SVG document to control the zoom level.

For more information, see [“SVGMAGNIFYBUTTON System Option” in SAS System Options: Reference](#).

Adding a Title to an SVG Document

You use the SVGTITLE= system option to add a title to the title bar of a window when the browser displays only the SVG document. If the SVG document is embedded in an HTML page, the svgtitle attribute on the <svg> tag has no effect. The static viewport example in the previous topic shows a title in the browser title bar.

For more information, see [“SVGTITLE= System Option” in SAS System Options: Reference](#).

Creating Stand-alone SVG Documents Using the ODS PRINTER Destination

Creating an SVG Document

To create an SVG document, you need to at least set the PRINTERPATH= system option to an SVG universal printer and specify the ODS PRINTER statement in your SAS program. Or, specify the PRINTER= option in the ODS PRINTER statement:

```
options printerpath=svg;
ods _all_ close;
ods printer;
proc sgplot data=sashelp.class;
    reg x=height y=weight / CLM CLI;
run;
ods printer close;
ods html;
```

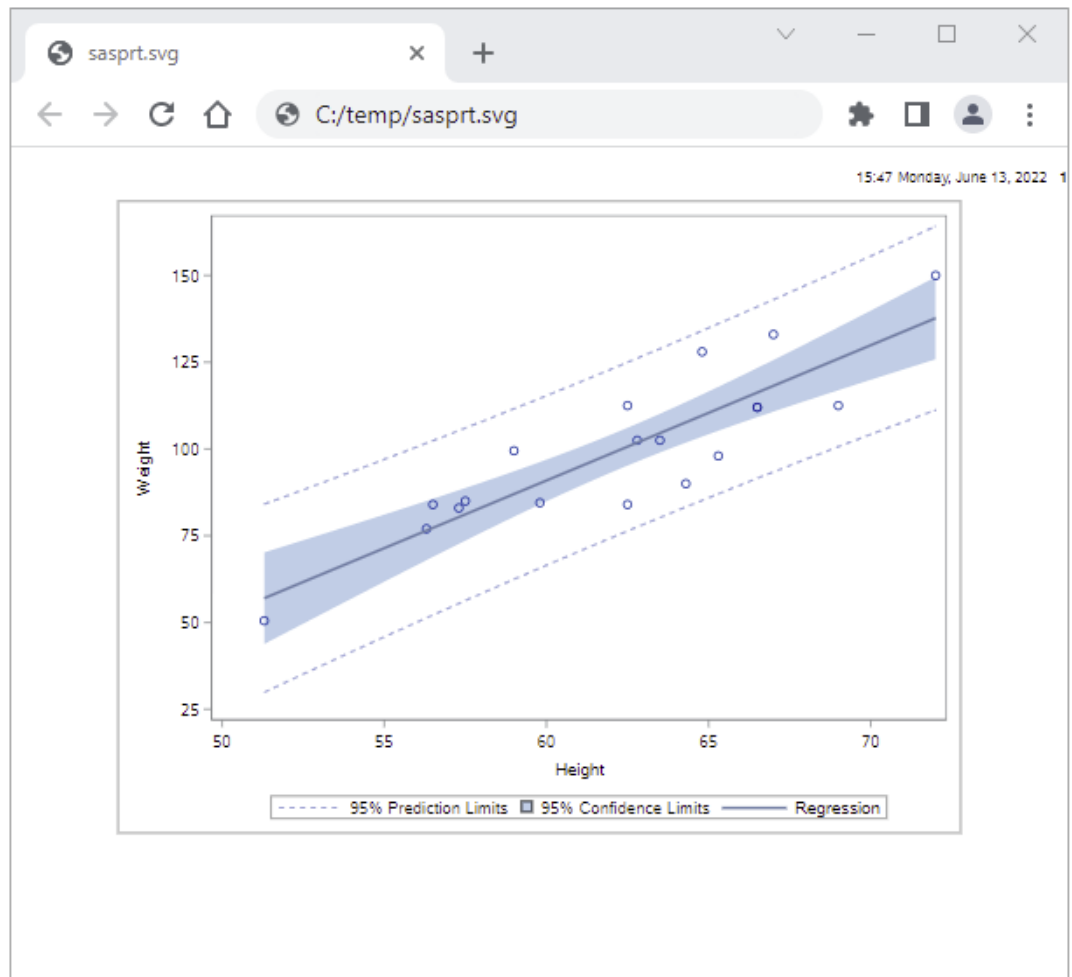
In this example, no specific SVG system option values were set to size the SVG document. Therefore, the viewBox is the default size specified by the PAPERSIZE= system option. The SVG document scales to the viewport because no value was specified for the SVGWIDTH= and SVGHEIGHT= system options. The following is the <svg> element that SAS creates:

```
<svg xmlns="http://www.w3.org/2000/svg"
    xmlns:xlink="http://www.w3.org/1999/xlink"
    xml:space="preserve" baseProfile="full" version="1.1"
    id="SVGMain" onload='SVGMain_Init("SVGMain")'
    viewBox="-1 -1 801 601">
```

SAS creates a single SVG document named `sasprt.svg` and stores it in a specific location, depending on your operating environment. Under Windows, the file is stored in the current directory. Under UNIX, the file is stored in your home directory. Under z/OS, the file is stored as a z/OS UNIX System Services Hierarchical File System (HFS) file, or as a z/OS data set. If the SVG file is written to a z/OS data set, it is written to PDSE library `userid.SASPR.T.SVG`. You can use the `FILE=` option in the ODS PRINTER statement to specify a different file name.

The following figure is an SVG file that uses the Adobe Acrobat SVG plug-in for Microsoft Internet Explorer. This file was created by using the SGPLOT procedure to plot the `Sashelp.Class` data set.

Figure 5.3 *Sashelp.Class Plotted by the SGPLOT Procedure as an SVG File*



When you use the SVG, SVGnotip, SVGt, SVGView, and SVGZ Universal Printers, SAS creates a single SVG document. Depending on the size of the SVG document, the browser might display the complete SVG document. Check the documentation for your browser to determine whether your browser has controls for viewing SVG documents. In the Adobe SVG Viewer plug-in for Internet Explorer, you can press the Alt key and the left mouse button to pan and move to different pages in a continuous, multi-page SVG document.

Multi-Page SVG Documents in a Single File

When a DATA step or procedure creates a new page in the output, a new SVG page is created in an SVG document. SAS creates either one file with multiple pages or multiple SVG files with one file for each SVG document page. The SVGCONTROLBUTTONS system option and the NEWFILE= option in the ODS PRINTER statement control whether a multi-page SVG document is one continuous file (with controls to navigate the pages in the file) or multiple SVG files.

SAS creates a single-file, multi-page SVG document with navigational controls when the NEWFILE= option of the ODS PRINTER statement is a value other than PAGE, and one of the following set of system options is specified:

- The PRINTERPATH= system option is set to SVG or SVGZ, and the SVGCONTROLBUTTONS system option is set.
- The PRINTERPATH= system option is set to SVGView.

The SVGView Universal Printer enables the SVGCONTROLBUTTONS system option.

If the SVGCONTROLBUTTONS system option is not specified or the Universal Printer is not SVGView, the SVG document is created in a continuous-page layout. Here is an example.

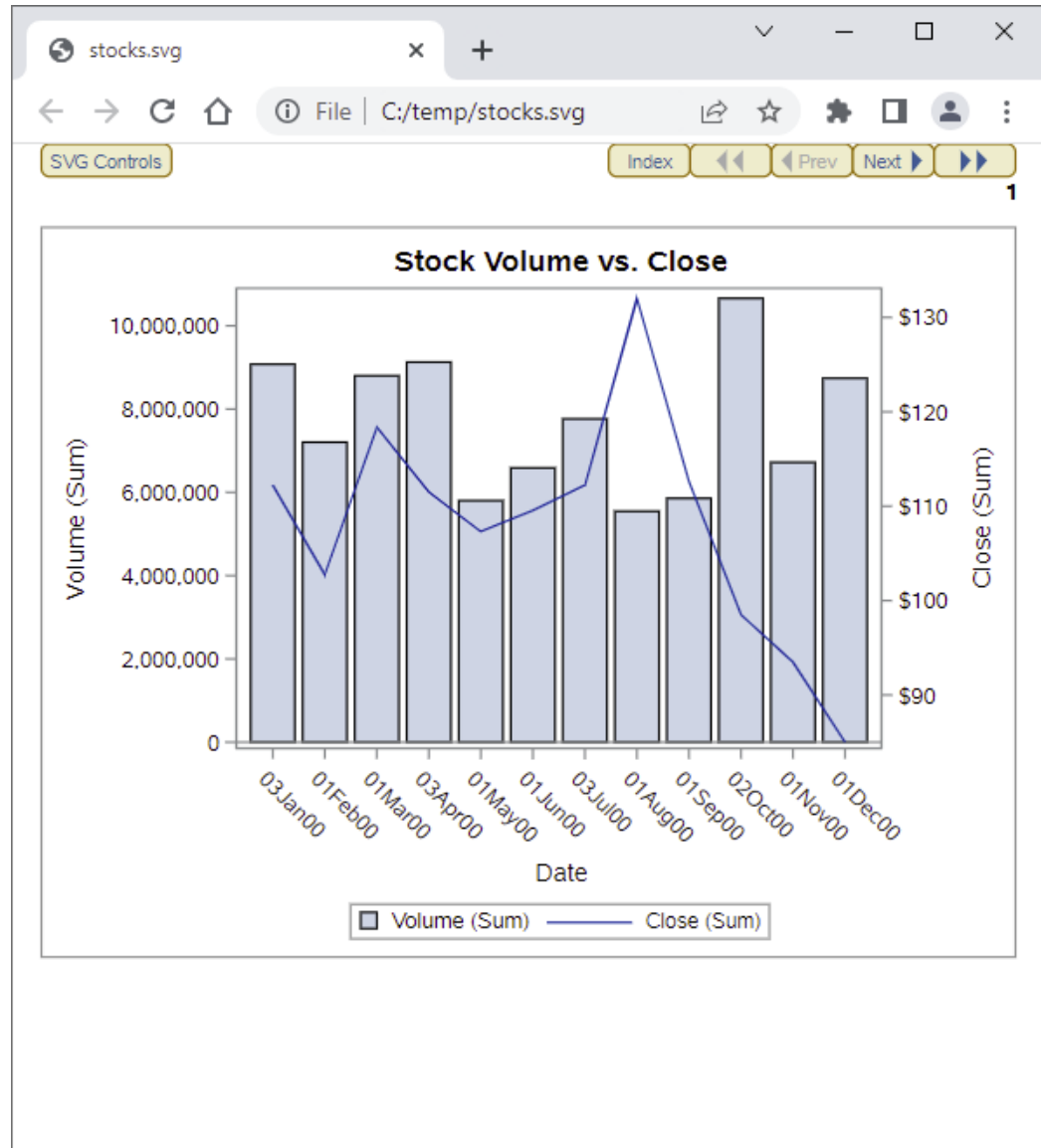
Example Code 5.1 Program that Creates a Multi-Page SVG Document

```
options nodate printerpath=(svgview stocks) papersize=("5in" "5in");
filename stocks 'c:\temp\stocks.svg';
ods _all_ close;
ods printer;
proc sgplot data=sashelp.stocks (where=(date >= "01jan2000"d
                                     and date <= "01jan2001"d
                                     and stock = "IBM"));
    title "Stock Volume vs. Close";
    vbar date / response=volume;
    vline date / response=close y2axis;
run;
title;
proc sgplot data=sashelp.stocks
    (where=(date >= "01jan2000"d and stock = "IBM"));
    title "Stock Trend";
    series x=date y=close;
    series x=date y=low;
    series x=date y=high;
run;
title;
title "Stock High, Low, and Close";
proc sgplot data=sashelp.stocks;
    where Date >= '01JAN2005'd and stock='IBM';
    highlow x=date high=high low=low
        / close=close;
run;
title;
```

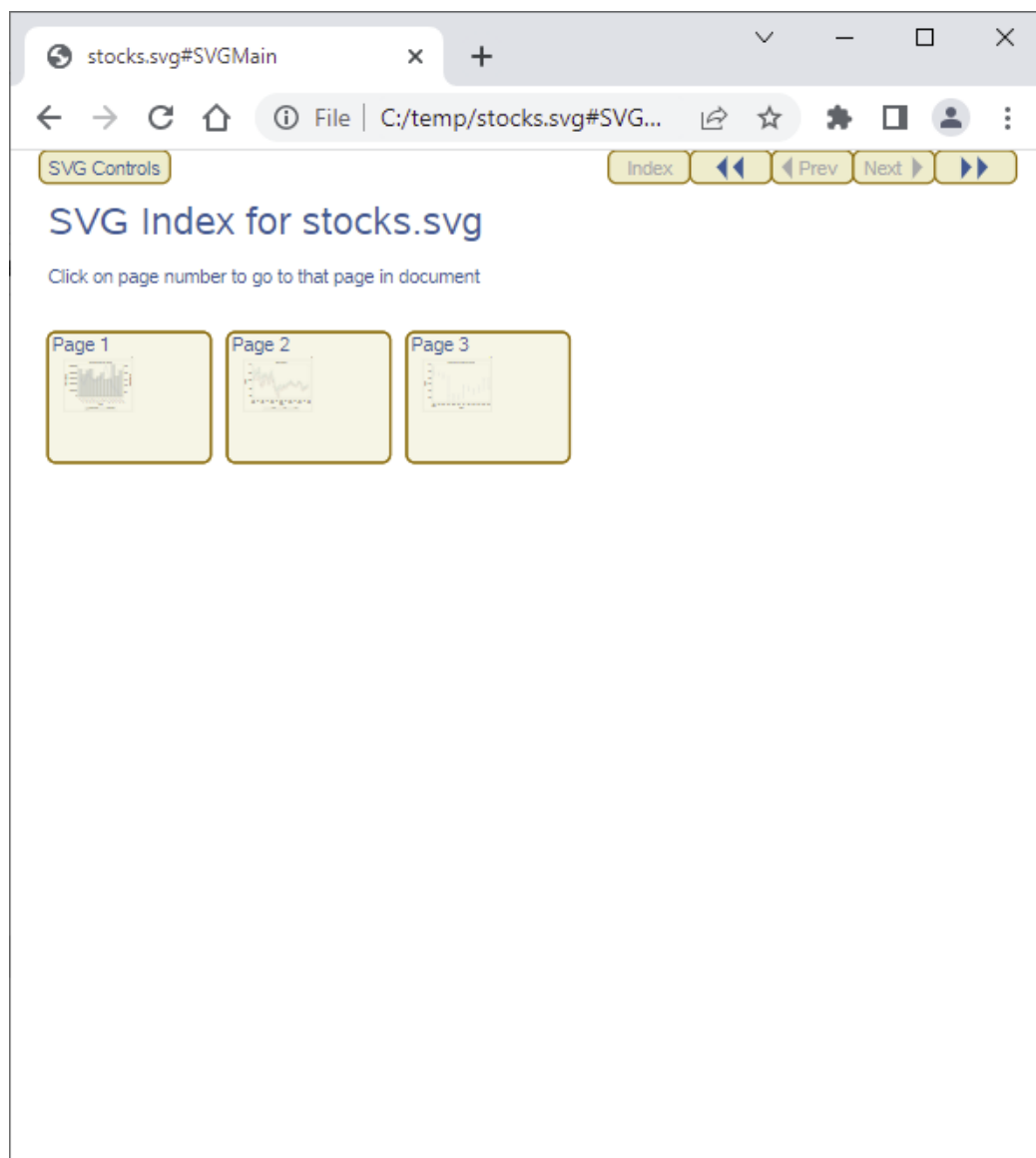
```
ods printer close;
options papersize=("","");
ods html;
```

Here is the program output.

