

PDF vs. HTML: Can't We All Just Get Along?

Scott Huntley, Cynthia Zender, SAS Institute

ABSTRACT

Have you ever asked, "Why doesn't my PDF output look just like my HTML output?" This paper explains the power and differences of each destination. You will learn how each destination works and understand why the output looks the way it does. Learn tips and tricks for how to modify your SAS® code to make each destination look more like the other. The tips span from beginner to advanced in all areas of reporting. Each destination is like a superhero, helping you transform your reports to meet all your needs. Learn how to use each ODS destination to the fullest extent of its powers.

INTRODUCTION

Superheroes are all different. Some superheroes have science and technology on their side (Ironman, Batman, Aquaman); some have super powers because they are from another planet or time (Thor, Superman); while others became accidental superheroes (Hulk, Spiderman). Just as superheroes are able to do marvelous things, in different ways, based on their abilities, each ODS destination has a set of abilities and superpowers based on the underlying architecture and purpose of the destination. This paper addresses two ODS destinations: PDF and HTML.

CREATION STORY

Every superhero has a creation story. PDF and HTML have creation stories too. HTML was probably the first ODS destination. With SAS version 7, ODS HTML and ODS RTF were initially introduced along with the ODS PRINTER destination. The PDF destination was not production until SAS 8.2. The underlying architecture of ODS dictated that style information (colors, fonts, borders, and so on) appropriate to each destination would be sent to each destination (using an ODS style template) along with the data from the SAS process or procedure. Then ODS would write instructions that could be rendered by a 3rd party application.

ODS HTML initially created HTML 3.2 compliant HTML tags or elements, in SAS 7. The rendering application for HTML output was a web browser (although Microsoft Word and Microsoft Excel have been able to open HTML files ever since Office 97). Initially, with ODS the Printer family consisted of ODS PRINT, ODS PS and ODS PCL. When you wanted a PDF file, initially, you created a PostScript file and needed to distill it to PDF form. In SAS 8.2, the ODS PDF destination became production.

Right from the beginning the underlying assumptions of HTML and PDF were different. HTML was designed for screen viewing. PDF could be viewed on the screen but it creates output that is in a format that is readable by Adobe or other 3rd party products that consume PDF. Basically it creates output that is printable. What you see in a viewer like Acrobat will print out with the same appearance. Another nice thing about the PDF destination was that it was a good way to deliver output that would be hard to change without using fancy Adobe editing tools. HTML was difficult to change, too, but mostly because not everyone knew HTML tags and instructions. But, HTML is not a "paged" destination, so things like page numbers and page breaks and printing control really work better in PDF than in HTML (where they are not used at all). The rendering browser controls how an HTML file will be printed, and to some extent you might be able to impact that printing using CSS @media instructions.

WHAT IS MEASURED OUTPUT?

The underlying architecture of PDF versus HTML has far-reaching impact on the output, beyond the concept of which application renders the output. Now we can look at a concrete example. In the code below, two separate reports are created, taking all the defaults and changing orientation. Other changes are minor. HTML uses the default HTMLBLUE style template and PDF uses the PRINTER style. By default PDF also includes a bookmark area, but that feature is easily turned off with the NOTOC option. All the reports use SASHELP.CARS, which is delivered with Base SAS, so it should be easy to replicate these results.

```
options orientation=portrait topmargin=.25in bottommargin=.25in
        leftmargin=.25in rightmargin=.25in number;
ods html file='c:\temp\default1.html' style=htmlblue;
```

```
ods pdf file='c:\temp\default1_portrait.pdf' style=printer notoc;
proc report data=sashelp.cars nowd;
  title 'Default Output Portrait Orientation';
  footnote 'The Footnote';
run;
ods _all_ close;

options orientation=landscape topmargin=.25in bottommargin=.25in
  leftmargin=.25in rightmargin=.25in number;
ods html file='c:\temp\default2.html' style=htmlblue;
ods pdf file='c:\temp\default2_portrait.pdf' style=printer notoc;
proc report data=sashelp.cars nowd;
  title 'Default Output Landscape Orientation';
  footnote 'The Footnote';
run;
ods _all_ close;
```

The HTML output is the same for both programs, as shown in Display 1. The reason that both outputs are the same is that HTML does not use the ORIENTATION option. Everything for HTML is written to a single HTML file, which represents one (1) web “page”. It is somewhat odd that the HTML specification has uses the term “page” to describe what is being displayed, since a single web page could be 5000 observations long, definitely too much to be printed on one physical page. And, as the annotations in red point out, on the HTML page, there is a single title at the top of the browser display and a single footnote at the bottom of the browser display.

Default Output Portrait Orientation														
Make	Model	Type	Origin	DT	MSRP	Invoice	Engine Size (L)	Cyl	Title at top of output			Wheelbase (IN)	Length (IN)	
Acura	MDX	SUV	Asia	All	\$36,945	\$33,337	3.5	6	265	17	23	4451	106	189
Acura	RSX Type S 2dr	Sedan	Asia	Front	\$23,820	\$21,761	2	4	200	24	31	2778	101	172
Acura	TSX 4dr	Sedan	Asia	Front	\$26,990	\$24,647	2.4	4	200	22	29	3230	105	183
Acura	TL 4dr	Sedan	Asia	Front	\$33,195	\$30,299	3.2	6	270	20	28	3575	108	186
Acura	3.5 RL 4dr	Sedan	Asia	Front	\$43,755	\$39,014	3.5	6	225	18	24	3880	115	197
Acura	3.5 RL w/Navigation 4dr	Sedan	Asia	Front	\$46,100	\$41,100	3.5	6	225	18	24	3893	115	197
Acura	NSX coupe 2dr manual S	Sports	Asia	Rear	\$89,765	\$79,978	3.2	6	290	17	24	3153	100	174
Audi	A4 1.8T 4dr	Sedan	Europe	Front	\$25,940	\$23,508	1.8	4	170	22	31	3252	104	179
Audi	A4 1.8T convertible 2dr	Sedan	Europe	Front	\$35,940	\$32,506	1.8	4	170	23	30	3638	105	180
Audi	A4 3.0 4dr	Sedan	Europe	Front	\$31,840	\$28,846	3	6	220	20	28	3462	104	179
Audi	A4 3.0 Quattro 4dr manual	Sedan	Europe	All	\$33,430	\$30,366	3	6	220	17	26	3583	104	179
Audi	A4 3.0 Quattro 4dr auto	Sedan	Europe	All	\$34,480	\$31,388	3	6	220	18	25	3627	104	179
Audi	A6 3.0 4dr	Sedan	Europe	Front	\$36,640	\$33,129	3	6	220	20	27	3561	109	192
Audi	A6 3.0 Quattro 4dr	Sedan	Europe	All	\$39,640	\$35,992	3	6	220	18	25	3880	109	192
Audi	A4 3.0 convertible 2dr	Sedan	Europe	Front	\$42,490	\$38,325	3	6	220	20	27	3814	105	180
Audi	A4 3.0 Quattro convertible 2dr	Sedan	Europe	All	\$44,240	\$40,075	3	6	220	18	25	4013	105	180
Audi	A6 2.7 Turbo Quattro 4dr	Sedan	Europe	All	\$42,840	\$38,840	2.7	6	250	18	25	3836	109	192
Audi	A6 4.2 Quattro 4dr	Sedan	Europe	All	\$49,690	\$44,936	4.2	8	300	17	24	4024	109	193
Volvo	S60 2.5 4dr	Sedan	Europe	All	\$31,745	\$29,916	2.5	5	208	20	27	3903	107	180
Volvo	S60 T5 4dr	Sedan	Europe	Front	\$34,845	\$32,902	2.3	5	247	20	28	3766	107	180
Volvo	S60 R 4dr	Sedan	Europe	All	\$37,560	\$35,382	2.5	5	300	18	25	3571	107	181
Volvo	S80 2.9 4dr	Sedan	Europe	Front	\$37,730	\$35,542	2.9	6	208	20	28	3576	110	190
Volvo	S80 2.5T 4dr	Sedan	Europe	All	\$37,885	\$35,688	2.5	5	194	20	27	3691	110	190
Volvo	C70 LPT convertible 2dr	Sedan	Europe	Front	\$40,565	\$38,203	2.4	5	197	21	28	3450	105	186
Volvo	C70 HPT convertible 2dr	Sedan	Europe	Front	\$42,565	\$40,083	2.3	5	242	20	26	3450	105	186
Volvo	S80 T6 4dr	Sedan	Europe	Front	\$45,210	\$42,573	2.9	6	268	19	26	3653	110	190
Volvo	V40	Wagon	Europe	Front	\$26,135	\$24,641	1.9	4	170	22	29	2822	101	180
Volvo	XC70	Wagon	Europe	All	\$35,145	\$33,112						23	109	186
The Footnote														
Footnote at bottom of output														

Display 1. Partial HTML Results

However, with PDF output, the concept of measured output comes into play. What is measured output in regard to ODS PDF? “Measured output” means ODS will determine the amount of available space for output. Unlike HTML the PDF destination is very concerned about height and width of output objects such as tables, text, graphs, and images. A good analogy is to think about a piece of paper. How much output can you fit on that paper? The height and width of the paper is fixed. Therefore, we have to make sure our output fits perfectly. ODS has to take measurable factors into account, including, but not limited to, paper-size being used, system margins, orientation of the output and font size of the output. This means that Printer family output, including ODS PDF is bound by physical limitations.

For example, the Portrait output is 26 pages and the Landscape output is 18 pages. What's the reason for the difference? In the Portrait output, the results do not all fit on a hypothetical piece of 8 1/2 x 11 paper. As shown in Display 2, the PDF output displays variables Make -- Invoice on page 1 and then for that same group of observations, displays variables EngineSize -- Length on page 2. On the other hand, with the landscape orientation, all of the variables can be displayed in landscape mode. Therefore, the page count is different.

The other concept of measured output is shown in both Display 2 and Display 3. This output shows how ODS measured the placement of the report rows, titles, and footnotes based on the fact that every character and report row and text string in the output must be accounted for when ODS PDF lays out the page in memory before writing to the output file (either the default file or the file you specify with the FILE= option).