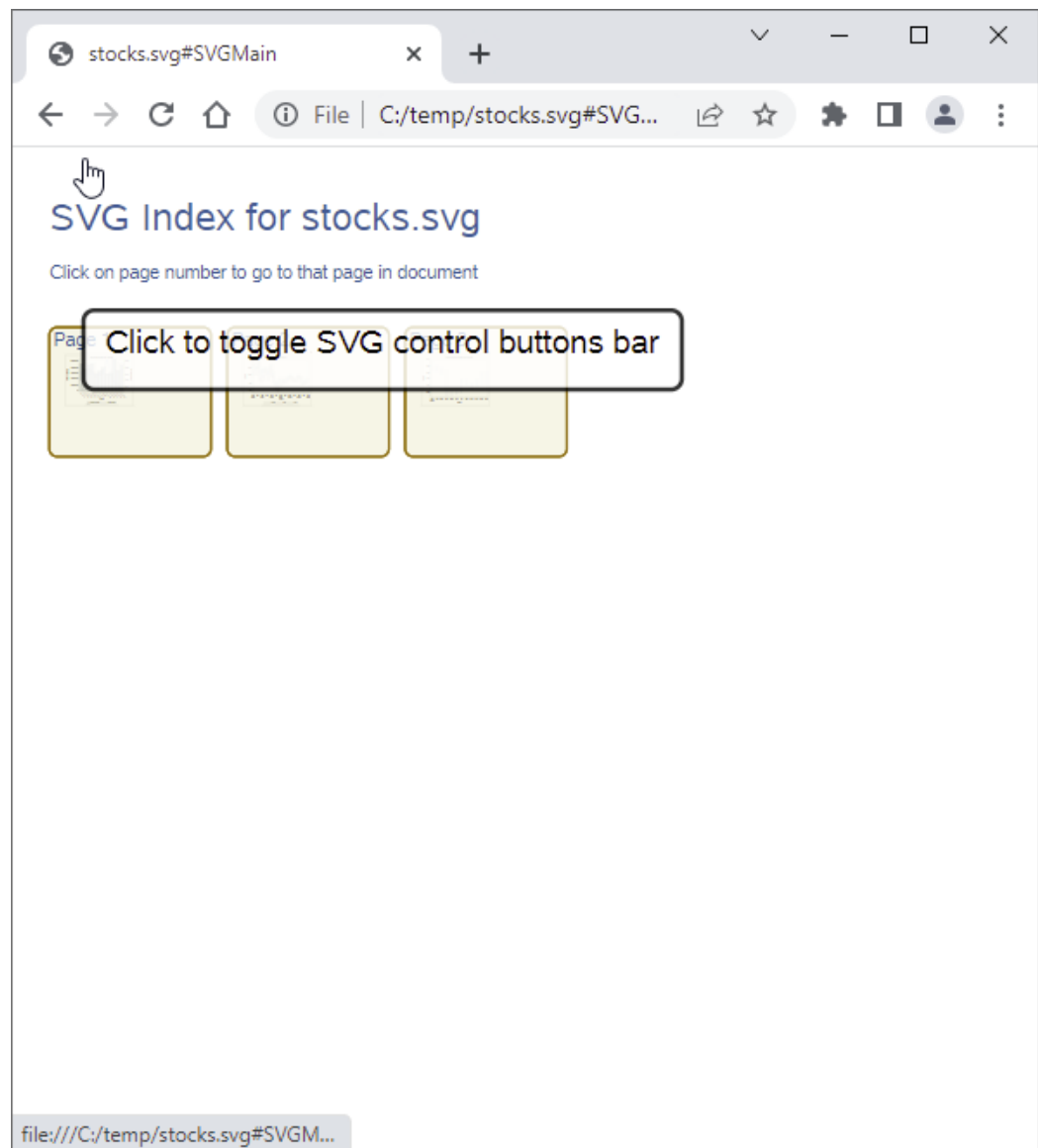
Figure 5.4 First Page of a Multi-page SVG File with Navigation Controls



To navigate the document, use your browser controls. The navigation controls enable you to go to the next page, the previous page, the first page, or the last page; to display an index of all pages; or to hide or show the controls. To display an index of all pages in the SVG file, select the **Index** button. To go to a specific page from the index, select the thumbnail image of the page.

Figure 5.5 Index of a Multi-page SVG File with Navigation Controls

You can hide the control buttons by selecting the **SVG Controls** button. The tooltip is displayed when the cursor is over the control. To show the navigation controls again, click in the top area of the output when you see the tooltip **Click to toggle SVG control button bar**. This is useful when you want to print a page in the document without the SVG controls.

Figure 5.6 A Multi-page SVG File That Hides the Navigational Controls

For information about the NEWFILE= option, see [ODS PRINTER Statement](#) in *SAS Output Delivery System: User's Guide*.

Animating Multi-Page SVG Files

You can animate multi-page SVG files using SAS system options. For more information, see [“Creating Animated GIF Images and SVG Documents”](#) on page 72.

Creating Separate Files for Multi-Page SVG Documents

You can create a separate file for each page in an SVG document by specifying the `NEWFILE=PAGE` option in the `ODS PRINTER` statement. A new page is created when a procedure explicitly starts a new page and not when the page size is exceeded. The first file is named *filename.svg*. Subsequent file names include an appended number, starting with the number 1: *filename1.svg*, *filename2.svg*, and so on.

Using the default file name `sasprt.svg`, the following code creates three files:

- `sasprt.svg` contains the output from the first `SGPLOT` procedure.
- `sasprt1.svg` contains the output from the second `SGPLOT` procedure.
- `sasprt2.svg` contains the output from the third `SGPLOT` procedure.

```
options nodate printerpath=svgview papersize=("5" "5");
ods _all_ close;
ods printer newfile=page;
proc sgplot data=sashelp.stocks (where=(date >= "01jan2000"d
                                     and date <= "01jan2001"d
                                     and stock = "IBM"));
    title "Stock Volume vs. Close";
    vbar date / response=volume;
    vline date / response=close y2axis;
run;
title;
proc sgplot data=sashelp.stocks
    (where=(date >= "01jan2000"d and stock = "IBM"));
    title "Stock Trend";
    series x=date y=close;
    series x=date y=low;
    series x=date y=high;
run;
title;
title "Stock High, Low, and Close";
proc sgplot data=sashelp.stocks;
    where Date >= '01JAN2005'd and stock='IBM';
    highlow x=date high=high low=low
        / close=close;
run;
title;
ods printer close;
options papersize=("","");
ods html;
```

For information about the `NEWFILE=` option, see [ODS PRINTER Statement](#) in *SAS Output Delivery System: User's Guide*.

Creating Overlaid Transparent SVG Documents

You use the SVTt Universal Printer to create a transparent SVG document in which the pages are transparent and can be overlaid. The following is a SAS program that overlays a bar chart on a map of the United States:

```
data boxanno;
  length function color style $20 text $16;
  retain xsys ysys '2' hsys '3' when 'a';
  set maps.uscity(keep=x y city state);
  where city='Raleigh' and state=stfips('NC');
  color='blue'; size=4; text='V'; position='5'; style='marker';
output;
  myx=x;
  myy=y;
  function='move';
  x=myx; y=myy; output;
  function='draw';
  x=myx-.432; y=myy+.0417; color='slategray'; line=1; size=.2;
style='solid'; output;
  function='move';
  x=myx; y=myy; output;
  function='draw';
  x=myx-.432; y=myy+.178; output;
  function='move';
  x=myx; y=myy; output;
  function='draw';
  x=myx-.251; y=myy+.178; output;
  function='move';
  x=myx; y=myy; output;
  function='draw';
  x=myx-.251; y=myy+.0417; output;

run;

%let name=annomap;
filename odsout '.';
ods _all_ close;
goptions reset=all border gunit=pct htitle=3 htext=2 ftext="arial/bo"
         iback='path-to-background-image'
         imagestyle=fit;
options printerpath=svgt papersize=("6in", "5in") nodate nonumber;
ods printer file='annomap.svg' ;
pattern1 v=s c=cornsilk;

title1 c=orange "SAS/GRAPH Map and Overlaid Bar Chart with
PRINTERPATH=SVGT";
proc gmap data=maps.us map=maps.us ;
id state;
choro state / levels=1 nolegend coutline=blue anno=boxanno
  des="" name="&name";
run;
```

```
quit;

options iback= hsize=1.46 vsize=1.11 horigin=1.54 vorigin=2.52
autosize=on dev=svg;

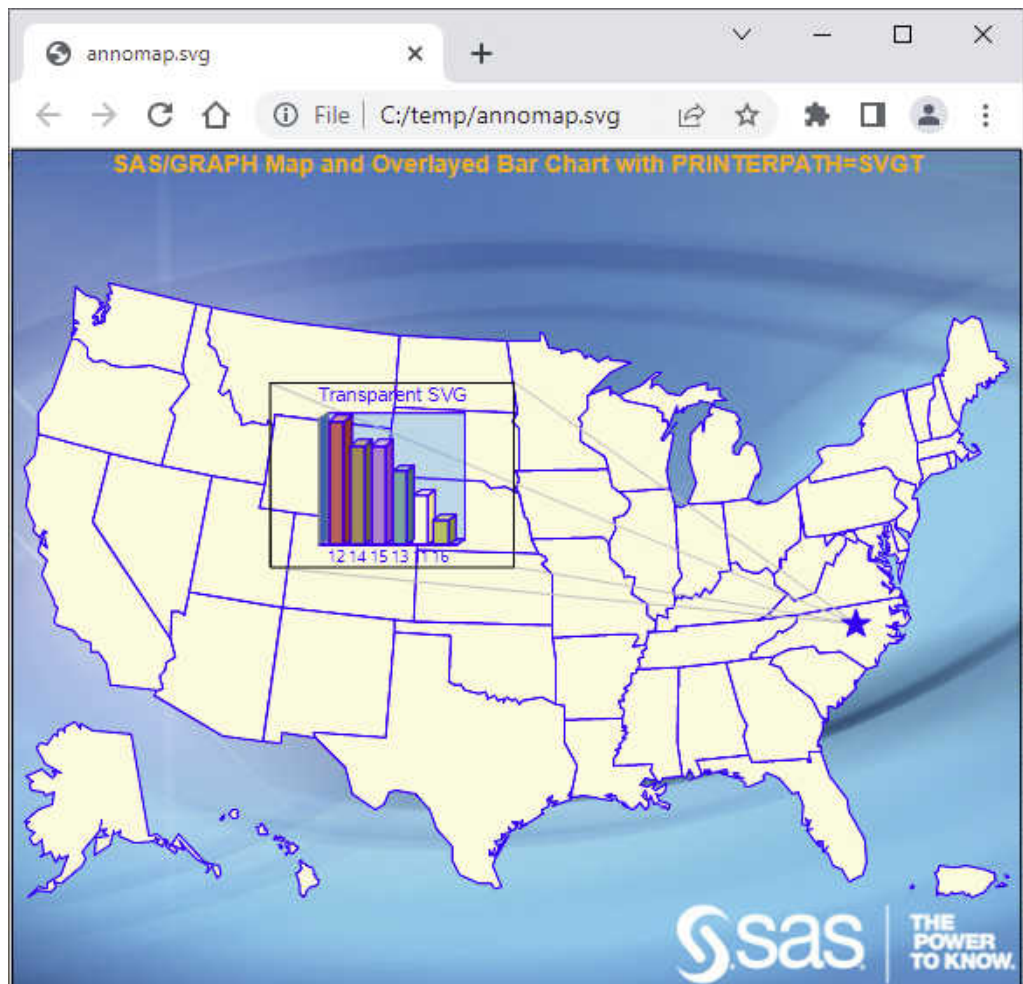
/* you must use the default ods style, for transparency to work */

options gunit=pct htext=8 ftext="Albany AMT" ;
title c=blue h=10 'Transparent SVG';
axis1 label=none value=none major=none minor=none style=0;
axis2 color=blue label=none offset=(7,7) value=(color=blue);
proc gchart data=sashelp.class;
  vbar3d age / discrete patternid=midpoint
  descending raxis=axis1 maxis=axis2 width=5 space=2
  frame cframe=rgba0195FF50 coutline=blue woutline=1
  des="" name="&name.b";
run;
quit;

ods printer close;
```

This code creates the following SVG:

Figure 5.7 A Bar Chart Overlaying a SAS/GRAPH Map



See [“Enhancing Drill-Down Behavior in SVG Presentations Using HTML Attributes” in SAS/GRAPH: Reference](#) for an example of using overlaid images for drill-down links in graphs.

Inserting SVG Images into HTML Documents

Overview of Inserting SVG Images into HTML Documents

To insert an SVG image into an HTML document, you can do one of the following:

- Insert the SVG image using an `<EMBED>` tag that references the SVG image file.

.....
Note: This is the only method that is supported by the ODS HTML destination.

- Insert the SVG image using an `<IMG>` tag that references the SVG image file (ODS HTML5 destination only).
- Insert the SVG image data inline in the HTML code (ODS HTML5 destination only).

.....
Note: This is the default method for the ODS HTML5 destination.

- Insert a link to the SVG image file using an `<A HREF="image-file-url">` tag that links to the SVG image file (ODS HTML5 destination only).

You can use ODS Graphics or SAS/GRAPH, if installed, to generate an SVG image. For information about how to generate SVG images in ODS Graphics, see [“Specifying the Image Format” in SAS ODS Graphics: Procedures Guide](#). For information about how to create SVG images in SAS/GRAPH, see [“Generating SVG, PNG, GIF, and TIFF Graphics” in SAS/GRAPH: Reference](#). You can use an ODS HTML5 statement to generate and insert SVG images into an HTML document or you can generate SVG images separately and manually insert them into an HTML document.

About the SVG Viewbox and Viewport Sizes

In an SVG file, the `<SVG>` element `viewBox=`, `height=`, and `width=` attributes specify the dimensions of the viewBox and the viewport. Here is an example:

```
<svg xmlns="http://www.w3.org/2000/svg" xmlns:xlink="http://www.w3.org/1999/xlink"
  xml:space="preserve" baseProfile="full" version="1.1" id="SVG_IDX"
  onload='SVG_IDX_Init("SVG_IDX")' viewBox="-1 -1 641 481" height="360px" width="480px">
```

In this example, the `viewBox` attribute specifies a viewBox that is 641 pixels wide and 481 pixels high. The `width=` and `height=` attributes specify a viewport that is

480 pixels wide and 360 pixels high. In this example, the viewport occupies 75% of the viewBox area. If the height and width attributes are not specified, the browser determines the image display size, unless the height and width attributes are specified in the <EMBED>, , or tag in the HTML code.

If you use ODS Graphics or SAS/GRAPH to create your SVG image and you used the default values for the SAS SVG system options, the dimensions of the viewBox and viewport are determined as follows:

- For an SVG image that is generated with ODS Graphics using the ODS HTML5 destination, the viewBox and viewport sizes are determined by the WIDTH= and HEIGHT= values specified in the ODS Graphics statement. By default, WIDTH=640px and HEIGHT=480px. The viewBox is sized slightly larger than the specified WIDTH= and HEIGHT=:

```
viewBox="-1 -1 641 481" height="480" width="640"
```

For an SVG image that is generated with ODS Graphics using the ODS HTML destination, the viewBox size is determined by the WIDTH= and HEIGHT= values specified in the ODS Graphics statement:

```
viewBox="-1 -1 641 481"
```

The viewport size is not specified.