Default Output Portrait Orientation

Make	Model	Type	Origin	DT	MSRP	Invoice
Acura	MDX	SUV	Asia	AD	\$36,945	\$33,337
Acura	RDX Type S 3dr	Sedan	Asia	Front	\$23,020	\$21,761
Acura	TSX 4dr	Sedan	Asia	Front	\$26,990	\$24,647
Acura	TL 4dr	Sedan	Asia	Front	\$33,195	\$30,299
Acura	3.5 RL 4dr	Sedan	Asia	Front	\$43,755	\$39,014
Acura	3.5 RL w/Navigation 4dr	Sedan	Asia	Front	\$46,100	\$41,100
Acura	NCC coupe 2dr manual S	Sports	Asia	Rear	\$89,765	\$79,978
Audi	A4 1.8T 4dr	Sedan	Europe	Front	\$25,940	\$23,568
Audi	A4 1.8T convertible 2dr	Sedan	Europe	Front	\$35,940	\$32,566
Audi	A4 3.0 4dr	Sedan	Europe	Front	\$31,840	\$28,844
Audi	A4 3.0 Quattro 4dr manual	Sedan	Europe	AD	\$33,470	\$30,366
Audi	A4 3.0 Quattro 4dr auto	Sedan	Europe	AD	\$34,480	\$31,386
Audi	A6 3.0 4dr	Sedan	Europe	Front	\$36,640	\$33,129
Audi	A6 3.0 Quattro 4dr	Sedan	Europe	AD	\$39,640	\$35,992
Audi	A4 3.0 convertible 2dr	Sedan	Europe	Front	\$42,490	\$38,325
Audi	A4 3.0 Quattro convertible 2dr	Sedan	Europe	AD	\$44,740	\$40,075
Audi	A6 2.7 Turbo Quattro 4dr	Sedan	Europe	AD	\$42,840	\$38,040
Audi	A6 4.2 Quattro 4dr	Sedan	Europe	AD	\$49,690	\$44,936
Audi	A8 1. Quattro 4dr	Sedan	Europe	AD	\$69,190	\$64,740
Audi	S4 Quattro 4dr	Sedan	Europe	AD	\$40,040	\$43,556
Audi	B5 6 4dr	Sports	Europe	Front	\$84,600	\$76,417
Audi	TT 1.8 convertible 2dr (coupe)	Sports	Europe	Front	\$35,940	\$32,512
Audi	TT 1.8 Quattro 2dr (convertible)	Sports	Europe	AD	\$37,390	\$33,891
Audi	TT 2.2 coupe 2dr (convertible)	Sports	Europe	AD	\$40,390	\$36,739
Audi	A8 3.0 Avant Quattro	Wagon	Europe	AD	\$40,840	\$37,066
Audi	S4 Avant Quattro	Wagon	Europe	AD	\$49,090	\$44,446
BMW	X3 3.0i	SUV	Europe	AD	\$37,000	\$33,873
BMW	X3 4.4i	SUV	Europe	AD	\$52,199	\$47,720
BMW	328i 4dr	Sedan	Europe	Rear	\$28,899	\$26,193
BMW	325Ci 3dr	Sedan	Europe	Rear	\$30,799	\$28,245
BMW	325Ci convertible 2dr	Sedan	Europe	Rear	\$37,999	\$34,800
BMW	328xi 4dr	Sedan	Europe	AD	\$30,245	\$27,745
BMW	330i 4dr	Sedan	Europe	Rear	\$35,499	\$32,525

Default Output Portrait Orientation

Engine Size (L)	Cyl	HP	MPG (City)	MPG (Hwy)	Weight (LBS)	Wheelbase (IN)	Length (IN)
3.5	6	265	17	23	4451	106	189
3	4	100	24	31	3778	101	172
2.4	4	200	22	29	3239	105	183
3.2	6	270	20	28	3675	106	186
3.5	6	225	18	24	3880	115	197
3.5	6	225	18	24	3893	115	197
3.2	6	290	17	24	3153	100	174
1.8	4	170	22	31	3252	104	179
1.8	4	170	23	30	3638	105	180
3	6	220	20	28	3462	104	179
3	6	220	17	26	3183	104	179
3	6	220	18	25	3627	104	179
3	6	220	20	27	3161	109	192
3	6	220	18	26	3880	109	192
3	6	220	20	27	3814	105	180
3	6	220	18	25	4013	105	190
2.7	6	250	18	25	3836	109	192
4.2	8	300	17	24	4024	109	192
4.2	8	330	17	24	4199	121	204
4.3	8	340	14	20	3825	104	179
4.3	8	410	15	23	4034	109	191
1.8	4	180	20	28	3131	95	159
1.8	4	225	20	28	3921	96	159
3.2	6	260	21	29	3391	96	159
3	6	220	18	25	4035	109	192
4.2	8	340	15	21	3936	104	179
3	6	225	16	23	4023	110	180
4.4	8	325	16	22	4624	111	194
2.5	6	184	20	29	3219	107	176
2.5	6	184	20	28	3197	107	177
2.5	6	184	19	27	3560	107	177
2.5	6	184	19	27	3461	107	176
3	6	225	20	30	3289	107	176

The Footnote

Display 2. Partial PDF Results Orientation=Portrait

When you download the programs that accompany this paper, you will see that we also applied a LABEL to some of the variables to make the Header cells smaller, such as using 'HP' for HorsePower and 'DT' for DriveTrain. Notice how the SAS title and footnote statements were used in the measured PDF output. The title and footnote appears at the top and bottom of each page, as they would, if you routed this output to a physical printer. In addition, the NUMBER system option was used to place page numbers in the PDF (but not the HTML) output.

default2_landscape.pdf - Adobe Reader

File Edit View Window Help

1 / 18 67%

Default Output Landscape Orientation

Make	Model	Type	Origin	DT	MSRP	Invoice	Engine Size (L)	Cyl	HP	MPG (City)	MPG (Hwy)	Weight (LBS)	Wheelbase (IN)	Length (IN)
Acura	NSX	SLV	Asia	AD	\$16,949	\$15,337	3.5	6	285	17	25	4431	106	189
Acura	RDX Type S 2dr	Sedan	Asia	From	\$22,829	\$21,761	2	4	200	24	21	2770	101	172
Acura	TLX 4dr	Sedan	Asia	From	\$26,999	\$26,467	2.4	4	200	22	29	3220	109	182
Acura	TLX 4dr	Sedan	Asia	From	\$22,195	\$20,299	2.2	4	170	20	28	2972	108	180
Acura	1.5 RL 4dr	Sedan	Asia	From	\$42,709	\$39,014	3.5	6	228	18	24	3880	115	197
Acura	1.5 RL w/Navigation 4dr	Sedan	Asia	From	\$46,590	\$42,590	3.5	6	228	18	24	3883	115	197
Acura	NSX coupe 2dr manual 2	Spots	Asia	From	\$89,769	\$79,979	3.2	6	200	17	24	3233	100	174
Audi	A4 1.8T 4dr	Sedan	Europe	From	\$25,940	\$23,566	1.8	4	170	22	31	3253	104	179
Audi	A4L 3T convertible 2dr	Sedan	Europe	From	\$25,940	\$22,500	1.8	4	170	22	30	3630	105	180
Audi	A4 1.8 4dr	Sedan	Europe	From	\$21,940	\$20,940	1	6	120	20	28	2482	104	179
Audi	A4 1.8 Quattro 4dr manual	Sedan	Europe	AD	\$27,439	\$26,364	1	6	120	17	26	2583	104	179
Audi	A4 1.8 Quattro 4dr auto	Sedan	Europe	AD	\$24,499	\$21,289	1	6	120	18	25	2427	104	179
Audi	A5 1.8 4dr	Sedan	Europe	From	\$26,640	\$25,129	1	6	120	20	27	2561	109	182
Audi	A5 2.0 Quattro 4dr	Sedan	Europe	AD	\$29,640	\$27,992	2	6	220	18	23	2880	109	182
Audi	A4 1.8 convertible 2dr	Sedan	Europe	From	\$42,499	\$38,129	1	6	120	20	27	2614	109	180
Audi	A4 1.8 Quattro convertible 2dr	Sedan	Europe	AD	\$46,240	\$40,075	1	6	120	18	25	4057	109	180
Audi	A5 2.7 Turbo Quattro 4dr	Sedan	Europe	AD	\$42,640	\$35,540	2.7	6	280	18	23	2856	109	182
Audi	A5 4.2 Quattro 4dr	Sedan	Europe	AD	\$49,099	\$44,926	4.2	8	300	17	24	4024	109	182
Audi	A5 4.2 Quattro 4dr	Sedan	Europe	AD	\$59,199	\$54,740	4.2	8	330	17	24	4399	121	204
Audi	S4 Quattro 4dr	Sedan	Europe	AD	\$48,040	\$41,594	4.2	8	340	14	20	3820	104	179
Audi	R5 4 4dr	Spots	Europe	From	\$54,600	\$76,427	4.2	8	450	15	22	4024	109	191
Audi	TT 1.8 convertible 2dr (coupe)	Spots	Europe	From	\$23,940	\$22,512	1.8	4	180	20	28	2121	95	159
Audi	TT 1.8 Quattro 2dr (convertible)	Spots	Europe	AD	\$27,299	\$22,891	1.8	4	225	20	28	2921	99	159
Audi	TT 2.2 coupe 2dr (convertible)	Spots	Europe	AD	\$40,590	\$36,739	3.2	6	250	21	29	3351	99	159

The Footnote

Default Output Landscape Orientation

Make	Model	Type	Origin	DT	MSRP	Invoice	Engine Size (L)	Cyl	HP	MPG (City)	MPG (Hwy)	Weight (LBS)	Wheelbase (IN)	Length (IN)
Audi	A4 1.8 Avant Quattro	Wagon	Europe	AD	\$40,040	\$37,060	1	6	120	18	25	4015	109	182
Audi	S4 Avant Quattro	Wagon	Europe	AD	\$49,099	\$44,440	4.2	8	340	15	21	3916	104	179
BMW	323i 4dr	SLV	Europe	AD	\$17,340	\$15,873	3	6	225	16	22	4023	119	180
BMW	323i 4dr	SLV	Europe	AD	\$12,145	\$47,729	4.4	8	225	16	22	4024	119	180
BMW	325i 4dr	Sedan	Europe	From	\$26,495	\$26,155	2.5	6	184	20	29	3219	107	176
BMW	325i 4dr	Sedan	Europe	From	\$30,705	\$28,345	2.5	6	184	20	29	3187	107	176
BMW	325i 4dr	Sedan	Europe	From	\$27,895	\$24,800	2.5	6	184	19	27	2560	107	177
BMW	325i 4dr	Sedan	Europe	AD	\$30,245	\$27,745	2.5	6	184	19	27	2461	107	176
BMW	330i 4dr	Sedan	Europe	From	\$25,495	\$22,325	3	6	225	20	30	3285	107	176
BMW	330i 4dr	Sedan	Europe	From	\$36,895	\$33,090	3	6	225	20	30	3285	107	176
BMW	330i 4dr	Sedan	Europe	AD	\$37,240	\$34,115	3	6	225	20	30	3483	107	176
BMW	325i 4dr	Sedan	Europe	From	\$29,095	\$26,620	2.5	6	184	19	29	2420	114	191
BMW	330i convertible 2dr	Sedan	Europe	From	\$44,295	\$40,530	3	6	225	18	26	3616	107	177
BMW	330i 4dr	Sedan	Europe	From	\$44,895	\$41,170	3	6	225	20	30	3472	114	191
BMW	340i 4dr	Sedan	Europe	From	\$74,395	\$70,270	4.4	8	225	18	26	3814	114	191
BMW	740i 4dr	Sedan	Europe	From	\$103,195	\$103,190	4.4	8	225	18	26	4776	119	190
BMW	740i 4dr	Sedan	Europe	From	\$73,195	\$66,830	4.4	8	225	18	26	4494	125	204
BMW	342 coupe 2dr	Spots	Europe	From	\$48,195	\$44,170	3.2	6	223	16	24	3415	108	177
BMW	342 convertible 2dr	Spots	Europe	From	\$56,395	\$51,815	3.2	6	223	16	23	3781	108	177
BMW	24 convertible 2 dr 2dr	Spots	Europe	From	\$33,895	\$31,065	2.5	6	184	20	28	2632	96	161
BMW	24 convertible 2 dr 2dr	Spots	Europe	From	\$43,040	\$37,175	3	6	225	21	29	2990	99	161
BMW	225i 4dr	Wagon	Europe	AD	\$22,640	\$20,110	2.5	6	184	19	24	2594	107	176
Buick	Rainier	SLV	USA	AD	\$27,899	\$24,237	4.2	8	275	15	21	4899	113	193
Buick	Rainier	SLV	USA	From	\$26,345	\$24,085	3.4	6	185	18	26	4024	113	197

The Footnote

Display 3. Partial PDF Results Orientation=Landscape

Now that we have shown some fundamental differences between HTML and PDF output, we can discuss more about how to take control of your output, beyond simple tricks like changing the labels.

HOW TO SIZE YOUR OUTPUT

Since PDF adheres to the concept of measured output, there is more you can do with PDF in terms of controlling the size of your output to maximize space on a physical page. For HTML, it is not so critical to maximize space on a physical page, since the concept of physical pages is not relevant to HTML. When we show some of the controls that apply to PDF, we will also show how those changes impact the HTML output.