If you specify values for system option SVGVIEWBOX=, its values determine the size of the viewBox. System option PAPERSIZE= has no effect when ODS Graphics is used with the ODS HTML5 or ODS HTML destination to generate the SVG image. If you specify values for system options SVGWIDTH= and SVGHEIGHT=, their values determine the viewport dimensions and override the WIDTH= and HEIGHT= options in the ODS GRAPHICS statement.

- For an SVG image that is generated with SAS/GRAPH using the ODS HTML or ODS HTML5 destination, the viewBox size is determined by the default value of system option PAPERSIZE=, which is based on locale. For example, in the English locale, PAPERSIZE= defaults to LETTER, which has a width of 8.5 inches (816 pixels) and a height of 11 inches (1056 pixels). The viewBox is sized slightly larger than the dimensions specified in PAPERSIZE=:

```
viewBox="-1 -1 817 1057"
```

Because there is no default value for the SVGHEIGHT= and SVGWIDTH= system options, the viewport dimensions are not specified in the default case. If you specify values for system option SVGVIEWBOX=, its values determine the size of the viewBox and override PAPERSIZE=. If you specify values for system options SVGWIDTH= and SVGHEIGHT=, their values determine the viewport dimensions.

- For an SVG image that is generated with ODS Graphics or SAS/GRAPH using the ODS PRINTER destination, the viewport size is determined by the default value of system option PAPERSIZE=. The viewBox is sized slightly larger than the dimensions specified in PAPERSIZE=:

```
viewBox="-1 -1 817 1057"
```

Because there is no default value for the SVGHEIGHT= and SVGWIDTH= system options, the viewport dimensions are not specified in the default case. If you specify values for system option SVGVIEWBOX=, its values determine the

size of the viewBox and override PAPERSIZE=. If you specify values for system options SVGWIDTH= and SVGHEIGHT=, their values determine the viewport dimensions.

The viewport dimensions specified in the SVG image file can be overridden by specifying the height and width attributes in the <EMBED>, , or tag in the HTML code.

Inserting an SVG Image Using an <EMBED> Tag

The following example uses the ODS HTML5 destination to insert an SVG image that is generated using SAS/GRAPH into an HTML document using an <EMBED> tag:

```
ods _all_ close;
options papersize=("6in", "4in") svgwidth="4.5in" svgheight="3in";
options reset=all device=svg;
ods html5 path="." file="MyGraph.html" options(svg_mode="embed");
proc gchart data=sashelp.cars;
    vbar type / name="MyGraph" sumvar=mpg_highway type=mean
        subgroup=origin;
run;
quit;
ods html5 close;
options papersize=(" ", " ") svgwidth=" " svgheight=" ";
options reset=all;
ods html;
```

Note: The default `svg_mode` for the HTML5 destination is `INLINE`. In order to embed the SVG image, you must specify `SVG_MODE="EMBED"` as an option in the ODS HTML5 statement.

This code creates two files: `mygraph.svg` and `MyGraph.html`. Here are the `viewBox`, `height`, and `width` attributes in the <SVG> element in file `mygraph.svg`:

```
<svg xmlns="http://www.w3.org/2000/svg" xmlns:xlink="http://www.w3.org/1999/xlink"
    xml:space="preserve" baseProfile="full" version="1.1" id="SVG_IDX"
    onload='SVG_IDX_Init("SVG_IDX")' viewBox="-1 -1 577 385" height="3in" width="4.5in">
```

The dimensions in the `viewBox=`, `height=`, and `width=` attributes are determined by the `PAPERSIZE=`, `SVGHEIGHT=`, and `SVGWIDTH=` system options.

Here is the <EMBED> element in file `MyGraph.html`:

```
<embed class="graph" style="border-width: 0; border-spacing: 1px"
    src=".\\gchart2.svg" type="image/svg+xml"/>
```

Because the `height=` and `width=` attributes are not specified in the <EMBED> tag, the `height=` and `width=` specifications in the SVG file determine the viewport size.

Inserting an SVG Image Using an Tag

The following example uses the ODS HTML5 destination to insert an SVG image that is generated using ODS Graphics into an HTML document using an tag:

```
ods _all_ close;
ods graphics / reset imagefmt=svg imgname="MyGraph" width=540px
height=405px;
ods html5 path="." file="MyGraph.html" options(svg_mode="img");
title "Stock Trend";
proc sgplot data=sashelp.stocks
  des="IBM Stock Trends";
  series x=date y=close;
  series x=date y=low;
  series x=date y=high;
  where date >= "01jan2000"d and stock = "IBM";
run;
ods html5 close;
ods html;
```

This code creates two files: MyGraph.svg and MyGraph.html. Here are the viewBox, height, and width attributes in the <SVG> element in file MyGraph.svg:

```
<svg xmlns="http://www.w3.org/2000/svg" xmlns:xlink="http://www.w3.org/1999/xlink"
xml:space="preserve" baseProfile="full" version="1.1" id="SVG_IDX"
onload='SVG_IDX_Init("SVG_IDX")' viewBox="-1 -1 541 406" height="405" width="540">
```

The dimensions in the viewBox=, height=, and width= attributes are determined by the WIDTH= and HEIGHT= option values in the ODS GRAPHICS statement.

Here is the element in file MyGraph.html:

```

```

Because the height= and width= attributes are specified in the tag, the height and width specifications in the <SVG> tag in file MyGraph.svg are ignored.

Inserting an SVG Image Using an Tag

The following example uses the ODS HTML5 destination to insert an SVG image that is generated using ODS Graphics into an HTML document using an tag:

```
data ibmstock;
  set sashelp.stocks;
  where year(date) between 2000 and 2004 and stock eq "IBM";
  year = year(date);
  keep year date open close volume;
run;
```

```

proc summary data=ibmstock nway;
    var close volume;
    by year notsorted;
    output out=ibm mean(close volume)=avgclose avgvolume;
run;

proc sort data=ibm;
    by year;

run;

%macro generateGraph(year);
    title2 "Closing Price and Volume in &year";
    proc sgplot
        data=ibmstock(where=(year(date) eq &year))
        des="Graph of IBM Stock Trends in &year";
        series x=date y=close / smoothconnect;
        series x=date y=volume / smoothconnect y2axis;
        xaxis tickvalueformat=monname3.;
    run;
    title2;
%mend generateGraph;

ods _all_ close;
ods graphics / reset imagefmt=svg imagename="MyGraph" width=540px
height=405px;
ods html5 path="." file="MyGraph.html" options(svg_mode="link");
ods layout gridded;
title1 "IBM Stock Trends: 2000 - 2004";

ods region;
title2 "Yearly Mean Close and Volume";
proc print data=ibm label noobs;
    var year avgclose avgvolume;
    label year="Year" avgclose="Average Close" avgvolume="Average
Volume";
run;

%generateGraph(2000);
%generateGraph(2001);
%generateGraph(2002);
%generateGraph(2003);
%generateGraph(2004);

title;
ods layout end;
ods html5 close;
ods html;

```

This code creates HTML file MyGraph.html, and SVG files MyGraph.svg–MyGraph4.svg. Here are the viewBox, height, and width attributes in the <SVG> element in the SVG files:

```

<svg xmlns="http://www.w3.org/2000/svg" xmlns:xlink="http://www.w3.org/1999/xlink"
xml:space="preserve" baseProfile="full" version="1.1" id="SVG_IDX1"
onload='SVG_IDX1_Init("SVG_IDX1")' viewBox="-1 -1 541 406" height="405" width="540">

```

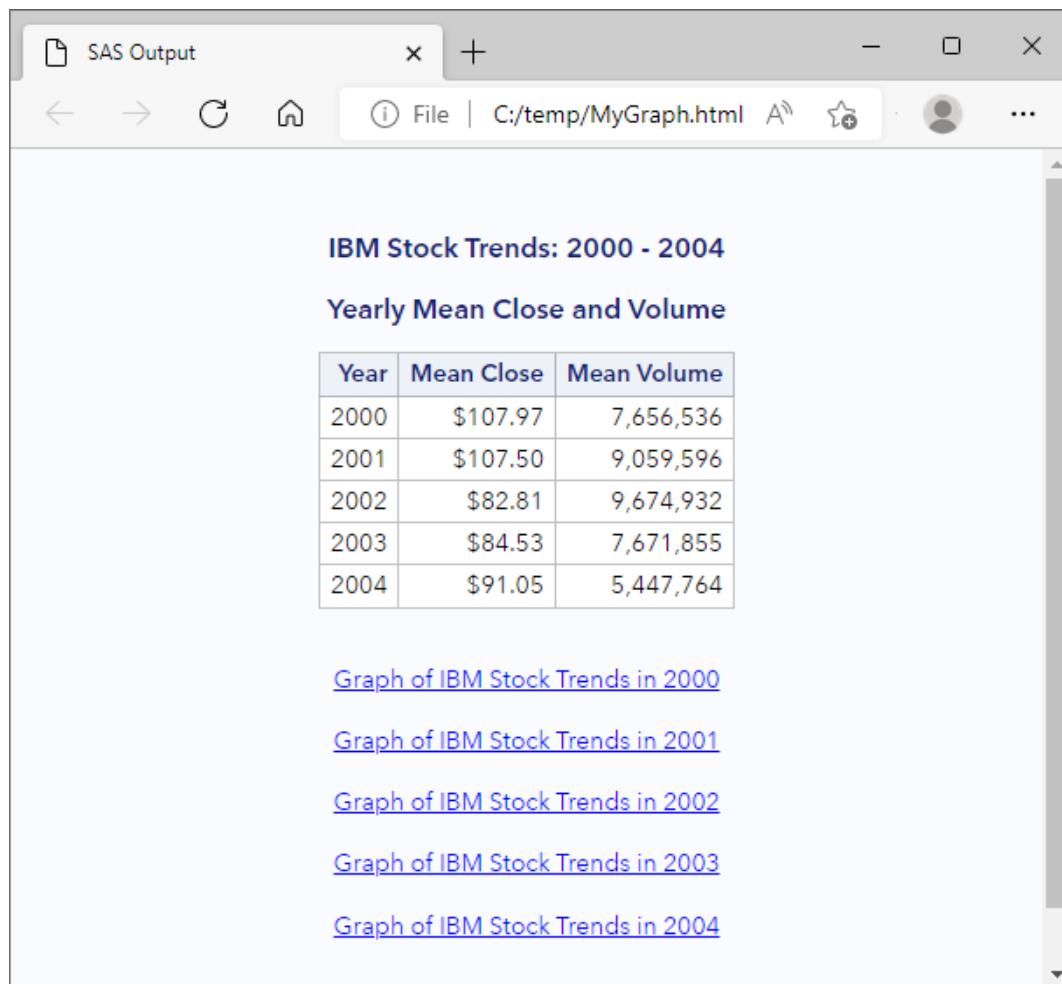
The dimensions in the `viewBox=`, `height=`, and `width=` attributes are determined by the `WIDTH=` and `HEIGHT=` option values in the ODS GRAPHICS statement.

Here is the `<IMG>` element for the first graph in file `MyGraph.html`:

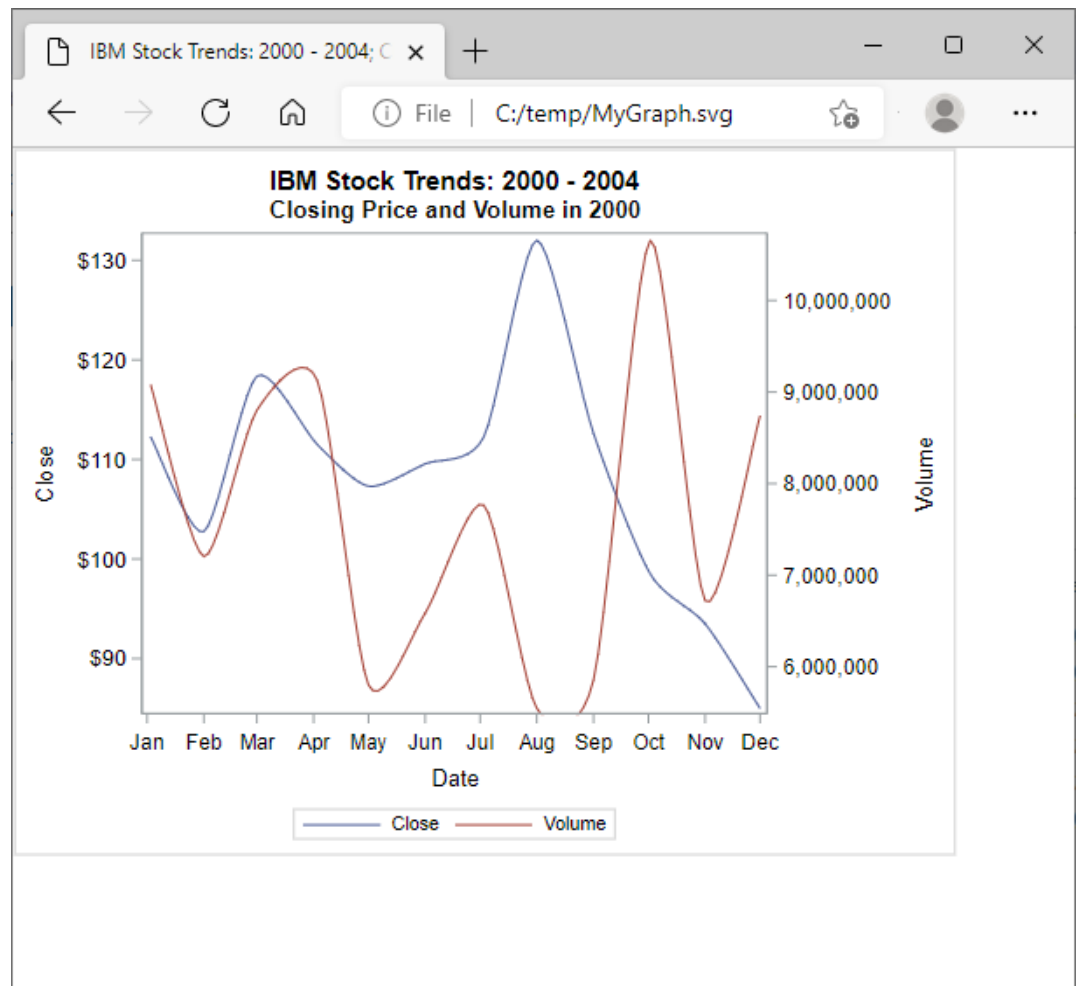
```
<a style="height: 405px; width: 540px" href="MyGraph.svg">
  Graph of IBM Stock Trends in 2000</a>
```

Because the `height=` and `width=` attributes are specified in the `<IMG>` tag, the height and width specifications in the `<SVG>` tag in the SVG files are ignored.

Here is the HTML file displayed in a browser.