One easy change to impact PDF output is to change margins and orientation as shown in the Display 2 and 3 and this will, in turn, impact number of pages that “fit” in each orientation. Other ways exist to impact the size of the output, one such is font size. We can make two simple changes to the PROC REPORT output to show how font size and cell padding can impact output. Consider the following change to the PROC REPORT code:

```
proc report data=sashelp.cars nowd
  style(report)={fontsize=9pt cellpadding=2px}
  style(header)={fontsize=9pt}
  style(column)={fontsize=9pt};
```

As shown in Display 4, the PDF output, now fits in 8 portrait pages instead of the 26 pages before. The reason for this difference is that those 2 simple changes allowed ODS to “measure” the PDF output differently, so that all the variables in the report row would fit into one portrait page. Although landscape output is not shown, using these same style overrides with PDF output caused the number of landscape pages to shrink to 11 pages instead of the original 18 pages.

Display 5. Partial PDF Results with Font and Large Cell Padding Changes

With this unusually large value for cell padding, the report row is again, too wide to fit on one portrait page and the number of total pages in the output has increased to 40. These considerations do have an impact on HTML, as shown in Display 6, but not as dramatic as increasing or decreasing the number of pages, because the only thing that happens is that you scroll a bit more or a bit less in the browser. Even though there are no page breaks, you can use these techniques with HTML to impact how much content fits on a single screen.

Make	Model	Type	Origin	DT	MSRP	Invoice	Engine Size (L)	Cyl	HP	MPG (City)	MPG (Hwy)	Weight (LBS)	Wheelbase (IN)	Length (IN)
Acura	MDX	SUV	Asia	All	\$36,945	\$33,337	3.5	6	265	17	23	4451	106	189
Acura	RSX Type S 2dr	Sedan	Asia	Front	\$23,820	\$21,761	2	4	200	24	31	2778	101	172
Acura	TSX 4dr	Sedan	Asia	Front	\$26,990	\$24,647	2.4	4	200	22	29	3230	105	183
Acura	TL 4dr	Sedan	Asia	Front	\$33,195	\$30,299	3.2	6	270	20	28	3575	108	186
Acura	3.5 RL 4dr	Sedan	Asia	Front	\$43,755	\$39,014	3.5	6	225	18	24	3880	115	197
Acura	3.5 RL w/Navigation 4dr	Sedan	Asia	Front	\$46,100	\$41,100	3.5	6	225	18	24	3893	115	197
Acura	NSX coupe 2dr manual S	Sports	Asia	Rear	\$89,765	\$79,978	3.2	6	290	17	24	3153	100	174
Audi	A4 1.8T 4dr	Sedan	Europe	Front	\$25,940	\$23,508	1.8	4	170	22	31	3252	104	179
Audi	A4 1.8T convertible 2dr	Sedan	Europe	Front	\$35,940	\$32,506	1.8	4	170	23	30	3638	105	180
Audi	A4 3.0 4dr	Sedan	Europe	Front	\$31,840	\$28,846	3	6	220	20	28	3462	104	179
Audi	A4 3.0 Quattro 4dr manual	Sedan	Europe	All	\$33,430	\$30,366	3	6	220	17	26	3583	104	179

Make	Model	Type	Origin	DT	MSRP	Invoice	Engine Size (L)	Cyl	HP	MPG (City)	MPG (Hwy)	Weight (LBS)	Wheelbase (IN)	Length (IN)
Acura	MDX	SUV	Asia	All	\$36,945	\$33,337	3.5	6	265	17	23	4451	106	189
Acura	RSX Type S 2dr	Sedan	Asia	Front	\$23,820	\$21,761	2	4	200	24	31	2778	101	172
Acura	TSX 4dr	Sedan	Asia	Front	\$26,990	\$24,647	2.4	4	200	22	29	3230	105	183
Acura	TL 4dr	Sedan	Asia	Front	\$33,195	\$30,299	3.2	6	270	20	28	3575	108	186
Acura	3.5 RL 4dr	Sedan	Asia	Front	\$43,755	\$39,014	3.5	6	225	18	24	3880	115	197
Acura	3.5 RL w/Navigation 4dr	Sedan	Asia	Front	\$46,100	\$41,100	3.5	6	225	18	24	3893	115	197

Display 6. Partial HTML Results with Font and Cell Padding Changes

PAGES OF OUTPUT (OR WHAT IF MY OUTPUT IS TOO TALL)

So far, the control over page breaking has been "implicit" or implied paging. When there are too many report rows to fit on a page (in paged destinations), a new page of output is started. The SAS titles and footnotes will take up space on every page in paged destinations, like PDF, but will appear only at the top and bottom of the output table in HTML.

We can move outside the world of the simple listing report and talk about explicit page breaking. An explicit page break is one that is inserted in procedure output by procedure controls, such as using the PAGE dimension in PROC TABULATE or the PAGE option in PROC REPORT or BY and PAGEBY with PROC PRINT. Since ODS destinations, except for LISTING, do not use LINESIZE and PAGESIZE options.

One simple way to break up the output is to add explicit page breaks using procedure controls. To that end, we will start with a switch of procedures and show implicit page breaks with PROC TABULATE and then explicit page breaks with a few other procedures.

The code that we are starting with is shown below:

```
ods html file='c:\temp\demo2_implicit_page.html' style=htmlblue;
ods pdf file='c:\temp\demo2_implicit_page.pdf' style=printer;
proc tabulate data=sashelp.cars;
  title '1) Implicit Page Break from Procedure';
  where make in ('Audi', 'Volvo', 'BMW', 'Chevrolet') and
        type in ('Sedan', 'Wagon');
  class make model type;
  var msrp mpg_highway mpg_city;
  table make * model,
        type*mean*(msrp mpg_highway mpg_city);
run;
```

```
ods _all_ close;
```

There are so many values for MAKE and MODEL that the TABULATE output is difficult to read, as shown in Display 7.

demo2_implicit_page.pdf - Adobe Reader

File Edit View Window Help

1 / 2 78.7%

1) Implicit Page Break from Procedure

Make	Model	Type					
		Sedan			Wagon		
		Mean			Mean		
		MSRP	MPG (Highway)	MPG (City)	MSRP	MPG (Highway)	MPG (City)
Audi	A4 1.8T 4dr	29940.00	31.00	22.00			
	A4 3.0 4dr	31840.00	28.00	20.00			
	A4 3.0 Quattro 4dr auto	34480.00	25.00	18.00			
	A4 3.0 Quattro 4dr manual	33430.00	26.00	17.00			
	A4 3.0 Quattro convertible 2dr	44240.00	25.00	18.00			
	A4 3.0 convertible 2dr	42490.00	27.00	20.00			
	A4 1.8T convertible 2dr	39940.00	30.00	23.00			
	A6 2.7 Turbo Quattro 4dr	42840.00	25.00	18.00			
	A6 3.0 4dr	36640.00	27.00	20.00			
	A6 3.0 Avant Quattro				40840.00	25.00	18.00
BMW	A6 3.0 Quattro 4dr	39640.00	25.00	18.00			
	A6 4.2 Quattro 4dr	49690.00	24.00	17.00			
	A8 L Quattro 4dr	69190.00	24.00	17.00			
	S4 Avant Quattro				49090.00	21.00	15.00
	S4 Quattro 4dr	40040.00	29.00	14.00			
	325Ci 2dr	30795.00	29.00	20.00			
	325Ci convertible 2dr	37995.00	27.00	19.00			
	325i 4dr	26495.00	29.00	20.00			
	325xi 4dr	30245.00	27.00	19.00			
	325xi Sport				32045.00	26.00	19.00
	330Ci 2dr	36995.00	30.00	20.00			
	330Ci convertible 2dr	44795.00	28.00	19.00			
	330i 4dr	31495.00	30.00	20.00			
	330xi 4dr	37245.00	29.00	20.00			
	525i 4dr	36995.00	28.00	19.00			
	530i 4dr	44995.00	30.00	20.00			
	545iA 4dr	54995.00	26.00	18.00			
	745Li 4dr	73195.00	26.00	18.00			
	745i 4dr	69195.00	26.00	18.00			

(Continued)

1) Implicit Page Break from Procedure

Make	Model	Type						
		Sedan			Wagon			
		Mean			Mean			
		MSRP	MPG (Highway)	MPG (City)	MSRP	MPG (Highway)	MPG (City)	
Chevrolet	Astro	26395.00	17.00	14.00				
	Aveo 4dr	11490.00	34.00	28.00				
	Aveo LS 4dr hatch	12585.00	34.00	28.00				
	Cavalier 2dr	14610.00	37.00	26.00				
	Cavalier 4dr	14610.00	37.00	26.00				
	Cavalier LS 2dr	16385.00	37.00	26.00				
	Impala 4dr	21900.00	32.00	21.00				
	Impala LS 4dr	25000.00	30.00	20.00				
	Impala SS 4dr	27995.00	28.00	18.00				
	Malibu 4dr	18995.00	34.00	24.00				
	Malibu LS 4dr	20370.00	30.00	22.00				
	Malibu LT 4dr	23495.00	32.00	23.00				
	Malibu Maxx LS				22225.00	30.00	22.00	
	Monte Carlo LS 2dr	21825.00	32.00	21.00				
	Monte Carlo SS 2dr	24225.00	28.00	18.00				
	Venture LS	27020.00	26.00	19.00				
	Volvo	C70 HPT convertible 2dr	42565.00	26.00	20.00			
		C70 LPT convertible 2dr	40565.00	28.00	21.00			
		S40 4dr	25135.00	29.00	22.00			
		S60 2.5 4dr	31745.00	27.00	20.00			
S60 R 4dr		37660.00	25.00	18.00				
S60 T5 4dr		34845.00	28.00	20.00				
S80 2.5T 4dr		37865.00	27.00	20.00				
S80 2.9 4dr		37730.00	28.00	20.00				
S80 T6 4dr		45710.00	26.00	19.00				
V40					26135.00	29.00	22.00	
XC70				35145.00	27.00	20.00		

Display 7. PDF Results from PROC TABULATE with Implicit Page Breaks

But, with a change to the table statement to create a PAGE dimension results in the output shown in Display 8.

```
table make,
      model,
      type*mean*(msrp mpg_highway mpg_city);
```

demo2_explicit_page.pdf - Adobe Reader

File Edit View Window Help

1 / 4 78.7%

2) Explicit Page Break from Procedure

1

Make Audi

Model	Type					
	Sedan			Wagon		
	Mean			Mean		
	MSRP	MPG (Highway)	MPG (City)	MSRP	MPG (Highway)	MPG (City)
A4 1.8T 4dr	29940.00	31.00	22.00			
A4 3.0 4dr	31840.00	28.00	20.00			
A4 3.0 Quattro 4dr auto	34480.00	25.00	18.00			
A4 3.0 Quattro 4dr manual	33430.00	26.00	17.00			
A4 3.0 Quattro convertible 2dr	44240.00	25.00	18.00			
A4 3.0 convertible 2dr	42490.00	27.00	20.00			
A4 1.8T convertible 2dr	39940.00	30.00	23.00			
A6 2.7 Turbo Quattro 4dr	42840.00	25.00	18.00			
A6 3.0 4dr	36640.00	27.00	20.00			
A6 3.0 Avant Quattro				40840.00	25.00	18.00
A6 3.0 Quattro 4dr	39640.00	25.00	18.00			
A6 4.2 Quattro 4dr	49690.00	24.00	17.00			
A8 L Quattro 4dr	69190.00	24.00	17.00			
S4 Avant Quattro				49090.00	21.00	15.00
S4 Quattro 4dr	40040.00	20.00	14.00			

2) Explicit Page Break from Procedure

2

Make BMW

Model	Type					
	Sedan			Wagon		
	Mean			Mean		
	MSRP	MPG (Highway)	MPG (City)	MSRP	MPG (Highway)	MPG (City)
325Ci 2dr	30795.00	29.00	20.00			
325Ci convertible 2dr	37995.00	27.00	19.00			
325i 4dr	26495.00	29.00	20.00			
325xi 4dr	30245.00	27.00	19.00			
325xi Sport				32045.00	26.00	19.00
330Ci 2dr	36995.00	30.00	20.00			
330Ci convertible 2dr	44795.00	28.00	19.00			
330i 4dr	31495.00	30.00	20.00			
330xi 4dr	37245.00	29.00	20.00			
525i 4dr	36995.00	28.00	19.00			
530i 4dr	44995.00	30.00	20.00			
545iA 4dr	54995.00	26.00	18.00			
745Li 4dr	73195.00	26.00	18.00			
745i 4dr	69195.00	26.00	18.00			

Display 8. PDF Results from PROC TABULATE with Explicit Page Breaks

Using the same page dimension technique with HTML causes a slightly different result. Using the default HTMLBLUE style, there is a horizontal rule at the logical page break. In addition, the SAS title repeats at the top of the table and, if there were a footnote, the footnote would appear under the table on each logical page.

2) Explicit Page Break from Procedure

Model	Type					
	Sedan			Wagon		
	Mean			Mean		
	MSRP	MPG (Highway)	MPG (City)	MSRP	MPG (Highway)	MPG (City)
A4 1.8T 4dr	25940.00		31.00	22.00		
A4 3.0 4dr	31040.00		28.00	20.00		
A4 3.0 Quattro 4dr auto	34480.00		25.00	18.00		
A4 3.0 Quattro 4dr manual	33430.00		26.00	17.00		
A4 3.0 Quattro convertible 2dr	44240.00		25.00	18.00		
A4 3.0 convertible 2dr	42490.00		27.00	20.00		
A41.8T convertible 2dr	35940.00		30.00	23.00		
A6 2.7 Turbo Quattro 4dr	42840.00		25.00	18.00		
A6 3.0 4dr	36640.00		27.00	20.00		
A6 3.0 Avant Quattro				40840.00	25.00	18.00
A6 3.0 Quattro 4dr	39640.00		25.00	18.00		
A6 4.2 Quattro 4dr	49690.00		24.00	17.00		
A8 L Quattro 4dr	69190.00		24.00	17.00		
S4 Avant Quattro				49090.00	21.00	15.00
S4 Quattro 4dr	48040.00		20.00	14.00		

2) Explicit Page Break from Procedure

Model	Type					
	Sedan			Wagon		
	Mean			Mean		
	MSRP	MPG (Highway)	MPG (City)	MSRP	MPG (Highway)	MPG (City)
325CI 2dr	30795.00		29.00	20.00		
328CI convertible 2dr	32065.00		32.00	18.00		

Display 9. HTML Results from PROC TABULATE with Explicit Page Breaks

If you have a browser that supports CSS (Cascading Style sheets), you might be grateful that the horizontal rule is there. In the style template, the horizontal rule comes with a CSS instruction for page breaking:

```
class html
  "Common HTML text used in the default style" /
  'expandAll' = "<span onclick=""expandCollapse()"">"
  'posthtml flyover line' = "</span><hr size=""3""/>"
  'prehtml flyover line' = "<span><hr size=""3""/>"
  'prehtml flyover bullet' = %nrstr("<span><b>&#183;</b>")
  'posthtml flyover' = "</span>"
  'prehtml flyover' = "<span>"
  'break' = "<br/>"
  'Line' = "<hr size=""3""/>"
  'PageBreakLine' =
    "<p style=""page-break-after: always;""><br/></p><hr size=""3""/>"
  'fake bullet' = %nrstr("<b>&#183;</b>");
```

This means that when you print from a browser, such as Internet Explorer, if the CSS command is respected, the page break command will be used by the browser, as shown in Display 10.

Print Preview

2 Page View Shrink To Fit

SAS Output Page 1 of 4

2) Explicit Page Break from Procedure

Make Audi

Model	Type					
	Sedan			Wagon		
	MSRP	MPG (Highway)	MPG (City)	MSRP	MPG (Highway)	MPG (City)
A4 1.8T 4dr	25940.00	31.00	22.00			
A4 3.0 4dr	31840.00	28.00	20.00			
A4 3.0 Quattro 4dr auto	34480.00	25.00	18.00			
A4 3.0 Quattro 4dr manual	33430.00	26.00	17.00			
A4 3.0 Quattro convertible 2dr	44240.00	25.00	18.00			
A4 3.0 convertible 2dr	42490.00	27.00	20.00			
A4 1.8T convertible 2dr	35940.00	30.00	23.00			
A6 2.7 Turbo Quattro 4dr	42840.00	25.00	18.00			
A6 3.0 4dr	36640.00	27.00	20.00			
A6 3.0 Avant Quattro				40640.00	25.00	18.00
A6 3.0 Quattro 4dr	39640.00	25.00	18.00			
A6 4.2 Quattro 4dr	49690.00	24.00	17.00			
A8 L Quattro 4dr	69190.00	24.00	17.00			
S4 Avant Quattro				49090.00	21.00	15.00
S4 Quattro 4dr	48040.00	20.00	14.00			

SAS Output Page 2 of 4

2) Explicit Page Break from Procedure

Make BMW

Model	Type					
	Sedan			Wagon		
	MSRP	MPG (Highway)	MPG (City)	MSRP	MPG (Highway)	MPG (City)
325Ci 2dr	30795.00	29.00	20.00			
325Ci convertible 2dr	37995.00	27.00	19.00			
325i 4dr	28495.00	29.00	20.00			
325xi 4dr	30245.00	27.00	19.00			
325xi Sport				32845.00	26.00	19.00
330Ci 2dr	36995.00	30.00	20.00			
330Ci convertible 2dr	44295.00	28.00	19.00			
330i 4dr	35495.00	30.00	20.00			
330xi 4dr	37245.00	29.00	20.00			
525i 4dr	39995.00	28.00	19.00			
530i 4dr	44995.00	30.00	20.00			
545iA 4dr	54995.00	26.00	18.00			
745Li 4dr	73195.00	26.00	18.00			
745i 4dr	69195.00	26.00	18.00			

Display 10. HTML Results Displayed in Internet Explorer's Print Preview Mode

However, once the style is changed to a different style like SEASIDE, the horizontal rule disappears as shown in Display 11, but the title and footnote repeat for every logical page.

Of course, the quickest way to generate either physical or logical pages with either PDF or HTML is to use BY-group processing. Consider Display 12, a listing report that uses a BY statement added to PROC REPORT. Just remember that anytime you use BY-group processing the data must be sorted or indexed on the BY variables.

```
proc sort data=sashelp.cars out=cars;    by make type;
  where make in ('Audi', 'Volvo', 'BMW', 'Chevrolet') and
         type in ('Sedan', 'Wagon');
run;
ods html file='c:\temp\demo2_use_by.html' style=htmlblue;
ods pdf file='c:\temp\demo2_use_by.pdf' notoc style=printer;
proc report data=cars nowd;    by make type;
run;

ods _all_ close;
```

2) Explicit Page Break from Procedure

Make Audi

Model	Type					
	Sedan			Wagon		
	Mean			Mean		
	MSRP	MPG (Highway)	MPG (City)	MSRP	MPG (Highway)	MPG (City)
A4 1.8T 4dr	25940.00	31.00	22.00			
A4 3.0 4dr	31840.00	28.00	20.00			
A4 3.0 Quattro 4dr auto	34480.00	25.00	18.00			
A4 3.0 Quattro 4dr manual	33430.00	26.00	17.00			
A4 3.0 Quattro convertible 2dr	44240.00	25.00	18.00			
A4 3.0 convertible 2dr	42490.00	27.00	20.00			
A4 1.8T convertible 2dr	35940.00	30.00	23.00			
A6 2.7 Turbo Quattro 4dr	42840.00	25.00	18.00			
A6 3.0 4dr	36640.00	27.00	20.00			
A6 3.0 Avant Quattro				40840.00	25.00	18.00
A6 3.0 Quattro 4dr	39640.00	25.00	18.00			
A6 4.2 Quattro 4dr	49690.00	24.00	17.00			
A8 L Quattro 4dr	69190.00	24.00	17.00			
S4 Avant Quattro				49090.00	21.00	15.00
S4 Quattro 4dr	48040.00	20.00	14.00			

2) Explicit Page Break from Procedure

Make BMW

Model	Type					
	Sedan			Wagon		
	Mean			Mean		
	MSRP	MPG (Highway)	MPG (City)	MSRP	MPG (Highway)	MPG (City)
325Ci 2dr	30795.00	29.00	20.00			
325Ci convertible 2dr	37995.00	27.00	19.00			
325i 4dr	28495.00	29.00	20.00			
325xi 4dr	30245.00	27.00	19.00			
325xi Sport				32845.00	26.00	19.00
330Ci 2dr	36995.00	30.00	20.00			
330Ci convertible 2dr	44295.00	28.00	19.00			

Display 11. HTML Results from PROC TABULATE Using SEASIDE Style

demo2_usie_by.pdf - Adobe Reader

File Edit View Window Help

1 / 8 67%

BY Group Causes Explicit Page Breaks

Make=Audi Type=Sedan

Make	Model	Type	Origin	DT	MSRP	Invoice	Engine Size (L)	Cyl	HP	MPG (City)	MPG (Hwy)	Weight (LBS)	Wheelbase (IN)	Length (IN)
Audi	A4 1.8T 4dr	Sedan	Europe	Front	\$25,940	\$21,309	1.8	4	170	22	31	2232	104	179
Audi	A4 3.0 convertible 2dr	Sedan	Europe	Front	\$31,840	\$21,559	1.8	4	170	23	29	2620	105	180
Audi	A4 3.0 4dr	Sedan	Europe	Front	\$31,840	\$26,844	3	6	225	28	20	3402	104	179
Audi	A4 3.0 Quattro 4dr manual	Sedan	Europe	Alt	\$33,430	\$30,366	3	6	225	17	26	3505	104	179
Audi	A4 3.0 Quattro 4dr auto	Sedan	Europe	Alt	\$34,480	\$31,380	3	6	225	18	25	3457	104	179
Audi	A4 3.0 4dr	Sedan	Europe	Front	\$31,840	\$21,229	3	6	225	28	27	2562	109	182
Audi	A4 3.0 Quattro 4dr	Sedan	Europe	Alt	\$39,640	\$39,092	3	6	225	18	25	3809	109	182
Audi	A4 3.0 convertible 2dr	Sedan	Europe	Front	\$42,490	\$38,225	3	6	225	28	27	3814	109	180
Audi	A4 3.0 Quattro convertible 2dr	Sedan	Europe	Alt	\$44,240	\$41,075	3	6	225	18	25	4015	105	180
Audi	A6 2.7 Turbo Quattro 4dr	Sedan	Europe	Alt	\$42,840	\$38,040	2.7	6	250	18	25	3826	109	182
Audi	A6 4.2 Quattro 4dr	Sedan	Europe	Alt	\$49,690	\$44,920	4.2	8	300	17	24	4024	109	182
Audi	A8 L Quattro 4dr	Sedan	Europe	Alt	\$69,190	\$64,748	4.2	8	374	17	24	6289	121	204
Audi	S4 Quattro 4dr	Sedan	Europe	Alt	\$48,040	\$41,350	4.2	8	240	14	20	3820	104	178

BY Group Causes Explicit Page Breaks

Make=Audi Type=Wagon

Make	Model	Type	Origin	DT	MSRP	Invoice	Engine Size (L)	Cyl	HP	MPG (City)	MPG (Hwy)	Weight (LBS)	Wheelbase (IN)	Length (IN)
Audi	A6 3.0 Avant Quattro	Wagon	Europe	Alt	\$40,840	\$37,000	3	6	220	18	25	4025	109	182
Audi	S4 Avant Quattro	Wagon	Europe	Alt	\$49,090	\$44,440	4.2	8	240	15	21	3824	104	178

Display 12. PDF Results from PROC REPORT Using BY-Group Processing

Even if you add BY-group processing, however, remember that other factors, such as font size, margin options and orientation can still impact your output.