Clicking the first graph link displays the following graph:

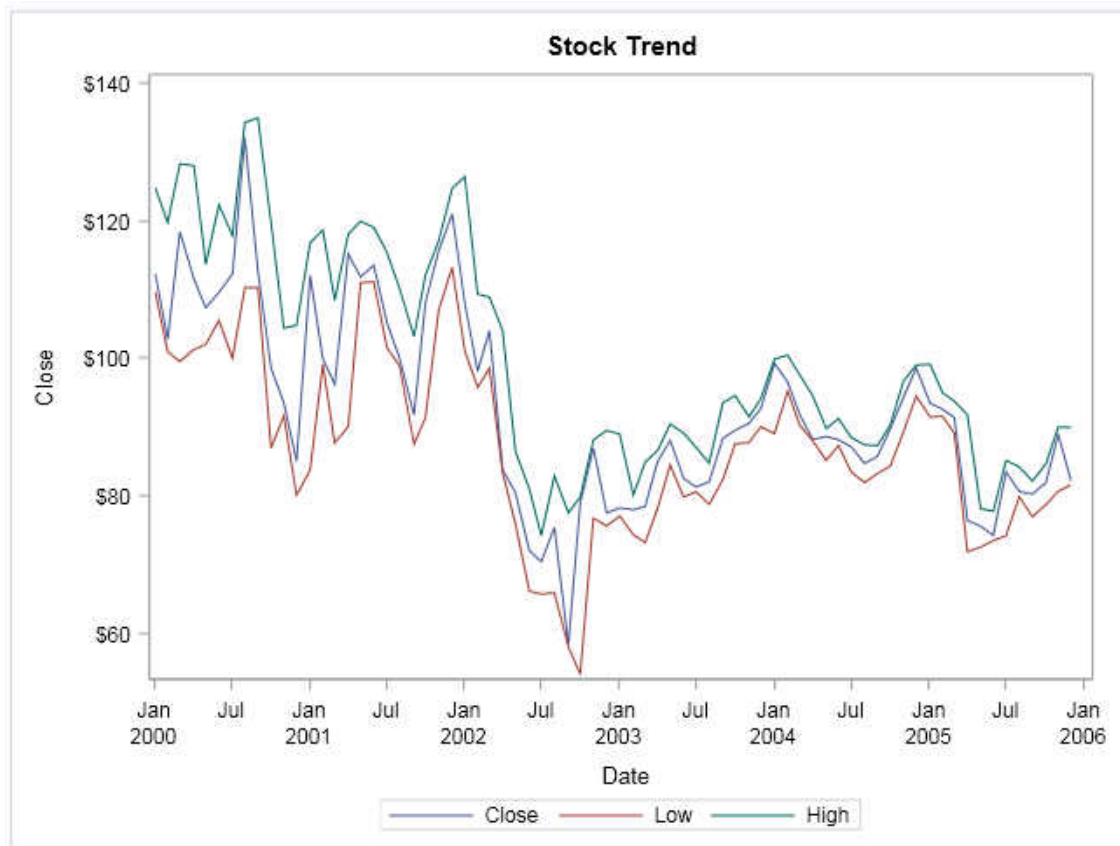


Inserting an SVG Image Inline in an HTML5 Document

The following example uses the ODS HTML5 destination to insert an SVG image that is generated using ODS Graphics inline in an HTML document:

```
ods html close;
ods html5 path="." file="MyGraph.html" options(svg_mode="inline");
ods graphics / reset imagefmt=svg;
proc sgplot data=sashelp.stocks
  (where=(date >= "01jan2000"d and stock = "IBM"));
  title "Stock Trend";
  series x=date y=close;
  series x=date y=low;
  series x=date y=high;
run;
ods html5 close;
ods html;
```

Here is the HTML file with the inline SVG graph:



This code creates HTML file MyGraph.html. Here are the viewBox, height, and width attributes in the <SVG> element in file MyGraph.html:

```
<svg xmlns="http://www.w3.org/2000/svg" xmlns:xlink="http://www.w3.org/1999/xlink"
  xml:space="preserve" baseProfile="full" version="1.1" id="SVG_IDX"
  onload='SVG_IDX_Init("SVG_IDX")' viewBox="-1 -1 641 481" height="480" width="640">
```

The dimensions in the viewBox=, height=, and width= attributes are determined by the default values of the ODS GRAPHICS statement WIDTH= and HEIGHT= options.

Creating TIFF Images Using Universal Printing

TIFF Images in SAS

Tagged image file format (TIFF) images are raster images that are commonly used in word processing, scanning, faxing, and other applications.

SAS supports TIFF 6.0 for RGBA and CMYK colors. When SAS creates a TIFF image, the image is compressed. If the NOUPRINTCOMPRESSION system option is set, the size of a TIFF image file that SAS creates is extremely large.

For a description of the TIFF printer, you can either view the printer in the SAS registry or submit the following QDEVICE procedure and view the output in the SAS log:

```
proc qdevice;
  printer tiff;
run;
```

See Also

[“Color Support for Universal Printers” in SAS Language Reference: Concepts](#)

The TIFF Universal Printers

SAS provides these Universal Printers:

Printer Name	Description
TIFF	produces TIFF images using RGBA color and transparency.
TIFFk	produces TIFF images using CMYK color. Transparency is not supported.

TIFF printers do not support multiple-page documents. If a procedure creates multiple pages or if more than one procedure is used in the code for ODS PRINTER output, only the first page is viewable.

Creating a TIFF Image

You can create a TIFF image using the ODS PRINTER statements. You specify the TIFF Universal Printer as the value of the PRINTERPATH= system option or as the value of the PRINTER= option in the ODS PRINTER statement.

Here is sample code to create a TIFF image:

```
ods html close;
ods printer printer=tiff;

...more SAS code...

ods printer close;
```

```
ods html;
```

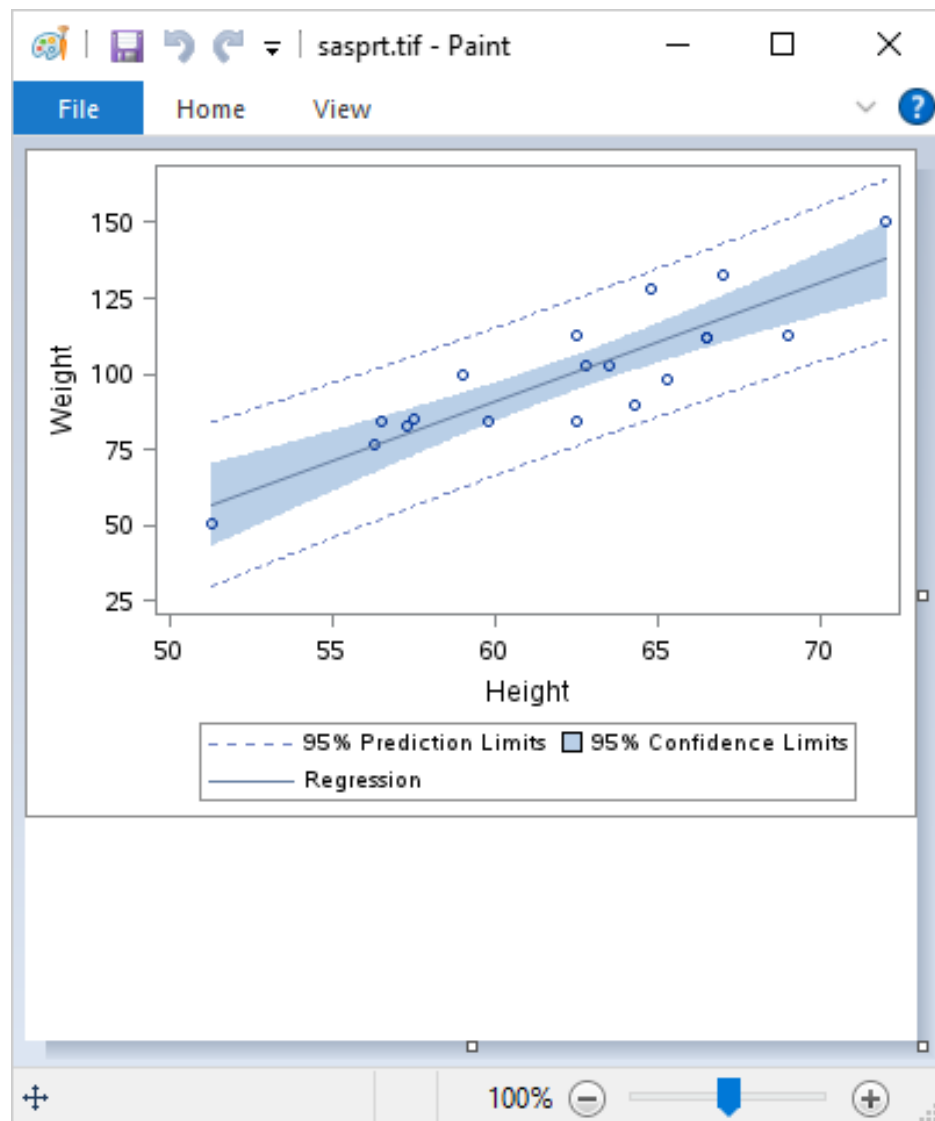
SAS creates the file sasprt.tif in the current directory.

In SAS/GRAPH, the TIFF device is a shortcut to the TIFF Universal Printer. For information about creating TIFF images using SAS/GRAPH devices, see [SAS/GRAPH: Reference](#).

Example of Creating a TIFF Image Using the ODS PRINTER Statement

Using the example data set Sashelp.Class and the SGPLOT procedure, the following ODS PRINTER statement prints the TIFF file sasprt.tif in the current directory:

```
options printerpath=tiff papersize=("4in" "4in") nodate;  
ods html close;  
ods printer;  
proc sgplot data=sashelp.class;  
    reg x=height y=weight / CLM CLI;  
run;  
ods printer close;  
ods html;
```



Creating Animated GIF Images and SVG Documents

About Animated GIF Images and SVG Documents

When you create a multi-page GIF image or SVG document using the ODS PRINTER destination, you can animate the GIF image or SVG document that is created by setting SAS system options. Each page in the GIF image or SVG

document creates one frame in the output file. The system options enable you to configure these animation attributes:

- start or stop creating an animated file
- the amount of time that a frame is in view
- whether frames are overlaid or are played sequentially
- the number of times an animation loop is repeated
- for SVG documents only, whether to immediately start the animation when the document is loaded in the web page
- for SVG documents only, whether a frame fades in and out of view and if during the fade-in and fade-out time, the frames are overlaid or played sequentially

SAS/GRAPH is required to create animated files for the ODS HTML5 and the ODS HTML destinations. For more information, see [SAS/GRAPH: Reference](#).

You set the options using the **OPTIONS** statement before opening the ODS **PRINTER** destination:

```
options printerpath=gif animation=start animduration=5 animloop=yes
noanimoverlay;
ods printer file='myfile.gif';
```

In this **OPTIONS** statement, the **ANIMATION** option starts creating the animation file, the **ANIMDURATION** option specifies that each frame is held for 5 seconds. The **ANIMLOOP** option specifies to continuously repeat the animation loop. The **NOANIMOVERLAY** option specifies that each frame is played sequentially.

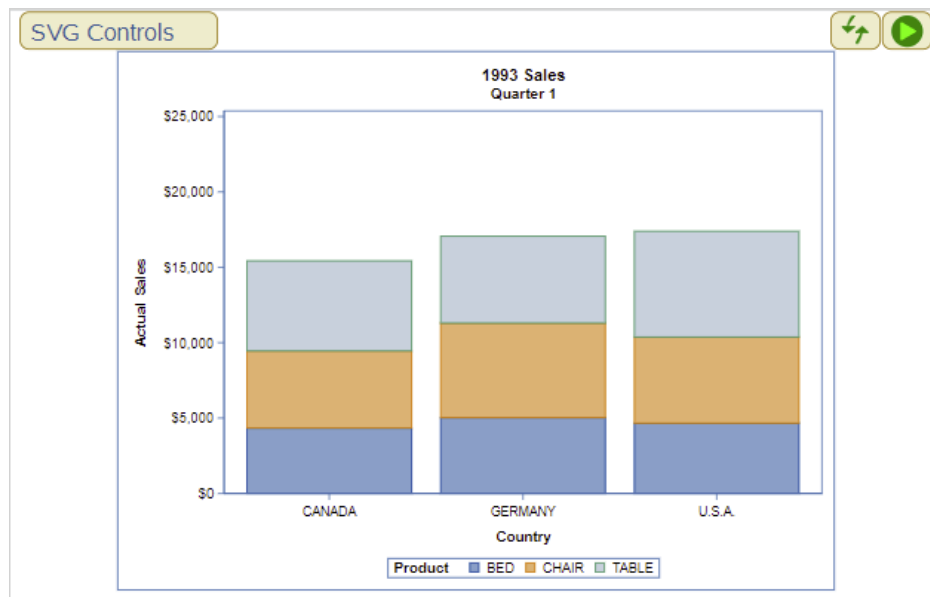
When the **PRINTERPATH=** option is set to **SVG**, you can use the **SVG** animation options to configure the fade-in and fade-out attributes and the **autoplay** attribute. The animation options that begin with **SVG** do not affect **GIF** images.

After you set the options and opened the **PRINTER** destination, proceed with your **SAS** code to create each frame in your file. The animation frame is created when you run **SAS** procedures. You can use animation options in between procedures to change the duration that a frame is held in view and the fade-in and fade-out times. For example, you can hold a particular frame in view for a longer period of time. You would use the **OPTIONS ANIMDURATION=** statement before a procedure to increase the time that the frame is held in view. Specify **ANIMATION=STOP** to end the creation of the animation file. Use the **ODS PRINTER CLOSE** statement to close the file.

TIP Be sure to specify **ANIMATION=STOP** after you create the frames for your animation file. If **ANIMATION=START** remains set, you might create an animation file unintentionally for subsequent procedure statements.

To embed the file in a web page or to create a link to the file from a web page, see “[SVG Documents in HTML Files](#)” in [SAS Language Reference: Concepts](#).

When an animated file is displayed in a browser, the animation control buttons can be used to reset () the animation, to pause () the animation, and to play () the animation. You can toggle **SVG Controls** to show or hide the control buttons. Here is one frame of an animated **SVG** document with the control buttons:



If you are creating SVG files that will be viewed on an iPad, a best practice is to use the SVGVIEW Universal Printer for optimal sizing.

You can view animated SVG files using Internet Explorer 9 or later. You can view animated GIF files in releases prior to Internet Explorer 9.

Animation System Options

You use the animation system options to configure the attributes of an animation GIF or SVG file. All of the system options can be configured for SVG documents. The options that begin with SVG are not applicable for GIF images. Here are the animation system options.