Another way to impact the information that appears on a page is to control the number of report columns that appear. There are several ways to do this. One way that works only for PDF (or RTF) is to use the COLUMNS= option of the PDF destination. COLUMNS= is a useful option when you have enough room on the physical page to display the report rows in multiple column format. The COLUMNS= option produces the output shown in Display 13. You use the COLUMNS= option on the ODS destination statement, as shown below.

```
ods pdf file='c:\temp\demo3_use_columns.pdf' notoc style=printer columns=2;
proc report data=cars nowd;
  column make model msrp;
  by type;
  . . . more code . . .
run;
ods _all_ close;
```

Display 13 shows how the BY group for TYPE=WAGON started a new page in the PDF results.

1a) COLUMNS= is a PDF Option			1	1a) COLUMNS= is a PDF Option			2
Type=Sedan				Type=Sedan			
Make	Model	MSRP		Make	Model	MSRP	
BMW	325i 4dr	\$38,495		Volvo	500 7.5T 4dr	\$37,885	
BMW	325Ci 2dr	\$39,795		Volvo	C70 1.9T convertible 2dr	\$40,565	
BMW	325Ci convertible 2dr	\$37,995		Volvo	C70 HPT convertible 2dr	\$42,565	
BMW	325xi 4dr	\$39,245		Volvo	500 T6 4dr	\$45,210	
BMW	330i 4dr	\$35,495					
BMW	330Ci 2dr	\$36,995					
BMW	330xi 4dr	\$37,245					
BMW	325i 4dr	\$39,995					
BMW	330Ci convertible 2dr	\$44,795					
BMW	330i 4dr	\$44,995					
BMW	545iA 4dr	\$54,995					
BMW	745i 4dr	\$69,195					
BMW	745Li 4dr	\$73,195					
Chevrolet	Aveo 4dr	\$11,690					
Chevrolet	Aveo LS 4dr hatch	\$12,585					
Chevrolet	Cavalier 2dr	\$14,610					
Chevrolet	Cavalier 4dr	\$14,810					
Chevrolet	Cavalier LS 2dr	\$16,305					
Chevrolet	Impala 4dr	\$21,900					
Chevrolet	Malibu 4dr	\$19,995					
Chevrolet	Malibu LT 4dr	\$20,335					
Type=Wagon				Type=Wagon			
Make	Model	MSRP		Make	Model	MSRP	
BMW	325xi Sport	\$32,845					
Chevrolet	Malibu Maxx LS	\$22,235					
Volvo	V40	\$36,135					
Volvo	XC70	\$35,145					

Display 13. PDF Results Using the COLUMNS= Option

The HTML destination does not support the COLUMNS= option. However, there is an ODS destination, called ODS TAGSETS.HTMLPANEL that will allow you to produce “paneled” output as shown in Display 14. TAGSETS.HTMLPANEL allows you to specify the number of “panel rows” and “panel columns” using “event” instructions

```
ods tagsets.htmlpanel style=htmlblue
  file='c:\temp\demo3_use_HTMLPANEL.html'
  options(panelcolumns='2' doc='help' panelborder='2'
    embedded_titles='No');
ods tagsets.htmlpanel event=row_panel(start);

. . . proc report code . . .
ods _all_ close;
```

1b) Use HTMLPANEL for HTML destination

Type=Sedan			Type=Wagon		
Make	Model	MSRP	Make	Model	MSRP
BMW	325i 4dr	\$28,495	BMW	325xi Sport	\$32,845
BMW	325Ci 2dr	\$30,795	Chevrolet	Malibu Maxx LS	\$22,225
BMW	325Ci convertible 2dr	\$37,995	Volvo	V40	\$26,135
BMW	325xi 4dr	\$30,245	Volvo	XC70	\$35,145
BMW	330i 4dr	\$35,495			
BMW	330Ci 2dr	\$36,995			
BMW	330xi 4dr	\$37,245			
BMW	525i 4dr	\$39,995			
BMW	330Ci convertible 2dr	\$44,295			
BMW	530i 4dr	\$44,995			
BMW	545iA 4dr	\$54,995			
BMW	745i 4dr	\$69,195			
BMW	745Li 4dr	\$73,195			
Chevrolet	Aveo 4dr	\$11,690			
Chevrolet	Aveo LS 4dr hatch	\$12,585			
Chevrolet	Cavalier 2dr	\$14,610			
Chevrolet	Cavalier 4dr	\$14,810			

Display 14. HTML Results Using the HTMLPANEL Destination

These last two examples allow you page control when your report information is narrow enough to fit multiple columns in a page. But, read on to find out what you do if your output truly is too wide.

WHAT IF MY OUTPUT IS TOO WIDE

Having output too wide is not a concern for HTML output. As mentioned above PDF has a fixed width and height, so having data too wide is a possibility. Here is a simple PROC PRINT that shows output that is too wide.

```
ods pdf file='panel.pdf' style=printer;
proc print data=sashelp.cars;
run;
ods _all_ close;
```

Obs	Make	Model	Type	Origin	DriveTrain	MSRP	Invoice	EngineSize
1	Acura	MDX	SUV	Asia	All	\$36,945	\$33,337	3.5
2	Acura	RSX Type S 2dr	Sedan	Asia	Front	\$23,820	\$21,761	2.0
3	Acura	TSX 4dr	Sedan	Asia	Front	\$26,990	\$24,647	2.4
4	Acura	TL 4dr	Sedan	Asia	Front	\$33,195	\$30,299	3.2
5	Acura	3.5 RL 4dr	Sedan	Asia	Front	\$43,755	\$39,014	3.5
6	Acura	3.5 RL w/Navigation 4dr	Sedan	Asia	Front	\$46,100	\$41,100	3.5
7	Acura	NSX coupe 2dr manual S	Sports	Asia	Rear	\$89,765	\$79,978	3.2
8	Audi	A4 1.8T 4dr	Sedan	Europe	Front	\$25,940	\$23,508	1.8
9	Audi	A4 1.8T convertible 2dr	Sedan	Europe	Front	\$35,940	\$32,506	1.8
10	Audi	A4 3.0 4dr	Sedan	Europe	Front	\$31,840	\$28,846	3.0
11	Audi	A4 3.0 Quattro 4dr manual	Sedan	Europe	All	\$33,430	\$30,366	3.0
12	Audi	A4 3.0 Quattro 4dr auto	Sedan	Europe	All	\$34,480	\$31,388	3.0
13	Audi	A6 3.0 4dr	Sedan	Europe	Front	\$36,640	\$33,129	3.0
14	Audi	A6 3.0 Quattro 4dr	Sedan	Europe	All	\$39,640	\$35,992	3.0
15	Audi	A4 3.0 convertible 2dr	Sedan	Europe	Front	\$42,490	\$38,325	3.0
16	Audi	A4 3.0 Quattro convertible 2dr	Sedan	Europe	All	\$44,240	\$40,075	3.0

Obs	Cylinders	Horsepower	MPG_City	MPG_Highway	Weight	Wheelbase	Length
1	6	265	17	23	4451	106	189
2	4	200	24	31	2778	101	172
3	4	200	22	29	3230	105	183
4	6	270	20	28	3575	108	186

Display 15. PDF Results Showing Paneled Output

As you can see, this data set contains 16 columns. ODS PDF determines that all columns will not fit in the available space. By using internal calculations it determines how many columns are within the page of output. Again I want to reiterate that when we say "page of output" that does not mean a physical page. A page of output is what ODS PDF calculates as the height and width available to show the output. When ODS PDF recognizes that 16 columns will not fit on the page, it determines how many will. A secondary calculation occurs that determines how many rows can fit on one page while still showing all the columns. The result is what we called "paneling" in ODS PDF. Looking at the above example you can see ODS PDF determined 16 observations showing all 16 columns could fit on one page. In order to fit all 16 columns, we will display a split table on the page. We display the first 9 columns of 16 observations on the top half of the page. Then we display the remaining 7 columns. You will also notice that PROC PRINT repeated the "OBS" column to assist you in matching up the split table by observation. Not only did our SAS code produce output too wide, it is also too tall, so we will have some pagination.

10	6	220	20	28	3462	104	179
11	6	220	17	26	3583	104	179
12	6	220	18	25	3627	104	179
13	6	220	20	27	3561	109	192
14	6	220	18	25	3880	109	192
15	6	220	20	27	3814	105	180
16	6	220	18	25	4013	105	180

PDF results showing Paneled output

2

Obs	Make	Model	Type	Origin	DriveTrain	MSRP	Invoice	EngineSize
17	Audi	A6 2.7 Turbo Quattro 4dr	Sedan	Europe	All	\$42,840	\$38,840	2.7
18	Audi	A6 4.2 Quattro 4dr	Sedan	Europe	All	\$49,690	\$44,936	4.2
19	Audi	A8 L Quattro 4dr	Sedan	Europe	All	\$69,190	\$64,740	4.2
20	Audi	S4 Quattro 4dr	Sedan	Europe	All	\$48,040	\$43,556	4.2
21	Audi	RS 6 4dr	Sports	Europe	Front	\$84,600	\$76,417	4.2
22	Audi	TT 1.8 convertible 2dr (coupe)	Sports	Europe	Front	\$35,940	\$32,512	1.8
23	Audi	TT 1.8 Quattro 2dr (convertible)	Sports	Europe	All	\$37,390	\$33,891	1.8

Display 16. Continued PDF Results Showing Paneled Output

Looking at output you can see how the bottom of page 1 completes the first 16 observations of the table. ODS PDF will start the next 16 observations on page 2. This will continue until all observations of the table are complete. In the above example we got 27 pages of output. Paneled output normally is not the preferred way of viewing your data. It can get confusing and cumbersome for your data to be fragmented and displayed over many pages.

If you **do** want to control paneling, you have to look to procedures like PROC REPORT to help you in this regard. You can control which variable starts the new panel and which variables repeat across panels through a use of the ID option and the PAGE option. For example, if you use the code below:

```
proc report data= cars nowd spanrows
    style(report)={width=100%};
title '1) Explicit Paneling';
column Make Type Model Origin DriveTrain MSRP Invoice EngineSize Cylinders
    Horsepower MPG_City MPG_Highway Weight Wheelbase Length;
define make / order id;
define type / order id;
define Cylinders / page;
break after make/page ;
run;
```

In this code, the values of MAKE and TYPE will repeat for every panel and the first panel will be composed of all the variables starting with MAKE and ending with ENGINESIZE; the second panel will contain MAKE and TYPE (repeated from the first panel), followed by the variables CYLINDERS through LENGTH, as listed in the COLUMN statement. This technique works for PDF, as shown in Display ~17x~.

The screenshot shows two pages of a PDF report generated by PROC REPORT. The first page (page 7) displays a table with columns: Make, Type, Model, Origin, DriveTrain, MSRP, Invoice, and Engine Size (L). The second page (page 8) displays a table with columns: Make, Type, Cylinders, Horsepower, MPG (City), MPG (Highway), Weight (LBS), Wheelbase (IN), and Length (IN). Both tables contain data for Mazda vehicles, with the first page showing the first 16 observations and the second page showing the next 16 observations.

Display 17. Explicit Paneling with PROC REPORT in PDF Output

What you notice in the PDF output is how the WIDTH=100% style override made both panels stretch from margin to margin. Then, the ID option in the DEFINE statement ensured that MAKE and TYPE were repeated for every panel. Finally, the PAGE option in the DEFINE statement for CYLINDERS caused the second panel to start with the CYLINDERS variable after the repeat of MAKE and TYPE. Notice how each panel appears on a separate page. We can control this using the STARTPAGE= option, by adding STARTPAGE=NO to the ODS PDF invocation statement. (In our program, we used a WHERE statement to limit the MAKE value to Mazda, in order to generate only one page of output, as shown in Display ~18x~.

```
ods pdf file='c:\temp\demo4_explicit_panel_startpage.pdf' style=printer
    startpage=no;
proc report data= cars nowd spanrows style(report)={width=100%};
where make = 'Mazda';
. . . rest of code same as previous code . . .
ods _all_ close;
```

The HTML output that is created using PROC REPORT and explicit paneling only can be compared to Display ~17x~ because HTML is not a "paged" destination and there is no STARTPAGE option available for the HTML destination. However, you can see in Display ~19x~, that the panels in the HTML output are divided onto logical "pages" which we can see by the use of the horizontal rule between each panel.

demo4_explicit_panel_startpage.pdf - Adobe Acrobat

File Edit View Document Comments Forms Tools Advanced Window Help

Create Combine Comment

1 / 1

49.4%

Find

2) Explicit Paneling with STARTPAGE for PDF

Make	Type	Model	Origin	DriveTrain	MSRP	Invoice	Engine Size (L)
Mazda	SUV	Tribute DN 2.0	Asia	All	\$21,087	\$19,742	2
	Sedan	MPV ES	Asia	Front	\$28,750	\$26,600	3
		Mazda3 i 4dr	Asia	Front	\$15,500	\$14,525	2
		Mazda3 s 4dr	Asia	Front	\$17,200	\$15,922	2.3
		Mazda6 i 4dr	Asia	Front	\$19,270	\$17,817	2.3
	Sports	MX-5 Miata LS convertible 2dr	Asia	Rear	\$25,193	\$23,285	1.8
		MX-5 Miata convertible 2dr	Asia	Rear	\$22,388	\$20,701	1.8
		RX-8 4dr automatic	Asia	Rear	\$25,700	\$23,794	1.3
		RX-8 4dr manual	Asia	Rear	\$27,200	\$25,179	1.3
	Truck	B2300 SX Regular Cab	Asia	Rear	\$14,840	\$14,070	2.3
		B4000 SE Cab Plus	Asia	All	\$22,350	\$20,482	4

Make	Type	Cylinders	Horsepower	MPG (City)	MPG (Highway)	Weight (LBS)	Wheelbase (IN)	Length (IN)
Mazda	SUV	4	130	22	25	3091	103	173
	Sedan	6	200	18	25	3812	112	188
		4	148	26	34	2696	104	178
		4	160	25	31	2762	104	179
		4	160	24	32	3042	105	187
	Sports	4	142	23	28	2387	89	156
		4	142	23	28	2387	89	156
		.	197	18	25	3053	106	174
		.	238	18	24	3029	106	174
	Truck	4	143	24	29	2960	112	188
		6	207	15	19	3571	126	203

Display 18. Explicit Paneling with PROC REPORT in PDF Output Using STARTPAGE=NO

Make	Type	Cylinders	Horsepower	MPG (City)	MPG (Highway)	Weight (LBS)	Wheelbase (IN)	Length (IN)
Mazda	SUV	4	130	22	25	3091	103	173
	Sedan	6	200	18	25	3812	112	188
		4	148	26	34	2696	104	178
		4	160	25	31	2762	104	179
		4	160	24	32	3042	105	187
	Sports	4	142	23	28	2387	89	156
		4	142	23	28	2387	89	156
			197	18	25	3053	106	174
			238	18	24	3029	106	174
	Truck	4	143	24	29	2960	112	188
		6	207	15	19	3571	126	203

Display 19. Explicit Paneling with PROC REPORT in HTML Output

While explicit paneling can also help if you have too many variables to fit on a printed page in the PDF destination, the fact still remains that paneled output is not always the best way to display your data. It might be that some of the changes we have already shown will be suitable for your purpose better.

If only ODS PDF had scroll bars like ODS HTML has to view the output. The PDF viewer (Acrobat Reader) does have a scroll bar, but ODS PDF still is creating an image of the report, as it will look if printed. So, if you do not want explicit paneling, how do we prevent paneling in ODS PDF? Above we talked about font size, cell padding, orientation, and margins to control our output. These are all great ideas but sometimes you might have to alter your output to not display all the available columns. Since paneling and controlling "pages" does not really impact HTML the way it impacts PDF, it's time to move on to other ways to control how your output looks.

HOW TO CONTROL THE OUTPUT

In this section, we are going to talk about controlling the output from two different standpoints: controlling the output with destination-specific options (this will mostly apply to ODS PDF) and controlling the output with style-specific options (this will apply to both ODS HTML and ODS PDF).

Earlier we discussed explicit and implicit page breaks. There is another way to create explicit breaks that you can insert yourself but only with the ODS PDF destination. The attribute is called STARTPAGE and as we have seen in the simple example above, it can easily turn off all page breaks. But, there are other values for STARTPAGE other than STARTPAGE=NO. In the ODS PDF statement you can enter STARTPAGE=YES|NO|NOW.

Using the option YES (which is the default) after every procedure block an explicit page break will be issued. Using the option NO means that no page break will be issued and ODS PDF will try to fit as much as possible until an implicit page break occurs. The last option is NOW, which issues an immediate page break at your request. If you started your output with STARTPAGE=NO in order to get the two panels for Mazda on one physical page, then you might use STARTPAGE=NOW to issue page breaks where you need them, such as before you generate the two panels for Volvo. This type of code can be easily "macro-ized" so that the panels look like multiple tables produced for all the variables within a group.

The code that produced Display ~20x~ shows using STARTPAGE=NOW to produce 2 pages of paneled output.

```
ods pdf file='demo4_panel_more_startpage.pdf' style=printer startpage=no;
** PROC REPORT for Mazda;
ods pdf startpage=now;
** PROC REPORT for Volvo;
ods _all_ close;
```

demo4_panel_more_startpage.pdf - Adobe Acrobat

File Edit View Document Comments Forms Tools Advanced Window Help

Sticky Note Text Edits Show

Create Combine Collaborate Secure Sign Forms Comment

1 / 2 54.8% Find

3a) Explicit Paneling with STARTPAGE for PDF

1

Make	Type	Model	Origin	DriveTrain	MSRP	Invoice	Engine Size (L)
Mazda	SUV	Tribute DXN 2.0	Asia	All	\$21,007	\$19,742	2
	Sedan	MPV ES	Asia	Front	\$20,750	\$20,600	3
		Mazda3 i 4dr	Asia	Front	\$15,500	\$14,525	2
		Mazda3 s 4dr	Asia	Front	\$17,200	\$15,922	2.3
		Mazda3 s 4dr	Asia	Front	\$19,270	\$17,817	2.3
	Sports	MX-5 Miata LS convertible 2dr	Asia	Rear	\$25,193	\$23,285	1.8
		MX-5 Miata convertible 2dr	Asia	Rear	\$22,300	\$20,791	1.8
		MX-5 4dr sun-moat	Asia	Rear	\$25,700	\$23,794	1.3
		MX-5 4dr manual	Asia	Rear	\$27,200	\$25,179	1.3
	Truck	B2300 BX Regular Cab	Asia	Rear	\$14,940	\$14,070	2.3
		B4000 SE Cab Plus	Asia	All	\$22,350	\$20,482	4

Make	Type	Cylinders	Horsepower	MPG (City)	MPG (Highway)	Weight (LBS)	Wheelbase (IN)	Length (IN)
Mazda	SUV	4	130	22	25	3091	103	173
	Sedan	6	200	18	25	3012	112	180
		4	148	26	34	2696	104	170
		4	180	23	31	2762	104	179
		4	180	24	32	3042	105	187
	Sports	4	142	23	28	2387	89	156
		4	142	23	28	2387	89	156
		197	18	25	3053	104	174	
		238	18	24	3029	104	174	
	Truck	4	143	24	29	2960	112	180
		6	207	15	19	3571	124	203

3b) Explicit Paneling with STARTPAGE for PDF

2

Make	Type	Model	Origin	DriveTrain	MSRP	Invoice	Engine Size (L)
Vaux	SUV	XC70 T6	Europe	All	\$41,250	\$38,051	2.8
	Sedan	C70 HPT convertible 2dr	Europe	Front	\$42,545	\$40,003	2.3
		C70 LPT convertible 2dr	Europe	Front	\$40,505	\$38,203	2.4
		S40 4dr	Europe	Front	\$25,135	\$23,701	1.9
		S60 2.5 4dr	Europe	All	\$31,745	\$29,916	2.5
		S60 R 4dr	Europe	All	\$37,560	\$35,362	2.5
		S60 T5 4dr	Europe	Front	\$34,045	\$32,962	2.3
		S80 2.5T 4dr	Europe	All	\$37,005	\$35,400	2.5
		S80 2.9 4dr	Europe	Front	\$37,730	\$35,542	2.9
		S80 T6 4dr	Europe	Front	\$45,210	\$42,873	2.9
Wagon	V40	Europe	Front	\$26,135	\$24,641	1.9	
	XC70	Europe	All	\$35,145	\$33,112	2.5	

Make	Type	Cylinders	Horsepower	MPG (City)	MPG (Highway)	Weight (LBS)	Wheelbase (IN)	Length (IN)
Vaux	SUV	6	265	15	20	4630	113	189
	Sedan	5	242	20	26	3450	105	186
		5	197	21	28	3450	105	186
		4	170	22	29	2767	101	178
		5	208	20	27	3943	107	180
		5	300	18	25	3571	107	181
		5	247	20	26	3766	107	180
		5	194	20	27	3691	110	190
		6	205	20	26	3576	110	190
		6	265	19	24	3453	110	190
Wagon	4	170	22	29	2022	101	180	
	5	205	20	27	3023	109	186	

Display 20. Explicit Paneling with PROC REPORT in HTML Output

But a more in depth discussion of PROC REPORT for paneling falls outside the scope of this paper. Suffice it to say that beyond implicit and explicit paneling, there is even more in the world of ODS to allow you to control placement of ODS objects in a specific layout using ODS LAYOUT. But, ODS LAYOUT is a very specialized layout tool and does not work for ODS HTML at all. Now we can move on to a discussion of what can be controlled using style templates and style overrides.

Of course, the best way to control output look and feel is with ODS style templates. For example, if both ODS PDF and ODS HTML use the JOURNAL style, we see the results as shown in Display ~21x~. The JOURNAL style is a great black and white style that is designed for journal publication. It has a crisp "no interior lines" table look with black borders between the header columns and the data cell area.

Generally speaking, color, and font changes will end up making HTML and PDF output look the same, as this style override in PROC REPORT shows.

```
proc report data=sashelp.cars nowd
  style(Header)={font_face='Albany AMT' font_size=12pt
                  color=cx004c1a background=cxccccff}
  style(column)={font_face='Albany AMT' font_size=11pt}
  style(summary)={font_face='Albany AMT' font_size=14pt
                  color=cx004c1a background=cxccccff};
  column make model ('Average' msrp invoice mpg_city mpg_highway);
  define make / group
    style(column)=Header{font_face='Albany AMT' font_size=12pt
                        color=cx004c1a background=cxccccff};
  define model / group
```



```

style(column)=Header{font_face='Albany AMT' font_size=12pt
                    color=cx004c1a background=cxccccff};

define msrp / mean;
define invoice / mean;
define mpg_city/mean f=comma6.;
define mpg_highway / mean f=comma6.;
rbreak after / summarize;
run;

```

These style override methods in PROC REPORT have the same effect on both outputs, as shown in Display ~22x~.