Table 5.4 Animation System Options and Valid Universal Printers

Description	Option Name	Valid Universal Printers
Starts or stops the creation of an animation file.	ANIMATION on page 83	GIF and SVG
Specifies the amount of time that each frame in an animation is held in view.	ANIMDURATION= on page 84	GIF and SVG
Specifies whether the animation loop is played continuously or is played once, or specifies a specific number of times that the animation loop is repeated.	ANIMLOOP= on page 85	GIF and SVG

Description	Option Name	Valid Universal Printers
ANIMLOOP=YES sets a continuous loop. ANIMLOOP=NO completes one loop. ANIMLOOP=n specifies a specific number of times to loop. For SVG documents, use ANIMLOOP=YES and ANIMLOOP=NO. Setting ANIMLOOP=n where $n > 0$ for SVG documents completes only one loop for the document.		
Specifies whether animation frames are overlaid or if they are played sequentially. If you overlay frames, your frames would require some level of transparency for the output not to appear overwritten.	ANIMOVERLAY on page 86	GIF and SVG
Specifies whether an SVG animation starts immediately in the web browser. If you specify NOSVGautoplay, start the animation by clicking .	SVGautoplay	SVG
Specifies the number of seconds for an SVG frame to fade into view.	SVGFADEIN=	SVG
Specifies whether an SVG frame overlaps the previous frame or if each frame is played sequentially when a frame is fading in and out.	SVGFADEMODE=	SVG
Specifies the number of seconds for an SVG frame to fade out of view.	SVGFADEOUT=	SVG

Example: Creating an Animated SVG Image

This example shows how to create a simple animated SVG graph of sales data in data set SASHELP.SALES. The animation shows quarterly sales of select products in 1993 and 1994 in 8 graphs. The animation loops through the graphs continuously

starting with 1993 Q1. Each graph is displayed for 3 seconds. After 3 seconds, the current graph fades out and the next graph fades in. Here is the code.

```
proc sort data=sashelp.prdsale out=work.sales; /* 1 */
    by year quarter;
run;

ods _all_ close; /* 2 */

ods graphics / reset border; /* 3 */
options printerpath=svg
    nobyline
    animduration=3
    svgfadein=1 svgfadeout=1
    noanimoverlay
    nodate nonumber
    papersize=("6in","4.5in")
    svgwidth="6in" svgheight="4.5in";

options animation=start; /* 4 */

ods printer style=plateau file="svganim.svg"; /* 5 */

title1 "#byval(year) Sales" lspace=1;
title2 "Quarter #byval(quarter)";
proc sgplot data=work.sales;
    vbar country / response=actual group=product stat=sum;
    yaxis values=(0 to 25000 by 5000);
    where product in ("BED" "TABLE" "CHAIR");
    by year quarter;
    format actual dollar8.0;
run;

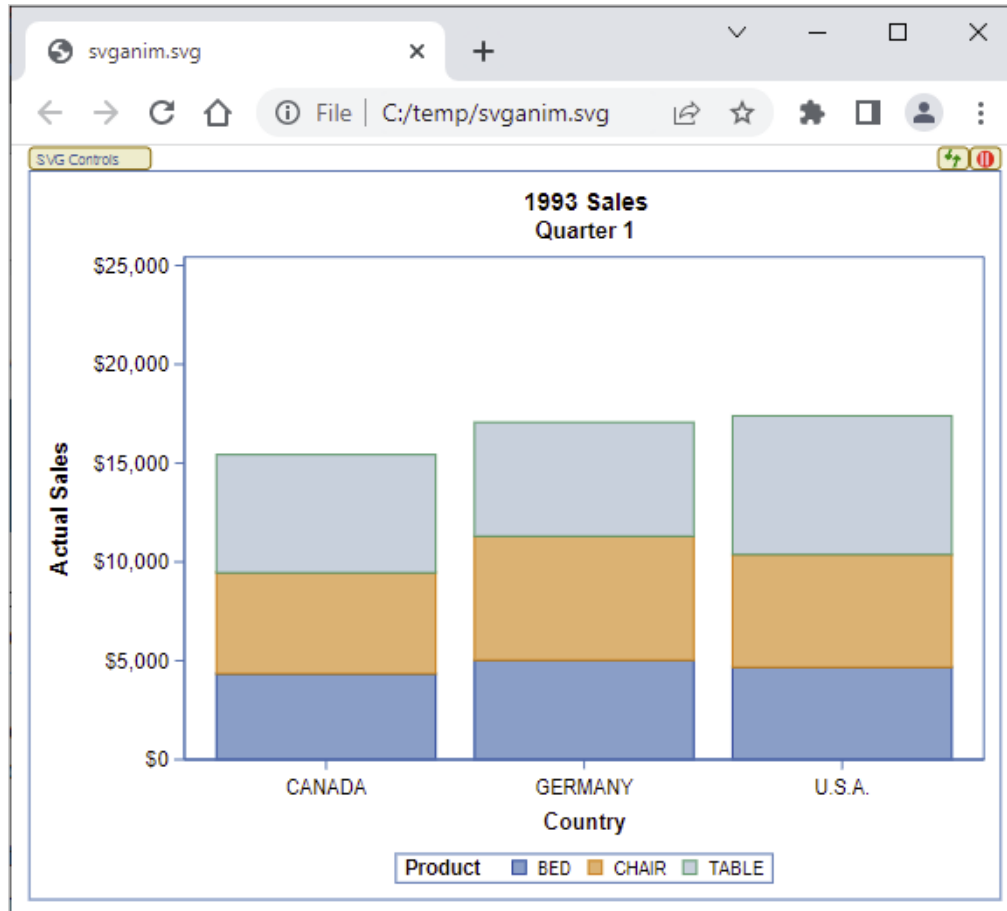
ods printer close;
options animation=stop; /* 6 */

options byline date number animduration=min /* 7 */
    papersize=("","") svgheight="" svgwidth=""
    byline;

ods html; /* 8 */
```

- 1 Sort the data in SASHELP.PRDSALE by variables Year and Quarter, and store the result in data set WORK.SALES.
- 2 Close all the currently open ODS destinations.
- 3 Specify the SVG printer, the animation parameters, and the output size.
- 4 Start the animation.
- 5 Generate the graphs.
- 6 Stop the animation.
- 7 Reset the system options.
- 8 Open the HTML destination.

When the SVG output file is opened in a browser, the first graph is displayed, and the animation begins. Here is the first graph.



Dictionary of Printing-Related System Options

<i>About SAS System Options</i>	80
<i>Summary of System Options that Control Universal Printing</i>	81
<i>Dictionary</i>	83
ANIMATION= System Option	83
ANIMDURATION= System Option	84
ANIMLOOP= System Option	85
ANIMOVERLAY System Option	86
BOTTOMMARGIN= System Option	87
COLOPHON= System Option	89
COLORPRINTING System Option	92
FONTEMBEDDING System Option	93
LEFTMARGIN= System Option	94
ORIENTATION= System Option	96
PAPERSIZE= System Option	98
PDFACCESS System Option	100
PDFASSEMBLY System Option	102
PDFCOMMENT System Option	103
PDFCONTENT System Option	105
PDFCOPY System Option	106
PDFFILLIN System Option	107
PDFPAGELAYOUT= System Option	109
PDFPAGEVIEW= System Option	110
PDFPASSWORD= System Option	111
PDFPRINT= System Option	113
PDFSECURITY= System Option	115
PRINTERPATH= System Option	117
RIGHTMARGIN= System Option	118
TOPMARGIN= System Option	120
UPRINTCOMPRESSION System Option	121

About SAS System Options

System options are instructions that affect the processing of an entire SAS program or SAS session from the time the option is specified until it is changed. For example, there are SAS system options that control the appearance of SAS output or how observations are handled in SAS data sets. Some system options affect only a particular SAS component or product. The documentation for that component or product provides the details for those system options. This document provides information about the system options that affect printed output. For details of all other system options and how to use them, see [“Definition of System Options” in SAS System Options: Reference](#).

There is a system option that embeds fonts in your output for intentional control of displaying or printing that output. See [“The FONTEMBEDDING System Option” on page 22](#) for details.

SAS provides a procedure, a statement, and a function that help discover or control the value of a SAS system option.

OPTIONS procedure

PROC OPTIONS processes a SAS system option. For details about the SAS procedure that discovers SAS system option settings, see [“OPTIONS Procedure” in SAS System Options: Reference](#).

PROC OPTIONS statement

The SAS OPTIONS statement specifies or changes the value of one or more SAS system options. This global change remains in effect for the rest of the job, session, SAS process, or until another SAS OPTIONS statement is issued. For details, see [“OPTIONS Statement” in SAS Global Statements: Reference](#).

SAS GETOPTION function

Retrieves the value of a SAS system option. For more information, see [“GETOPTION Function” in SAS System Options: Reference](#).

See Also

For links to the component and product documents that describe system options affecting them, see [“SAS System Options Documented in Other SAS Publications” in SAS System Options: Reference](#).

Summary of System Options that Control Universal Printing

The following table lists system options that control Universal Printing.

Table 6.1 *System Options That Control Universal Printing*

System Option	Description
ANIMATION= (p. 83)	Specifies whether to start or stop creating animation frames.
ANIMDURATION= (p. 84)	Specifies the amount of time that each frame in the animation is held in view.
ANIMLOOP= (p. 85)	Specifies whether the animation loop is played continuously or is played once, or specifies a specific number of times that the animation loop is repeated.
ANIMOVERLAY (p. 86)	Specifies whether animation frames are overlaid or whether they are played sequentially.
BOTTOMMARGIN= (p. 87)	Specifies the size of the margin at the bottom of the page for printing.
COLOPHON= (p. 89)	Specifies a text string that is embedded in a graphic file or a PDF that is created by Universal Printers. The text string is not displayed in the rendered graphic or PDF.
COLORPRINTING (p. 92)	Specifies color printing, if it is supported.
FONTEMBEDDING (p. 93)	Specifies whether font embedding is enabled in Universal Printer and SAS/GRAPH printing.
LEFTMARGIN= (p. 94)	Specifies the size of the margin on the left side of the page.
ORIENTATION= (p. 96)	Specifies the paper orientation to use (either portrait, landscape, reverse-portrait, or reverse-landscape) for the whole document or for changing the orientation of individual pages in a document.

System Option	Description
PAPERSIZE= (p. 98)	Specifies the paper size to use when printing.
PDFACCESS (p. 100)	Specifies whether PDF documents can be edited.
PDFACCESS (p. 100)	Specifies whether PDF documents can be edited.
PDFASSEMBLY (p. 102)	Specifies whether PDF documents can be assembled.
PDFCOMMENT (p. 103)	Specifies whether PDF document comments can be modified.
PDFCONTENT (p. 105)	Specifies whether the contents of a PDF document can be changed.
PDFCOPY (p. 106)	Specifies whether text and graphics from a PDF document can be copied.
PDFFILLIN (p. 107)	Specifies whether PDF forms can be filled in.
PDFPAGELAYOUT= (p. 109)	Specifies the page layout for PDF documents.
PDFPAGEVIEW= (p. 110)	Specifies the page viewing mode for PDF documents.
PDFPASSWORD= (p. 111)	Specifies the password to use to open a PDF document and the password used by a PDF document owner to set document permissions and restrictions.
PDFPRINT= (p. 113)	Specifies the resolution to print PDF documents.
PDFSECURITY= (p. 115)	Specifies the level of encryption for PDF documents.
PRINTERPATH= (p. 117)	Specifies a printer for Universal Printing print jobs.
RIGHTMARGIN= (p. 118)	Specifies the size of the margin on the right side of the page.
TOPMARGIN= (p. 120)	Specifies the size of the margin at the top of the page.
UPRINTCOMPRESSION (p. 121)	Specifies whether to enable the compression of files created by some Universal Printer and SAS/GRAPH devices.

Dictionary

ANIMATION= System Option

Specifies whether to start or stop creating animation frames.

Valid in:	Autoexec file, OPTIONS statement,
Categories:	Log and Procedure Output Control: SVG Log and Procedure Output Control: Animation
PROC OPTIONS GROUP=	SVG Animation
Alias:	ANIMATE
Default:	The shipped default is STOP.
Notes:	This option can be restricted by a site administrator. For more information, see “Restricted Options” in SAS System Options: Reference . The option is valid for GIF and SVG animation.

Syntax

ANIMATION=[START](#) | [STOP](#)

Syntax Description

START

specifies to begin creating animation frames.

STOP

specifies to stop creating animation frames.

Tip Specify ANIMATION=STOP after you have created the frames for your animation file. If ANIMATION=START remains specified, you might unintentionally create an animation file for subsequent procedure statements.

See Also

- [“About Animated GIF Images and SVG Documents” on page 72](#)

System Options:

- “ANIMDURATION= System Option” on page 84
- “ANIMLOOP= System Option” on page 85
- “ANIMOVERLAY System Option” on page 86
- “SVGAUTOPLAY System Option” in *SAS System Options: Reference*
- “SVGFADEIN= System Option” in *SAS System Options: Reference*
- “SVGFADEMODE= System Option” in *SAS System Options: Reference*
- “SVGFADEOUT= System Option” in *SAS System Options: Reference*

ANIMDURATION= System Option

Specifies the amount of time that each frame in the animation is held in view.

Valid in:	Autoexec file, OPTIONS statement,
Categories:	Log and Procedure Output Control: SVG Log and Procedure Output Control: Animation
PROC OPTIONS GROUP=	SVG Animation
Default:	The shipped default is MIN.
Notes:	This option can be restricted by a site administrator. For more information, see “Restricted Options” in SAS System Options: Reference . The option is valid for GIF and SVG animation.

Syntax

ANIMDURATION=*duration* | MIN

Syntax Description

duration

specifies the number of seconds to hold an animation frame in view. The number can be any numeric representation (for example, .01, 5, 6.5).

Defaults 0 for GIF frames

0.1 for SVG frames

MIN

specifies to use the device driver’s default value for the amount of time that each frame is held in view.

See Also

- [“Creating Animated GIF Images and SVG Documents” on page 72](#)

System Options:

- [“ANIMATION= System Option” on page 83](#)
- [“ANIMLOOP= System Option” on page 85](#)
- [“ANIMOVERLAY System Option” on page 86](#)
- [“SVGAUTOPLAY System Option” in *SAS System Options: Reference*](#)
- [“SVGFADEIN= System Option” in *SAS System Options: Reference*](#)
- [“SVGFADEMODE= System Option” in *SAS System Options: Reference*](#)
- [“SVGFADEOUT= System Option” in *SAS System Options: Reference*](#)

ANIMLOOP= System Option

Specifies whether the animation loop is played continuously or is played once, or specifies a specific number of times that the animation loop is repeated.

Valid in: Autoexec file, OPTIONS statement,

Categories: Log and Procedure Output Control: SVG
Log and Procedure Output Control: Animation

PROC OPTIONS GROUP= SVG
Animation

Default: The shipped default is YES.

Notes: This option can be restricted by a site administrator. For more information, see [“Restricted Options” in *SAS System Options: Reference*](#).
The option is valid for GIF and SVG animation.

Syntax

ANIMLOOP=[YES](#) | [NO](#) | *n*

Syntax Description

YES

specifies that animation loops continuously repeat.

NO

specifies that animation loops complete one loop.

n

specifies the number of times that an animation loop is continuously repeated. The number must be an integer.

Range 0–2147483647

Interaction For SVG documents only, the value 0 enables the animation to loop continuously. When *n* is any value other than 0, the pages of an SVG animation loop *n* times. Instead of using ANIMLOOP=*n*, you can use ANIMLOOP=YES and ANIMLOOP=NO.

Details

Use the NOANIMOVERLAY option for animations that loop continuously so that each page is replaced by the next frame.

See Also

- [“Creating Animated GIF Images and SVG Documents” on page 72](#)

System Options:

- [“ANIMATION= System Option” on page 83](#)
- [“ANIMDURATION= System Option” on page 84](#)
- [“ANIMOVERLAY System Option” on page 86](#)
- [“SVGautoplay System Option” in SAS System Options: Reference](#)
- [“SVGFADEIN= System Option” in SAS System Options: Reference](#)
- [“SVGFADEMODE= System Option” in SAS System Options: Reference](#)
- [“SVGFADEOUT= System Option” in SAS System Options: Reference](#)

ANIMOVERLAY System Option

Specifies whether animation frames are overlaid or whether they are played sequentially.

Valid in:	Autoexec file, OPTIONS statement,
Categories:	Log and Procedure Output Control: Animation Log and Procedure Output Control: SVG
PROC OPTIONS GROUP=	ANIMATION SVG
Default:	The shipped default is ANIMOVERLAY.

Notes:

This option can be restricted by a site administrator. For more information, see [“Restricted Options” in SAS System Options: Reference](#).

The option is valid for GIF and SVG animation.

Syntax

ANIMOVERLAY

NOANIMOVERLAY

Syntax Description

ANIMOVERLAY

specifies that each frame in an animation overlays the previous frame. All frames remain in view for the duration of the animation.

NOANIMOVERLAY

specifies that each frame is played sequentially. When a frame is played, it replaces the previous frame.

See Also

- [“Creating Animated GIF Images and SVG Documents” on page 72](#)

System Options:

- [“ANIMATION= System Option” on page 83](#)
- [“ANIMDURATION= System Option” on page 84](#)
- [“ANIMLOOP= System Option” on page 85](#)
- [“SVGautoplay System Option” in SAS System Options: Reference](#)
- [“SVGFADEIN= System Option” in SAS System Options: Reference](#)
- [“SVGFADEMODE= System Option” in SAS System Options: Reference](#)
- [“SVGFADEOUT= System Option” in SAS System Options: Reference](#)

BOTTOMMARGIN= System Option

Specifies the size of the margin at the bottom of a printed page.

Valid in: Autoexec file, OPTIONS statement,

Category: Log and Procedure Output Control: ODS Printing

PROC OPTIONS ODSPRINT

GROUP=

Default: The shipped default is 0.000 in.

Note: This option cannot be restricted by a site administrator. For more information, see [“Restricted Options” in SAS System Options: Reference](#).

Syntax

BOTTOMMARGIN=*margin-size*<*margin-unit*>

Syntax Description

margin-size

specifies the size of the margin.

Restriction The bottom margin should be small enough so that the top margin plus the bottom margin is less than the height of the paper.

Interaction Changing the value of this option might result in changes to the value of the PAGESIZE= system option.

<*margin-unit*>

specifies the units for margin-size. The margin-unit can be *in* for inches or *cm* for centimeters. <*margin-unit*> is saved as part of the value of the BOTTOMMARGIN system option.

Default inches

Details

All margins have a minimum that is dependent on the printer and the paper size.

Example

```
options bottommargin=10cm;
```

See Also

- Chapter 2, “Setting Up and Managing Universal Printing,” on page 11
- “Understanding ODS Destinations” in *SAS Output Delivery System: User’s Guide*

System Options:

- “LEFTMARGIN= System Option” on page 94
- “RIGHTMARGIN= System Option” on page 118

■ “TOPMARGIN= System Option” on page 120

COLOPHON= System Option

Specifies a text string that is embedded in a graphic file or a PDF that is created by Universal Printers. The text string is not displayed in the rendered graphic or PDF.

Valid in:	Autoexec file, OPTIONS statement,
Category:	Log and Procedure Output Control: ODS Printing
PROC OPTIONS GROUP=	ODSPRINT
Note:	This option can be restricted by a site administrator. For more information, see “Restricted Options” in SAS System Options: Reference .

Syntax

COLOPHON=*“text-string”*

Syntax Description

text-string

specifies the text that is embedded in a graphic file.

Length up to 4,000 characters.

Requirement The text string must be enclosed in quotation marks.

Interaction Text strings that are longer than the value of the LRECL= option are truncated. The default value of the LRECL= option is 32767. Temporarily increase the value of the LRECL= option for longer text strings.

Details

A colophon is a printer's mark. You can use the COLOPHON= option to add a signature, an identification, or a comment to Universal Printer output files. The text string is not visible in the graphic or PDF when the output file is displayed or printed. You can view the colophon text string by using a text editor or a third-party application.

Example

This example adds the text "Simple Text String" to the output file that is created by the various Universal Printers:

```
ods html
close;

ods listing
close;

%macro
  ctext(printer,file,ext);

  filename sasprt "c:\temp
\&file..&ext";

  options printerpath=("&printer"
sasprt)

  colophon="Simple Text
String";

  ods
printer;

  title
"&printer";

  proc print
data=sashelp.class;

run;

  ods printer
close;

%mend;

%ctext(PCL5c,
coloph1,pcl);

%ctext(GIF,
coloph1,gif);

%ctext(Postscript,
coloph1,psl);

%ctext(PDF,
coloph1,pdf);
```

```
%ctext (PNG,
coloph1,png);

%ctext (SVG,
coloph1,svg);

%ctext (EMF,
coloph1,emf);

%ctext (TIFF,coloph1.tif);

ods listing;
```

Using Notepad, you can see the text string “Simple Text String” in the SVG output file.

```
<svg xmlns="http://www.w3.org/2000/svg" xmlns
<!-- Simple Text String -->
<desc></desc>
<defs>
<clipPath id="SVGMain_clipPage1">
<rect x="-1" y="-1" width="801" height="601"
</clipPath>
</defs>
<g id="SVGMain_Page1" clip-path="url(#SVGmai
<rect x="0" y="0" width="800" height="600" s
stroke-linejoin: round; "/>
<rect x="262" y="39" width="37" height="25"
<rect x="300" y="39" width="51" height="25"
<rect x="352" y="39" width="34" height="25"
<rect x="387" y="39" width="37" height="25"
<rect x="425" y="39" width="55" height="25"
<rect x="481" y="39" width="57" height="25"
<rect x="262" y="65" width="37" height="25"
<rect x="262" y="91" width="37" height="25"
<rect x="262" y="117" width="37" height="25"
<rect x="262" y="143" width="37" height="25"
<rect x="262" y="169" width="37" height="25"
```

The text string is not visible in the output file:

SVG

Obs	Name	Sex	Age	Height	Weight
1	Alfred	M	14	69.0	112.5
2	Alice	F	13	56.5	84.0
3	Barbara	F	13	65.3	98.0
4	Carol	F	14	62.8	102.5
5	Henry	M	14	63.5	102.5

COLORPRINTING System Option

Specifies whether to print in color if color printing is supported.

Valid in: Autoexec file, OPTIONS statement,

Category: Log and Procedure Output Control: ODS Printing

PROC OPTIONS
GROUP= ODSPRINT

Default: The shipped default is COLORPRINTING.

Note: This option can be restricted by a site administrator. For more information, see [“Restricted Options” in SAS System Options: Reference](#).

Syntax

COLORPRINTING

NOCOLORPRINTING

Syntax Description

COLORPRINTING

specifies to attempt to print in color.

NOCOLORPRINTING

specifies not to print in color.

Details

Most SAS system options are initialized with default settings when SAS is invoked. However, the default settings and option values for some SAS system options might vary both by operating environment and by site. For details, see the SAS documentation for your operating environment.

See Also

[Chapter 2, “Setting Up and Managing Universal Printing,” on page 11](#)

FONTEMBEDDING System Option

Specifies whether font embedding is enabled in Universal Printer and SAS/GRAPH printing.

Valid in:	Autoexec file, OPTIONS statement,
Category:	Log and Procedure Output Control: ODS Printing
PROC OPTIONS GROUP=	ODSPRINT
Default:	The shipped default is FONTEMBEDDING.
Note:	This option can be restricted by a site administrator. For more information, see “Restricted Options” in SAS System Options: Reference .

Syntax

FONTEMBEDDING

NOFONTEMBEDDING

Syntax Description

FONTEMBEDDING

specifies to enable font embedding.

NOFONTEMBEDDING

specifies to disable font embedding.

Details

Font embedding is used mainly by Universal Printing. Not all printers support font embedding. To determine whether the printer that you are using supports font embedding, use the QDEVICE procedure. If **Font Embedding** is listed in the SAS log with a value of **Option** or **Always**, the printer supports font embedding. Here is a partial log output from the QDEVICE procedure:

```
369 proc qdevice report=general;
370     printer pdf;
371 run;
```

```

      Name: PDF
Description: Portable Document Format Version 1.4
      Module: SASPDPDF
      Type: Universal Printer
      Registry: SASHELP
      Prototype: PDF Version 1.4
Default Typeface: Cumberland AMT
```

```

Typeface Alias: Courier
  Font Style: Regular
  Font Weight: Normal
  Font Height: 8 points
  Font Version: Version 1.03
Maximum Colors: 16777216
  Visual Color: Direct Color
  Color Support: RGBA
  Destination: sasprt.pdf
  I/O Type: DISK
  Data Format: PDF

...more registry settings...
Compression Method: FLATE
  Font Embedding: Option
  Animation: Unsupported

```

When FONTEMBEDDING is set, fonts can be embedded, or included, in the output files that are created by the Universal Printer and SAS/GRAPH. Output files with embedded fonts do not rely on fonts being installed on the computer that is used to view or print the output file. File size is increased for vector output for printers such as PDF and PostScript.

When NOFONTEMBEDDING is set, the output files rely on the fonts being installed on the computer that is used to view or print the font. If a font is not found on the computer, the printer or the application that displays the output might perform font substitution. Image output is not affected when NOFONTEMBEDDING is set.

To determine which fonts are substituted for a given printer, use the Print Setup window or the QDEVICE procedure to display the Printer Setup properties. Under **Fonts**, any individual fonts that are listed are recognized by the printer. All other fonts, including those that are available via a link in the SAS Registry, are substituted in the document when the document is created.

See Also

- [SAS/GRAPH: Reference](#)
- [“Universal Printing” in SAS Language Reference: Concepts](#)

LEFTMARGIN= System Option

Specifies the print margin for the left side of the page.

Valid in:	Autoexec file, OPTIONS statement,
Category:	Log and Procedure Output Control: ODS Printing
PROC OPTIONS	ODSPRINT
GROUP=	

Default: The shipped default is 0.000 in.

Note: This option cannot be restricted by a site administrator. For more information, see [“Restricted Options” in SAS System Options: Reference](#).

Syntax

LEFTMARGIN=*margin-size*<*margin-unit*>

Syntax Description

margin-size

specifies the size of the left print margin.

Restriction The left margin should be small enough so that the left margin plus the right margin is less than the width of the paper.

Interaction Changing the value of this option might result in changes to the value of the LINESIZE= system option.

<*margin-unit*>

specifies the units for margin-size. The margin-unit can be *in* for inches or *cm* for centimeters. <*margin-unit*> is saved as part of the value of the LEFTMARGIN system option whether it is specified.

Default inches

Details

All margins have a minimum that is dependent on the printer and the paper size.

See Also

- [Chapter 2, “Setting Up and Managing Universal Printing,” on page 11](#)
- [“Understanding ODS Destinations” in SAS Output Delivery System: User’s Guide](#)

System Options:

- [“BOTTOMMARGIN= System Option” on page 87](#)
- [“LINESIZE= System Option” in SAS System Options: Reference](#)
- [“RIGHTMARGIN= System Option” on page 118](#)
- [“TOPMARGIN= System Option” on page 120](#)

ORIENTATION= System Option

Specifies the paper orientation to use when printing to a printer.

Valid in:	Autoexec file, OPTIONS statement,
Category:	Log and Procedure Output Control: ODS Printing
PROC OPTIONS GROUP=	ODSPRINT
Default:	The shipped default is PORTRAIT.
Requirement:	The ORIENTATION= option is valid only for paper sizes that are defined in the SAS Registry. The option is ignored for custom paper sizes.
Note:	This option cannot be restricted by a site administrator. For more information, see “Restricted Options” in SAS System Options: Reference .
Tip:	If the orientation of a PDF document is changed after the PDF destination is opened and before the PDF destination is closed, any setting for margins is taken from the OPTIONS statement in place before the ODS PDF FILE= statement. If no OPTIONS statement is used to explicitly set the margins, the margin settings are retrieved from the SAS registry, using PROC REGISTRY. For more info, see “REGISTRY Procedure” in Base SAS Procedures Guide .

Syntax

ORIENTATION=[PORTRAIT](#) | [LANDSCAPE](#) | [REVERSEPORTRAIT](#) | [REVERSELANDSCAPE](#)

Syntax Description

PORTRAIT

specifies the paper orientation as portrait.

LANDSCAPE

specifies the paper orientation as landscape.

REVERSEPORTRAIT

specifies the paper orientation as reverse portrait. Use this value to control the top of the page relative to how the paper is inserted into the input paper tray. REVERSEPORTRAIT can be used when you print preprinted or punched forms.

REVERSELANDSCAPE

specifies the paper orientation as reverse landscape. Use this value to control the top of the page relative to how the paper is inserted into the input paper tray. REVERSELANDSCAPE can be used when you print preprinted or punched forms.

Details

Changing the value of this option might result in changes to the values of the portable `LINESIZE=` and `PAGESIZE=` system options.

You can change the orientation between document pages for the following output types:

- the WORD destination
- the RTF destination
- a Universal Printing printer

Note: Changing the orientation between document pages is supported only for Universal Printing. It is not supported for Windows printing.

Use the `OPTIONS` statement between the steps that create output to change the page orientation.

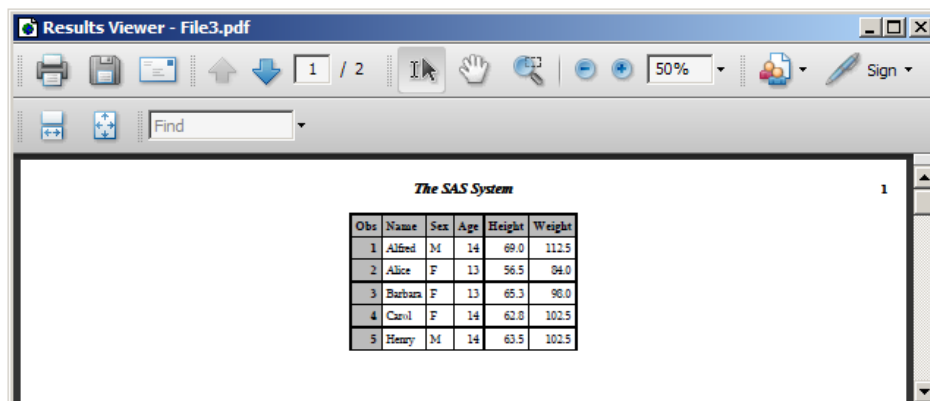
Example

This example creates a PDF file with both portrait and landscape orientations.

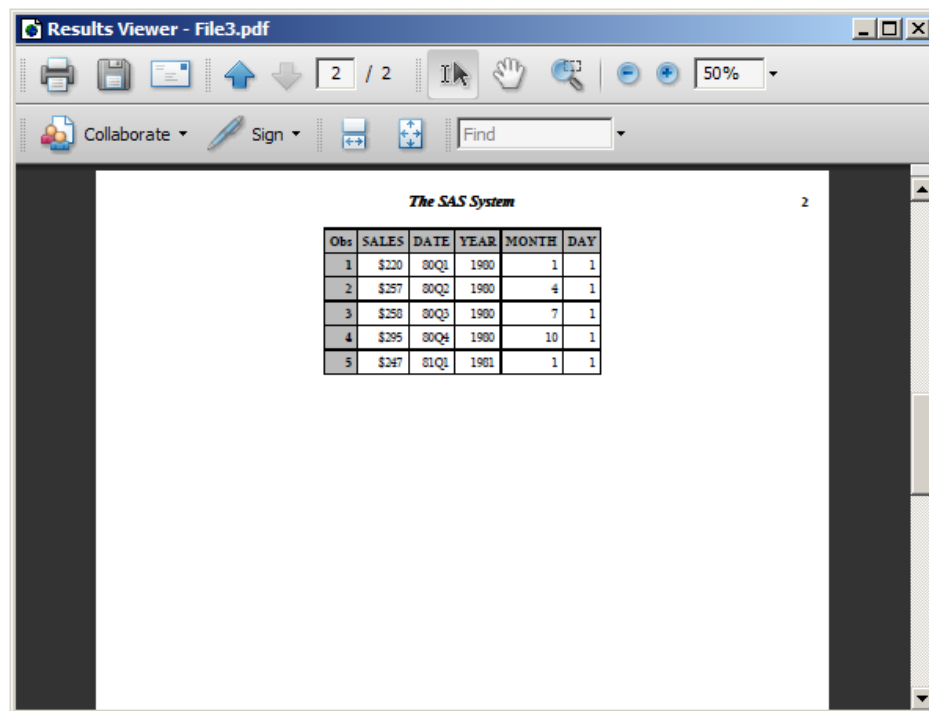
```
options orientation=landscape obs=5;
ods pdf file="File3.pdf";
proc print data=sashelp.class;
run;
options orientation=portrait;
proc print data=sashelp.retail; run;
ods pdf close;
```

Here is the output:

Output 6.1 *The First Page of the PDF Has a Landscape Orientation*



Obs	Name	Sex	Age	Height	Weight
1	Alfred	M	14	69.0	112.5
2	Alice	F	13	56.5	84.0
3	Barbara	F	13	65.3	90.0
4	Carol	F	14	62.0	102.5
5	Henry	M	14	63.5	102.5

Output 6.2 The Second Page of the PDF Has a Portrait Orientation

See Also

- [“Universal Printing” in SAS Language Reference: Concepts](#)

System Options:

- [“LINESIZE= System Option” in SAS System Options: Reference](#)
- [“PAGESIZE= System Option” in SAS System Options: Reference](#)

PAPERSIZE= System Option

Specifies the paper size to use for printing.