Make	Model	Type	MSRP	Invoice
Acura	MDX	SUV	\$36,945	\$33,337
Acura	RSX Type S 2dr	Sedan	\$23,820	\$21,761
Acura	TSX 4dr	Sedan	\$26,990	\$24,647
Acura	TL 4dr	Sedan	\$33,195	\$30,299
Acura	3.5 RL 4dr	Sedan	\$43,755	\$39,014
Acura	3.5 RL w/Navigation 4dr	Sedan	\$46,100	\$41,100
Acura	NSX coupe 2dr manual S	Sports	\$89,765	\$79,978
Audi	A4 1.8T 4dr	Sedan	\$25,940	\$23,508
Audi	A4 1.8T convertible 2dr	Sedan	\$35,940	\$32,506
Audi	A4 3.0 4dr	Sedan	\$31,840	\$28,846
Audi	A4 3.0 Quattro 4dr manual	Sedan	\$33,430	\$30,366
Audi	A4 3.0 Quattro 4dr auto	Sedan	\$34,480	\$31,388
Audi	A6 3.0 4dr	Sedan	\$36,640	\$33,129
Audi	A6 3.0 Quattro 4dr	Sedan	\$39,640	\$35,992
Audi	A4 3.0 convertible 2dr	Sedan	\$42,490	\$38,325
Audi	A4 3.0 Quattro convertible 2dr	Sedan	\$44,240	\$40,075
Audi	A6 2.7 Turbo Quattro 4dr	Sedan	\$42,840	\$38,840
Audi	A6 4.2 Quattro 4dr	Sedan	\$49,690	\$44,936
Audi	A8 L Quattro 4dr	Sedan	\$69,190	\$64,740
Audi	S4 Quattro 4dr	Sedan	\$48,040	\$43,556
Audi	RS 6 4dr	Sports	\$84,600	\$76,417
Audi	TT 1.8 convertible 2dr (coupe)	Sports	\$35,940	\$32,512
Audi	TT 1.8 Quattro 2dr (convertible)	Sports	\$37,390	\$33,891
Audi	TT 3.2 coupe 2dr (convertible)	Sports	\$40,590	\$36,739
Audi	A6 3.0 Avant Quattro	Wagon	\$40,840	\$37,060

Display 21. Using the Same Style Template for PDF and HTML Output

Make	Model	MSRP	Invoice	Average MPG (City)	Average MPG (Highway)
Acura	3.5 RL 4dr	\$43,755	\$39,014	18	24
Acura	3.5 RL w/Navigation 4dr	\$46,100	\$41,100	18	24
Acura	NSX coupe 2dr manual S	\$89,765	\$79,978	17	24
Audi	A4 3.0 Quattro convertible 2dr	\$44,240	\$40,075	18	25
Audi	A4 3.0 convertible 2dr	\$42,490	\$38,325	20	27
Audi	A6 2.7 Turbo Quattro 4dr	\$42,840	\$38,840	18	25
Audi	A6 3.0 Avant Quattro	\$40,840	\$37,060	18	25
Audi	A6 4.2 Quattro 4dr	\$49,690	\$44,936	17	24
Audi	A8 L Quattro 4dr	\$69,190	\$64,740	17	24
Audi	RS 6 4dr	\$84,600	\$76,417	15	22
Audi	S4 Avant Quattro	\$49,090	\$44,446	15	21
Audi	S4 Quattro 4dr	\$48,040	\$43,556	14	20
Audi	TT 3.2 coupe 2dr (convertible)	\$40,590	\$36,739	21	29
		\$53,172	\$48,094	17	24

Display 22. Using STYLE= Statement-level Overrides for PDF and HTML Output

One challenge that people always worry about is how to make cells a specific height or width and another challenge is how to flow long text within a cell. As it turns out, all of these challenges can be met by using STYLE overrides. The CELLHEIGHT attribute is good when you want to call attention to one report row in particular.

The code to use CELLHEIGHT is illustrated in this example with PROC REPORT, changes to the previous program were minimal, so only the changed statements are shown below.

```
proc report data=sashelp.cars nowd
. . . more code . . .
  style(summary)={font_face='Albany AMT' font_size=14pt
                    color=cx004c1a background=cxcccff
                    cellheight=1.25in vjust=m};
. . . more code . . .
compute msrp;
  if msrp.mean gt 80000 then do;
    call define(_col_,'style','style={cellheight=.75in vjust=m}');
  end;
```

1) Change CELLHEIGHT		Average			
Make	Model	MSRP	Invoice	MPG (City)	MPG (Highway)
Acura	3.5 RL 4dr	\$43,755	\$39,014	18	24
	3.5 RL w/Navigation 4dr	\$46,100	\$41,100	18	24
	NSX coupe 2dr manual S	\$89,765	\$79,978	17	24
Audi	A4 3.0 Quattro convertible 2dr	\$44,240	\$40,075	18	25
	A4 3.0 convertible 2dr	\$42,490	\$38,325	20	27
	A6 2.7 Turbo Quattro 4dr	\$42,840	\$38,840	18	25
	A6 3.0 Avant Quattro	\$40,840	\$37,060	18	25
	A6 4.2 Quattro 4dr	\$49,690	\$44,936	17	24
	A8 L Quattro 4dr	\$69,190	\$64,740	17	24
	RS 6 4dr	\$84,600	\$76,417	15	22
	S4 Avant Quattro	\$49,090	\$44,446	15	21
TT	S4 Quattro 4dr	\$48,040	\$43,556	14	20
	TT 3.2 coupe 2dr (convertible)	\$40,590	\$36,739	21	29
		\$53,172	\$48,094	17	24

Display 23. Changing CELLHEIGHT for PDF and HTML Output

The output shown in Display ~23x~ shows that both destinations treated the SUMMARY line as expected, with an increased CELLHEIGHT and a vertical alignment of the middle of the cell. However, what we see for the result of the CALL DEFINE conditional style change is that both destinations will not make just one cell in a row higher than the cells in the rest of the row. So the CELLHEIGHT change impacted the entire row. However, PDF treated the VJUST=M override as an instruction to impact the whole row. While for the HTML destination, the VJUST=M had an impact on only the MSRP cell, as coded. So, if all we do is change the first argument in the CALL DEFINE from _COL_ to _ROW_, then we see that both destinations treat all the cells in the report row the same, as shown in Display ~24x~.

1) Change CELLHEIGHT with _ROW_

Make	Model	Average			
		MSRP	Invoice	MPG (City)	MPG (Highway)
Acura	3.5 RL 4dr	\$43,755	\$39,014	18	24
	3.5 RL w/Navigation 4dr	\$46,100	\$41,100	18	24
	NSX coupe 2dr manual 5	\$89,765	\$79,978	17	24
Audi	A4 3.0 Quattro convertible 2dr	\$44,240	\$40,075	18	25
	A4 3.0 convertible 2dr	\$42,490	\$38,325	20	27
	A6 2.7 Turbo Quattro 4dr	\$42,840	\$38,840	18	25
	A6 3.0 Avant Quattro	\$40,840	\$37,060	18	25
	A6 4.2 Quattro 4dr	\$49,690	\$44,936	17	24
	A8 L Quattro 4dr	\$69,190	\$64,740	17	24
	RS 6 4dr	\$84,600	\$76,417	15	22
	S4 Avant Quattro	\$49,090	\$44,446	15	21
	S4 Quattro 4dr	\$48,040	\$43,556	14	20
	TT 3.2 coupe 2dr (convertible)	\$40,590	\$36,739	21	29
		\$53,172	\$48,094	17	24

1) Change CELLHEIGHT with _ROW_

Make	Model	Average			
		MSRP	Invoice	MPG (City)	MPG (Highway)
Acura	3.5 RL 4dr	\$43,755	\$39,014	18	24
	3.5 RL w/Navigation 4dr	\$46,100	\$41,100	18	24
	NSX coupe 2dr manual 5	\$89,765	\$79,978	17	24
Audi	A4 3.0 Quattro convertible 2dr	\$44,240	\$40,075	18	25
	A4 3.0 convertible 2dr	\$42,490	\$38,325	20	27
	A6 2.7 Turbo Quattro 4dr	\$42,840	\$38,840	18	25
	A6 3.0 Avant Quattro	\$40,840	\$37,060	18	25
	A6 4.2 Quattro 4dr	\$49,690	\$44,936	17	24
	A8 L Quattro 4dr	\$69,190	\$64,740	17	24
	RS 6 4dr	\$84,600	\$76,417	15	22
	S4 Avant Quattro	\$49,090	\$44,446	15	21
	S4 Quattro 4dr	\$48,040	\$43,556	14	20
	TT 3.2 coupe 2dr (convertible)	\$40,590	\$36,739	21	29
		\$53,172	\$48,094	17	24

Display 24. Changing Entire Row Height and Vertical Alignment

Sometimes people assume that when they have very long text that the solution is the CELLHEIGHT style attribute, but that is not the case. With ODS, the height of a cell will increase to automatically fit what is inside the cell. You can actually impact the height of a cell that is going to contain long text by adjusting the WIDTH or CELLWIDTH attribute. One thing about both PDF and HTML is that the destinations are smart enough to flow or wrap text within a specified cell width. For the next example, SASHELP.CARS has been modified to have some phrases by Lewis Carroll added to some of the values for the MODEL variable. You can see the full code when you download the ZIP file of programs.

The program that alters the cell width attribute. Note that you need to use CELLWIDTH= for version 9.1.3 or earlier and you can use WIDTH= or CELLWIDTH= in version 9.2 or later.

```
proc report data=cars nowd;
  title '1a) Narrow Width Causes Wrapping That Influences Cell Height';
  column make model type msrp invoice;
  define model / style(column)={width=1.0in};
run;

proc report data=cars nowd;
  title '1b) Making Model Column Wider Changes Wrapping and Height';
  column make model type msrp invoice;
  define model / style(column)={width=2.0in};
run;
```

Notice how, in Display ~25x~, the cell heights for all the data cells have automatically adjusted to accommodate the defined width of 1". But in Display ~26x~, which was the result of only changing the CELLWIDTH attribute to 2", some of the higher cells in the first example now have enough width to "spread out". Therefore, the cells are not as high in report 1a as they were in report 1b.

You might have noticed that some of our code uses WIDTH versus CELLWIDTH or HEIGHT versus CELLHEIGHT or OUTPUTWIDTH versus WIDTH. The fact is that any version of SAS after 9.2 will use either attribute name. This is true of other attributes, such as FOREGROUND versus COLOR and BACKGROUND versus BACKGROUND_COLOR. But if you are still using version 9.1.3 or earlier, then you need to use the original style attribute, as documented for your version of SAS. The relevant documentation topic is entitled, "Style Attributes and Their Values."