Valid in: Autoexec file, OPTIONS statement,

Categories: Environment Control: Language Control
Log and Procedure Output Control: ODS Printing

PROC OPTIONS
GROUP= LANGUAGECONTROL
ODSPRINT

Default: The default is LETTER or A4, depending on the locale.

Note: This option cannot be restricted by a site administrator. For more information, see [“Restricted Options” in SAS System Options: Reference](#).

Syntax

PAPERSIZE=LOCALE | *paper_size_name*
 | (*width_value*<,>*height_value*)
 | ('*width_value*'<,>'*height_value*')
 | ("*width_value*"<,>"*height_value*")

Syntax Description

LOCALE

specifies to use the value of the LOCALE= system option to determine the value of the PAPERSIZE= option. Depending on the locale, the PAPERSIZE= option is set to either LETTER or A4.

See [“LOCALE System Option” in SAS National Language Support \(NLS\): Reference Guide](#)

paper_size_name

specifies a predefined paper size.

Default	LETTER or A4, depending on the locale
Restriction	The maximum length is 200 characters.
Requirements	When the name of a predefined paper size contains spaces, enclose the name in single or double quotation marks. A space is required between the width and the height values if you do not use either single or double quotation marks for values.
Tip	For more information, use the Registry Editor, or use PROC REGISTRY to obtain a list of supported paper sizes. Additional values can be added.

("width-value", "height-value")

specifies paper width and height as positive floating-point values.

Default	inches
Range	<i>in</i> or <i>cm</i> for <i>width_value</i> , <i>height_value</i>
Requirement	If you specify <i>width-value</i> and <i>height-value</i> with spaces between the number and the unit, the values must be enclosed in quotation marks (for example, "5 in" "7 in").
Interaction	When you specify a custom paper size, the ORIENTATION= system option is ignored and the orientation is determined by the width and height values. The orientation is portrait if the paper height is more than the width. The orientation is landscape if the paper width is more than the height. Reverse orientations are not supported for custom paper sizes.

Details

If you specify a predefined paper size or a custom size that is not supported by your printer, the printer default paper size is used. The printer default paper size is locale dependent and can be changed using the Page Setup dialog box.

Fields that specify values for paper sizes can either be separated by blanks or commas.

Note: Changing the value of this option can result in changes to the values of the portable `LINESIZE=` and `PAGESIZE=` system options.

Comparisons

The first `OPTIONS` statement sets a paper size value that is a paper size name from the SAS Registry. The second `OPTIONS` statement sets a specific width and height for a paper size.

```
options papersize="480x640 Pixels";  
options papersize=(4.5 7);
```

In the first example, quotation marks are required because there is a space in the name.

In the second example, quotation marks are not required. When no measurement units are specified, SAS writes the following warning to the SAS log:

```
WARNING: Units were not specified on the PAPERSIZE option. Inches will be used.  
WARNING: Units were not specified on the PAPERSIZE option. Inches will be used.
```

You can avoid the warning message by adding the unit type, `in` or `cm`, to the value with no space separating the value and the unit type:

```
options papersize=(4.5in 7in);
```

See Also

- [“Universal Printing” in SAS Language Reference: Concepts](#)

System Options:

- [“ORIENTATION= System Option” on page 96](#)

PDFACCESS System Option

Specifies whether PDF documents can be edited.

Valid in:	Autoexec file, OPTIONS statement,
Category:	Log and Procedure Output Control: PDF
PROC OPTIONS GROUP=	PDF
Default:	The shipped default is PDFACCESS.
Requirement:	Adobe Acrobat Reader or Professional 5.0 and later versions
Notes:	Document properties might differ between the Documents Security window, the Show Details window, and SAS PDF option values for secure documents. The values might differ if a document password has only an Open password, or if the Open and Owner passwords are the same value. These windows display property values correctly when both Open and Owner have passwords, the passwords are not the same, and the document is opened with the Open password. This option can be restricted by a site administrator. For more information, see “Restricted Options” in SAS System Options: Reference.

Syntax

PDFACCESS

NOPDFACCESS

Syntax Description

PDFACCESS

specifies that PDF documents can be edited.

NOPDFACCESS

specifies that PDF documents cannot be edited.

Details

The PDFACCESS option can affect the Document Editing Enabled document property.

The document property values for PDF security are not changed when you set PDFSECURITY=NONE. The results are the same as not specifying the option at all.

The following table shows how the Document Editing Enabled document property is set when you specify the PDFACCESS option and set the PDFSECURITY= option to HIGH:

	PDFACCESS PDFSECURITY=HIGH	NOPDFACCESS PDFSECURITY=HIGH
Document Editing Enabled	Allowed	Allowed

See Also

- [“Securing ODS-Generated PDF Files” in SAS Output Delivery System: User’s Guide](#)

System Options:

- [“PDFSECURITY= System Option” on page 115](#)

PDFASSEMBLY System Option

Specifies whether PDF documents can be assembled.

Valid in: Autoexec file, OPTIONS statement,

Category: Log and Procedure Output Control: PDF

PROC OPTIONS
GROUP= PDF

Default: The shipped default is NOPDFASSEMBLY.

Requirement: Adobe Acrobat Reader or Professional 5.0 and later versions

Notes: Document properties might differ between the Documents Security window, the Show Details window, and SAS PDF option values for secure documents. The values might differ if a document password has only an Open password, or if the Open and Owner passwords are the same value. These windows display property values correctly when both Open and Owner have passwords, the passwords are not the same, and the document is opened with the Open password.

This option can be restricted by a site administrator. For more information, see [“Restricted Options” in SAS System Options: Reference](#).

Syntax

PDFASSEMBLY

NOPDFASSEMBLY

Syntax Description

PDFASSEMBLY

specifies that PDF documents can be assembled.

NOPDFASSEMBLY

specifies that PDF documents cannot be assembled.

Details

When a PDF document is assembled, pages can be rotated, inserted, and deleted, and bookmarks and thumbnail images can be added.

The PDFASSEMBLY option can affect the Document Assembly document property.

The document property values for PDF security are not changed when you set PDFSECURITY=NONE. The results are the same as not specifying the option at all.

The following table shows how the Document Assembly document property is set when you specify the PDFASSEMBLY option and set the PDFSECURITY= option to HIGH:

	PDFASSEMBLY PDFSECURITY=HIGH	NOPDFASSEMBLY PDFSECURITY=HIGH
Document Assembly	Allowed	Allowed

See Also

- [“Securing ODS-Generated PDF Files” in SAS Output Delivery System: User's Guide](#)

System Options:

- [“PDFSECURITY= System Option” on page 115](#)

PDFCOMMENT System Option

Specifies whether PDF document comments can be modified.

Valid in: Autoexec file, OPTIONS statement,

Category: Log and Procedure Output Control: PDF

PROC OPTIONS
GROUP= PDF

Default: The shipped default is NOPDFCOMMENT.

Requirement: Adobe Acrobat Reader or Professional 5.0 and later versions

Notes: Document properties might differ between the Documents Security window, the Show Details window, and SAS PDF option values for secure documents. The values might differ if a document password has only an Open password, or if the Open and Owner passwords are the same value. These windows display property values correctly when both Open and Owner have passwords, the passwords are not the same, and the document is opened with the Open password.

This option can be restricted by a site administrator. For more information, see [“Restricted Options” in SAS System Options: Reference](#).

Syntax

PDFCOMMENT

NOPDFCOMMENT

Syntax Description

PDFCOMMENT

specifies that PDF document comments can be modified.

NOPDFCOMMENT

specifies that PDF document comments cannot be modified.

Details

The PDFCOMMENT option can affect the Commenting document property.

The document property values for PDF security are not changed when you set PDFSECURITY=NONE. The results are the same as not specifying the option at all.

When PDFSECURITY=NONE, the PDFCOMMENT option is enabled and PDF document comments can be modified.

The following table shows how the Commenting document properties are set when you specify the PDFCOMMENT option and set the PDFSECURITY= option to HIGH:

	PDFCOMMENT PDFSECURITY=HIGH	NOPDFCOMMENT PDFSECURITY=HIGH
Commenting	Allowed	Not Allowed

See Also

- [“Securing ODS-Generated PDF Files” in SAS Output Delivery System: User's Guide](#)

System Options:

- [“PDFFILLIN System Option” on page 107](#)
- [“PDFSECURITY= System Option” on page 115](#)

PDFCONTENT System Option

Specifies whether the contents of a PDF document can be changed.

Valid in: Autoexec file, OPTIONS statement,

Category: Log and Procedure Output Control: PDF

PROC OPTIONS
GROUP= PDF

Default: The shipped default is NOPDFCONTENT.

Requirement: Adobe Acrobat Reader or Professional 3.0 and later versions

Notes: Document properties might differ between the Documents Security window, the Show Details window, and SAS PDF option values for secure documents. The values might differ if a document password has only an Open password, or if the Open and Owner passwords are the same value. These windows display property values correctly when both Open and Owner have passwords, the passwords are not the same, and the document is opened with the Open password.

This option can be restricted by a site administrator. For more information, see [“Restricted Options” in SAS System Options: Reference](#).

Syntax

PDFCONTENT

NOPDFCONTENT

Syntax Description

PDFCONTENT

specifies that the contents of a PDF document can be changed.

NOPDFCONTENT

specifies that the contents of a PDF document cannot be changed.

Details

The PDFCONTENT option can affect the Changing the Document document property.

The document property values for PDF security are not changed when you set PDFSECURITY=NONE. The results are the same as not specifying the option at all.

The following table shows how the Changing the Document document property is set when you specify the PDFCONTENT option and set the PDFSECURITY= option to HIGH:

	PDFCONTENT PDFSECURITY=HIGH	NOPDFCONTENT PDFSECURITY=HIGH
Changing the Document	Allowed	Not Allowed

See Also

- [“Securing ODS-Generated PDF Files” in SAS Output Delivery System: User's Guide](#)

System Options:

- [“PDFSECURITY= System Option” on page 115](#)

PDFCOPY System Option

Specifies whether text and graphics from a PDF document can be copied.

Valid in: Autoexec file, OPTIONS statement,

Category: Log and Procedure Output Control: PDF

PROC OPTIONS
GROUP= PDF

Default: The shipped default is PDFCOPY.

Requirement: Adobe Acrobat Reader or Professional 3.0 and later versions

Notes: Document properties might differ between the Documents Security window, the Show Details window, and SAS PDF option values for secure documents. The values might differ if a document password has only an Open password, or if the Open and Owner passwords are the same value. These windows display property values correctly when both Open and Owner have passwords, the passwords are not the same, and the document is opened with the Open password.

This option can be restricted by a site administrator. For more information, see [“Restricted Options” in SAS System Options: Reference](#).

Syntax

PDFCOPY

NOPDFCOPY

Syntax Description

PDFCOPY

specifies that text and graphics from a PDF document can be copied.

NOPDFCOPY

specifies that text and graphics from a PDF document cannot be copied.

Details

The PDFCOPY option can affect the Content Copying document property.

The document property values for PDF security are not changed when you set PDFSECURITY=NONE. The results are the same as not specifying the option at all.

The following table shows how the Content Copying document property is set when you specify the PDFCOPY option and set the PDFSECURITY= option to HIGH:

	PDFCOPY PDFSECURITY=HIGH	NOPDFCOPY PDFSECURITY=HIGH
Content Copying	Allowed	Not Allowed

See Also

- [“Securing ODS-Generated PDF Files” in SAS Output Delivery System: User’s Guide](#)

System Options:

- [“PDFSECURITY= System Option” on page 115](#)

PDFFILLIN System Option

Specifies whether PDF forms can be filled in.

Valid in: Autoexec file, OPTIONS statement,

Category: Log and Procedure Output Control: PDF

PROC OPTIONS PDF

GROUP=

- Default:** The shipped default is PDFFILLIN.
- Requirement:** Adobe Acrobat Reader or Professional 5.0 and later versions
- Notes:** Document properties might differ between the Documents Security window, the Show Details window, and SAS PDF option values for secure documents. The values might differ if a document password has only an Open password, or if the Open and Owner passwords are the same value. These windows display property values correctly when both Open and Owner have passwords, the passwords are not the same, and the document is opened with the Open password.
- This option can be restricted by a site administrator. For more information, see [“Restricted Options” in SAS System Options: Reference](#).

Syntax

PDFFILLIN

NOPDFFILLIN

Syntax Description

PDFFILLIN

specifies that PDF forms can be filled in.

NOPDFFILLIN

specifies that PDF forms cannot be filled in.

Details

The PDFFILLIN option can affect the Form Field Fill-in or Signing document property.

The document property values for PDF security are not changed when you set PDFSECURITY=NONE. The results are the same as not specifying the option at all.

The Form Field Fill-in or Signing document property is set by the PDFFILLIN option only when PDFSECURITY=HIGH. When PDFSECURITY=HIGH, PDFCOMMENT and PDFFILLIN can be set independently.

The following table shows how the Form Field Fill-in or Signing document property is set when you specify the PDFFILLIN option and set the PDFSECURITY= option to HIGH:

	PDFFILLIN	NOPDFFILLIN
	PDFSECURITY=HIGH	PDFSECURITY=HIGH
Form Field Fill-in or Signing	Allowed	Not Allowed

See Also

- [“Securing ODS-Generated PDF Files” in SAS Output Delivery System: User’s Guide](#)

System Options:

- [“PDFCOMMENT System Option” on page 103](#)
- [“PDFSECURITY= System Option” on page 115](#)

PDFPAGELAYOUT= System Option

Specifies the page layout for PDF documents.

Valid in: Autoexec file, OPTIONS statement,

Category: Log and Procedure Output Control: PDF

PROC OPTIONS
GROUP= PDF

Default: The shipped default is DEFAULT.

Requirement: Adobe Acrobat Reader or Professional 5.0 and later versions

Notes: Document properties might differ between the Documents Security window, the Show Details window, and SAS PDF option values for secure documents. The values might differ if a document password has only an Open password, or if the Open and Owner passwords are the same value. These windows display property values correctly when both Open and Owner have passwords, the passwords are not the same, and the document is opened with the Open password.

This option can be restricted by a site administrator. For more information, see [“Restricted Options” in SAS System Options: Reference](#).

Syntax

PDFPAGELAYOUT= [DEFAULT](#) | [SINGLEPAGE](#) | [CONTINUOUS](#)
| [FACING](#) | [CONTINUOUSFACING](#)

Syntax Description

DEFAULT

specifies to use the current page layout for Acrobat Reader.

SINGLEPAGE

specifies to display one page at a time in the viewing area.

CONTINUOUS

specifies to display all document pages in the viewing area in a single column.

FACING

specifies to display only two pages in the viewing area, with the even pages on the left and the odd pages on the right.

Requirement Acrobat Reader 5.0 or later version is required.

CONTINUOUSFACING

specifies to display all pages in the viewing area, two pages side by side. The even pages are displayed on the left, and the odd pages display on the right.

See Also

- [“Securing ODS-Generated PDF Files” in SAS Output Delivery System: User’s Guide](#)

System Options:

- [“PDFPAGEVIEW= System Option” on page 110](#)

PDFPAGEVIEW= System Option

Specifies the page viewing mode for PDF documents.

Valid in: Autoexec file, OPTIONS statement,

Category: Log and Procedure Output Control: PDF

PROC OPTIONS
GROUP= PDF

Default: The shipped default is DEFAULT.

Requirement: Adobe Acrobat Reader or Professional 5.0 and later versions

Note: This option can be restricted by a site administrator. For more information, see [“Restricted Options” in SAS System Options: Reference](#).

Syntax

PDFPAGEVIEW= [DEFAULT](#) | [ACTUAL](#) | [FITPAGE](#) | [FITWIDTH](#) | [FULLSCREEN](#)

Syntax Description

DEFAULT

specifies to use the current page view setting for Acrobat Reader.

ACTUAL

specifies to set the page view setting to 100%.

FITPAGE

specifies to view a page using the full extent of the viewing window, maintaining the height and width aspect ratio.

FITWIDTH

specifies to view a page using the full width of the viewing window. The height of the document is not scaled to fit the page.

FULLSCREEN

specifies to view a page using the full screen. This option disables the table of contents, bookmarks, and all other document access aids, such as accessing a specific page.

See Also

- [“Securing ODS-Generated PDF Files” in SAS Output Delivery System: User's Guide](#)

System Options:

- [“PDFPAGEVIEW= System Option” on page 110](#)

PDFPASSWORD= System Option

Specifies the password to use to open a PDF document and the password used by a PDF document owner to set document permissions and restrictions.

Categories:	Log and Procedure Output Control: PDF System Administration: Security
PROC OPTIONS GROUP=	PDF SECURITY
Alias:	PDFPW
Requirement:	Adobe Acrobat Reader or Professional 3.0 and later versions
Notes:	This option cannot be restricted by a site administrator. For more information, see “Restricted Options” in SAS System Options: Reference . The OPTIONS procedure displays passwords in the SAS log as 8 Xs, regardless of the actual password length.

Syntax

```
PDFPASSWORD=(OPEN=password | OPEN="password"
    < OWNER=password | OWNER="password">)
```

```
PDFPASSWORD=(OWNER=password | OWNER="password")
```

<OPEN=password | OPEN="password">)

PDFPASSWORD=(OPEN=password | OPEN="password")

PDFPASSWORD=(OWNER=password | OWNER="password")

Syntax Description

OPEN="password"

specifies the password to open a PDF document. Enclosing the password in single or double quotation marks is optional.

password

specifies a set of characters, up to 32 characters, that are used to validate that a user has permission to open a PDF document.

Restriction The OPEN password must be different from the OWNER password.

OWNER="password"

specifies the password for the PDF document owner. The OWNER= password is required to set the document permissions and restrictions. Enclosing the password in quotation marks is optional.

password

specifies a set of characters, up to 32 characters, that are used to validate the owner of a PDF document.

Restriction The OWNER password must be different from the OPEN password.

Details

You can set the PDFPASSWORD= option at any time, but it is ignored until the PDFSECURITY= system option is set to HIGH. When the PDFSECURITY= option is set to NONE, passwords for a PDF document are not needed.

The following table explains how passwords are used to open PDF documents and to set the permissions and restrictions using the PDF system options:

PDFPASSWORD= Option Setting	PDF Document Permissions and Restrictions
OPEN=password	A password is required to open the PDF document. No matter what PDF options have been set, no permissions and restrictions have been set because the OWNER=password is not used.
OWNER=password	Use only this option to set permissions and restrictions on the PDF document and to not require a password to open the

PDFPASSWORD= Option Setting	PDF Document Permissions and Restrictions
	document. The value of the PDF options that are set when you create the PDF are applied to the PDF document.
OPEN= <i>password</i> OWNER= <i>password</i>	Use both arguments to require a password to open the PDF document and to set permissions and restrictions on the PDF document. Either password can be used to open the document. The value of the PDF options that are set when you create the PDF are applied to the PDF document.

See Also

- [“Securing ODS-Generated PDF Files” in SAS Output Delivery System: User’s Guide](#)

System Options:

- [“PDFPAGEVIEW= System Option” on page 110](#)
- [“PDFSECURITY= System Option” on page 115](#)

PDFPRINT= System Option

Specifies the resolution to print PDF documents.

Valid in:	Autoexec file, OPTIONS statement,
Category:	Log and Procedure Output Control: PDF
PROC OPTIONS GROUP=	PDF
Default:	The shipped default is HRES for Acrobat Reader or Professional 5.0 and later versions.
Requirement:	Adobe Acrobat Reader or Professional 3.0 and later versions, depending on PDFPRINT setting
Note:	This option can be restricted by a site administrator. For more information, see “Restricted Options” in SAS System Options: Reference .

Syntax

PDFPRINT=HRES | LRES | NONE

Syntax Description

HRES

specifies to print PDF documents at the highest resolution available on the printer.

Restriction PDFPRINT=HRES can be set only when the PDFSECURITY option is set to HIGH.

Requirement Acrobat Reader or Professional 5.0 and later versions.

LRES

specifies to print PDF documents at a lower resolution for draft-quality documents.

Restriction PDFPRINT=LRES can be set only when the PDFSECURITY option is set to HIGH.

Requirement Acrobat Reader or Professional 5.0 and later versions.

NONE

specifies the PDF documents have no print resolution.

Restriction PDFPRINT=NONE can be set only when the PDFSECURITY option is set to HIGH

Requirement Any version of Acrobat Reader or Professional.

Details

The document property values for PDF security are not changed when you set PDFSECURITY=NONE. The results are the same as not specifying the option at all.

When PDFSECURITY= is set to HIGH, the value of the Printing document property is determined by the value of the PDFPRINT= option:

	PDFPRINT=HRES	PDFPRINT=LRES	PDFPRINT=NONE
	PDFSECURITY=HIGH	PDFSECURITY=HIGH	PDFSECURITY=HIGH
Printing	High Resolution	Low Resolution (150 dpi)	None

See Also

- [“Securing ODS-Generated PDF Files” in SAS Output Delivery System: User’s Guide](#)

System Options:

■ “PDFPAGEVIEW= System Option” on page 110

PDFSECURITY= System Option

Specifies the level of encryption for PDF documents.

Valid in:	Autoexec file, OPTIONS statement,
Categories:	Log and Procedure Output Control: PDF System Administration: Security
PROC OPTIONS GROUP=	PDF SECURITY
Default:	The shipped default is NONE.
Restriction:	The PDFSECURITY option is valid for UNIX, Windows, and z/OS operating systems, but only in countries where importing encryption software is legal.
Requirement:	When PDFSECURITY=HIGH, you must use Adobe Acrobat 9 or later or Adobe Reader 9 or later to open the document.
Note:	This option can be restricted by a site administrator. For more information, see “Restricted Options” in SAS System Options: Reference .

Syntax

PDFSECURITY=[HIGH](#) | [NONE](#)

Syntax Description

HIGH

specifies that SAS encrypts PDF documents. Starting with [2023.07](#), PDFSECURITY=HIGH sets the encryption level to 256-bit AES encryption. In prior releases, RC4 128-bit encryption was used to encrypt PDF documents.

Requirement When PDFSECURITY=HIGH, you must use Adobe Acrobat 9 or later or Adobe Reader 9 or later to open the document.

Interaction At least one password must be set using the PDFPASSWORD= system option when PDFSECURITY=HIGH.

NONE

specifies that no encryption is performed on PDF documents.

The document property values for PDF security are not changed when you set PDFSECURITY=NONE. The results are the same as not specifying the option at all.

Details

The following table shows the default document properties that are set when the PDFSECURITY= option is set to NONE or HIGH. When PDFSECURITY=NONE, there are no restrictions on PDF documents.

	NONE	HIGH
Printing	Allowed	High Resolution
Changing the Document	Allowed	Not Allowed
Commenting	Allowed	Not Allowed
Form Field Fill-in or Signing	Allowed	Allowed
Document Assembly	Allowed	Not Allowed
Content Copying	Allowed	Allowed
Document Editing Enabled	Allowed	Allowed
Page Extraction	Allowed	Not Allowed
Encryption Level	None	Starting with 2023.07, 256-bit AES In prior releases, 128-bit RC4

Note: Document properties might differ between the Documents Security window, the Show Details window, and SAS PDF option values for secure documents. The values might differ if a document password has only an Open password, or if the Open and Owner passwords are the same value. These windows display property values correctly when both Open and Owner have passwords, the passwords are not the same, and the document is opened with the Open password.

See Also

- [“Securing ODS-Generated PDF Files” in SAS Output Delivery System: User's Guide](#)

System Options:

- [“PDFACCESS System Option” on page 100](#)
- [“PDFASSEMBLY System Option” on page 102](#)
- [“PDFCOMMENT System Option” on page 103](#)

- “PDFCONTENT System Option” on page 105
- “PDFCOPY System Option” on page 106
- “PDFFILLIN System Option” on page 107
- “PDFPASSWORD= System Option” on page 111
- “PDFPRINT= System Option” on page 113

PRINTERPATH= System Option

Specifies the name of a registered printer to use for Universal Printing.

Valid in: Autoexec file, OPTIONS statement,

Category: Log and Procedure Output Control: ODS Printing

PROC OPTIONS
GROUP= ODSPRINT

Default: Under UNIX, the default is PostScript Level 1.

Notes: This option cannot be restricted by a site administrator. For more information, see [“Restricted Options” in SAS System Options: Reference](#).

This option cannot be restricted by a site administrator. For more information, see [“Restricted Options” in SAS System Options: Reference](#).

Syntax

PRINTERPATH=(*'printer-name'* <fileref>)

Syntax Description

'printer-name'

must be one of the printers defined in the SAS Registry

Requirement When the *printer name* contains blanks, you must enclose it in quotation marks.

fileref

is an optional fileref. If a fileref is specified, it must be defined with a FILENAME statement or an external allocation. If a fileref is not specified, the default output destination can specify a printer in the Printer Setup dialog box, which you open by selecting **File** ⇒ **Printer Setup**. Parentheses are required only when a *fileref* is specified.

Details

If the PRINTERPATH= option is not a null string, then Universal Printing is used. If the PRINTERPATH= option does not specify a valid Universal Printing printer, then the default Universal Printer is used.

Comparisons

A related system option SYSPRINT specifies which operating system printer is used for printing. PRINTERPATH= specifies which Universal Printing printer is used for printing.

The operating system printer specified by the SYSPRINT option is used when PRINTERPATH="" (two double quotation marks with no space between them sets a null string).

Example

The following example specifies an output destination that is different from the default:

```
options PRINTERPATH=(corelab out);  
filename out 'your_file';
```

Operating Environment Information: In some operating environments, setting the PRINTERPATH= option might not change the setting of the PMENU print button, which might continue to use operating environment printing. See the SAS documentation for your operating environment for more information.

The PRINTERPATH option is used only for ODS PRINTER and when the DEVICE= system option is set to SASPRTC, SASPRTRTG, SASPRTM, or SASPRT. If DEVICE=WINPRTC, WINPRTG, or WINPRTM, the devices behave respectively as SASPRTC, SASPRTG, or SASPRTM.

See Also

[Chapter 2, “Setting Up and Managing Universal Printing,” on page 11](#)

RIGHTMARGIN= System Option

Specifies the print margin for the right side of the page.

Valid in: Autoexec file, OPTIONS statement,

Category: Log and Procedure Output Control: ODS Printing

PROC OPTIONS ODSPRINT
GROUP=

Default: The shipped default is 0.000 in.

Note: This option cannot be restricted by a site administrator. For more information, see [“Restricted Options” in SAS System Options: Reference](#).

Syntax

RIGHTMARGIN=*margin-size*<*margin-unit*>

Syntax Description

margin-size

specifies the size of the margin.

Restriction The right margin should be small enough so that the left margin plus the right margin is less than the width of the paper.

Interaction Changing the value of this option might result in changes to the value of the LINESIZE= system option.

<*margin-unit*>

specifies the units for margin-size. The margin-unit can be *in* for inches or *cm* for centimeters. <*margin-unit*> is saved as part of the value of the RIGHTMARGIN system option.

Default inches

Details

All margins have a minimum that is dependent on the printer and the paper size.

See Also

Statements:

- [Chapter 2, “Setting Up and Managing Universal Printing,” on page 11](#)

System Options:

- [“BOTTOMMARGIN= System Option” on page 87](#)
- [“LEFTMARGIN= System Option” on page 94](#)
- [“TOPMARGIN= System Option” on page 120](#)

TOPMARGIN= System Option

Specifies the print margin at the top of the page.

Valid in: Autoexec file, OPTIONS statement,

Category: Log and Procedure Output Control: ODS Printing

PROC OPTIONS
GROUP= ODSPRINT

Default: The shipped default is 0.000 in.

Note: This option cannot be restricted by a site administrator. For more information, see [“Restricted Options” in SAS System Options: Reference](#).

Syntax

TOPMARGIN=*margin-size*<*margin-unit*>

Syntax Description

margin-size

specifies the size of the margin.

Restriction The bottom margin should be small enough so that the top margin plus the bottom margin is less than the height of the paper.

Interaction Changing the value of this option might result in changes to the value of the PAGESIZE= system option.

<*margin-unit*>

specifies the units for margin-size. The margin-unit can be *in* for inches or *cm* for centimeters. <*margin-unit*> is saved as part of the value of the TOPMARGIN system option.

Default inches

Details

All margins have a minimum that is dependent on the printer and the paper size. The default value of the TOPMARGIN system option is 0.00 in.

See Also

- Chapter 2, “Setting Up and Managing Universal Printing,” on page 11

System Options:

- “BOTTOMMARGIN= System Option” on page 87
- “LEFTMARGIN= System Option” on page 94
- “RIGHTMARGIN= System Option” on page 118

UPRINTCOMPRESSION System Option

Specifies whether to enable the compression of files created by some Universal Printer and SAS/GRAPH devices.

Valid in:	Autoexec file, OPTIONS statement,
Category:	Log and Procedure Output Control: ODS Printing
PROC OPTIONS GROUP=	ODSPRINT
Alias:	UPC NOUPC
Default:	The shipped default is UPRINTCOMPRESSION.
Notes:	In SAS Viya, this option is valid only for Universal Printer devices. This option can be restricted by a site administrator. For more information, see “Restricted Options” in SAS System Options: Reference .

Syntax

UPRINTCOMPRESSION

NOUPRINTCOMPRESSION

Syntax Description

UPRINTCOMPRESSION

specifies to enable compression of files created by some Universal Printers and some SAS/GRAPH devices.

NOUPRINTCOMPRESSION

specifies to disable compression of files created by some Universal Printers and some SAS/GRAPH devices.

Details

The following table lists the Universal Printers and the SAS/GRAPH devices that are affected by the UPRINTCOMPRESSION system option:

Universal Printers	SAS/GRAPH Device Drivers
PDF	PDF, PDF/A, PDF/C
SVG	SVG

The ODS PRINTER statement option, COMPRESS=, takes precedence over the UPRINTCOMPRESSION system option.