1a) Narrow Width Causes Wrapping That Influences Cell Height

Make	Model	Type	MSRP	Invoice
Acura	3.5 RL 4dr	Sedan	\$43,755	\$39,014
Acura	3.5 RL w/Navigation 4dr	Sedan	\$46,100	\$41,100
Acura	NSX coupe 2dr manual S Twas brillig and the slithy toves, did gyre and gimble in the wabe. All mimsy were the borogroves, and the mome raths outgrabe.	Sports	\$89,765	\$79,978
Audi	A4 3.0 convertible 2dr Twas brillig and the slithy toves, did gyre and gimble in the wabe. All mimsy were the borogroves, and the mome raths outgrabe.	Sedan	\$42,490	\$38,325
Audi	A4 3.0 Quattro convertible 2dr Twas brillig and the slithy toves, did gyre and gimble in the wabe. All mimsy were the borogroves, and the mome raths outgrabe.	Sedan	\$44,240	\$40,075
Audi	A6 2.7 Turbo Quattro 4dr	Sedan	\$42,840	\$38,840
Audi	A6 4.2 Quattro 4dr	Sedan	\$49,690	\$44,936

Display 25. Changing Cell Width to “Flow” Long Text

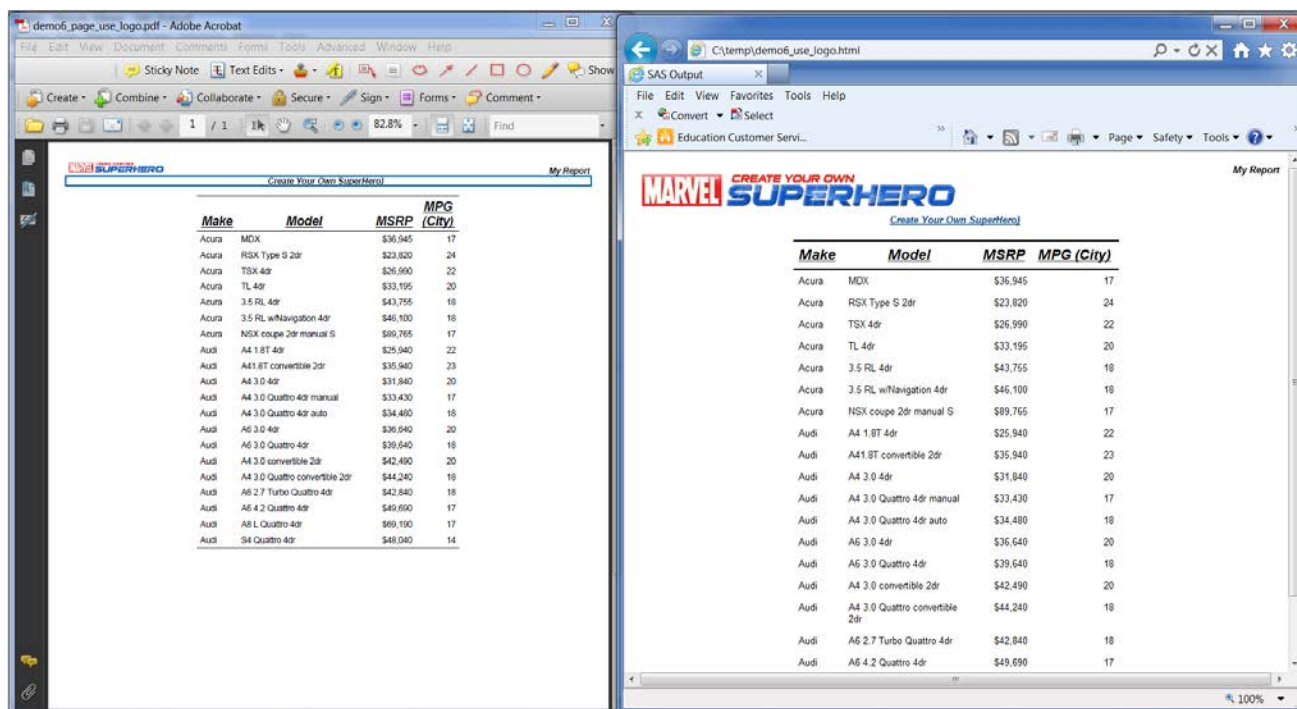
1b) Making Model Column Wider Changes Wrapping and Height

Make	Model	Type	MSRP	Invoice
Acura	3.5 RL 4dr	Sedan	\$43,755	\$39,014
Acura	3.5 RL w/Navigation 4dr	Sedan	\$46,100	\$41,100
Acura	NSX coupe 2dr manual S Twas brillig and the slithy toves, did gyre and gimble in the wabe. All mimsy were the borogroves, and the mome raths outgrabe.	Sports	\$89,765	\$79,978
Audi	A4 3.0 convertible 2dr Twas brillig and the slithy toves, did gyre and gimble in the wabe. All mimsy were the borogroves, and the mome raths outgrabe.	Sedan	\$42,490	\$38,325
Audi	A4 3.0 Quattro convertible 2dr Twas brillig and the slithy toves, did gyre and gimble in the wabe. All mimsy were the borogroves, and the mome raths outgrabe.	Sedan	\$44,240	\$40,075
Audi	A6 2.7 Turbo Quattro 4dr	Sedan	\$42,840	\$38,840
Audi	A6 4.2 Quattro 4dr	Sedan	\$49,690	\$44,936
Audi	A8 L Quattro 4dr	Sedan	\$69,190	\$64,740
Audi	S4 Quattro 4dr	Sedan	\$48,040	\$43,556
Audi	RS 6 4dr	Sports	\$84,600	\$76,417
Audi	TT 3.2 coupe 2dr (convertible)	Sports	\$40,590	\$36,739
Audi	A6 3.0 Avant Quattro	Wagon	\$40,840	\$37,060
Audi	S4 Avant Quattro	Wagon	\$49,090	\$44,446

Display 26. Adjusting Width Impacts Cell Height

One of the last style changes we will show is a new (with SAS 9.2) style attribute that can be used for text decoration to underline, overline, or strike through text. For example, when people use the JOURNAL style, they frequently want to underline the column header values. In the past, this was something that you could do only with RTF or HTML using destination-specific syntax. However, starting with SAS 9.2, using the TEXTDECORATION style attribute, this feature became available for PDF, as well as the other destinations. Consider the following code that produced Display ~27x~. Not only does it underline the column headers, but it also uses the LINK= option of the SAS TITLE statement to insert a hyperlink into the REPORT and ODS ESCAPECHAR with the PREIMAGE attribute to insert a logo into the report.

```
ods escapechar='^';
proc report data=sashelp.cars(obs=20) nowd
  style(report)={width=50%}
  style(header)={fontweight=bold fontsize=14pt textdecoration=underline};
  where substr(make,1,1) in ('A','B','C');
  title j=1 '^{'style[preimage="c:\sgf2014\create_superhero.png"] }'
    j=r 'My Report';
  title2 link="http://marvel.com/games/play/31/create_your_own_superhero"
    'Create Your Own SuperHero}';
  column make model msrp mpg_city;
run;
```



Display 27. Using ODS ESCAPECHAR and Other Style Changes

What you will notice is that the hyperlink for the Marvel.com site is highlighted differently in PDF (with a blue box) versus HTML, but both links, when clicked, will take you to a website where you can create your own superhero. For this paper, Cynthia was HTMLWoman and Scott was PDF Guy.

The final word on controlling your output is that it will all come down to style in the end. Whether you use style templates, style overrides or ODS ESCAPECHAR, it will be worth your time to invest in a study of ODS styles. Last, but certainly not least, we are going to focus on our two destinations and how they work with non-tabular output.

NON TABULAR OUTPUT

There are other types of output other than tables in your output. We have mentioned Titles and Footnotes. But there are other types of non-tabular output like images, SAS/GRAPH, and ODS GRAPHICS we have to think about when designing our output. Each destination handles these differently. We will discuss images first. Three very common

ways to use images are as a background, pre-image, and post-image. In the ODS PDF destination the DPI (dot per inch) of the image determines the size that will be displayed. It's good to remember the higher the image DPI the more precise the appearance of the image. Here is an example using the same image with different image DPIs.

```
ods escapechar = '^';
title '^{style [preimage="c:\sgf2014\SAS_logo150.jpg"] my title w/SAS @ 150 DPI}';
ods pdf file="prt_image.pdf";
proc print data=sashelp.class(obs=1);run;
ods text = '^{style [preimage="c:\sgf2014\SAS_logo100.jpg"] text statement w/SAS @ 100 DPI}';
ods _all_ close;
```



my title w/SAS @ 150 DPI

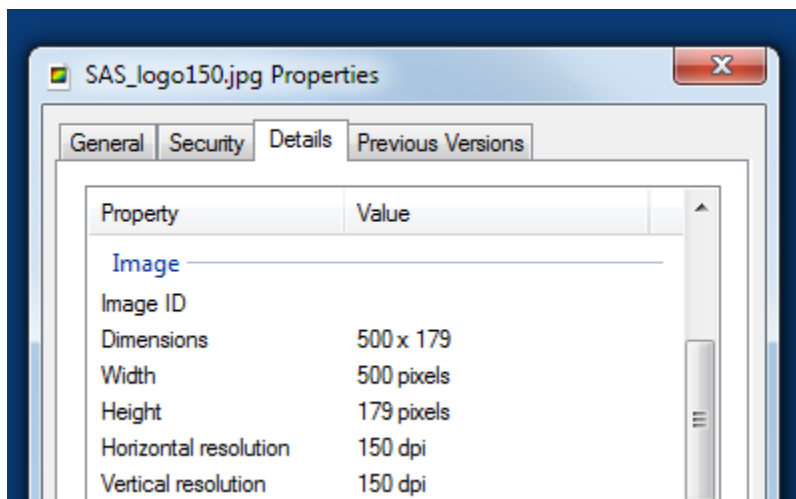
Obs	Name	Sex	Age	Height	Weight
1	Alfred	M	14	69	112.5



text statement w/SAS @ 100 DPI

Display 28. Images Displayed within ODS PDF Destination

Notice the pre-image on the title statement has image DPI of 150. The image on the text statement below the table has image DPI of 100. Understanding the different sizes of the images will help you plan the space you use on the page. Using the properties of the image file will tell you initial size of the image.



Display 29. Image Property File

The properties of the image will show you the height, width, and DPI of the image. Using simple math you can divide the height and width by the DPI to calculate its size.

Width will be $500 / 150 = 3.33$ inches

Height will be $170 / 150 = 1.19$ inches

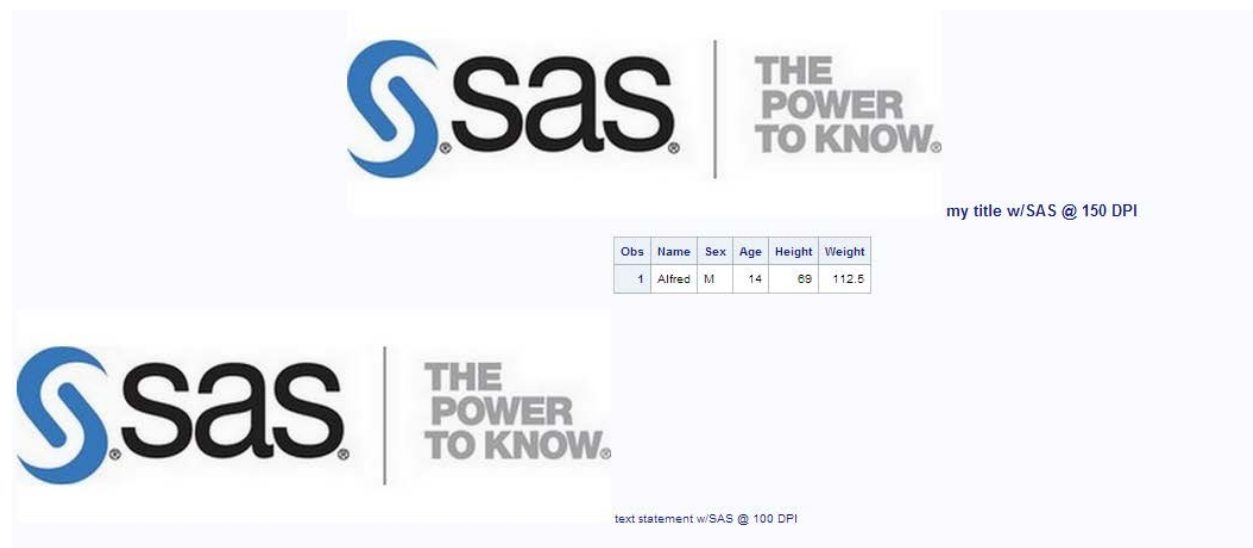
The other image had an image DPI of 100. That image has a larger height and width since its divisor is only 100 rather than 150.

ODS PDF has a default DPI equal to 150. When ODS PDF encounters an image that is not 150 DPI, it will perform some internal calculations to scale the image. The second image will grow bigger since the image DPI was less than the PDF's default DPI. A good guideline is to always use images with a DPI of 150 to ensure no surprises occur when placing an image inside your PDF file. Of course using images with multiple image DPIs within one output will result in scaling.

If you cannot change your image DPI, you can always set the default DPI for ODS PDF by using the "DPI=n" (where n is the DPI of your choice) attribute on your ODS PDF statement.

```
ods pdf file="prt_image.pdf" dpi=100;
```

Now ODS HTML handles images much differently than ODS PDF. Here is the output of ODS HTML using the same program.

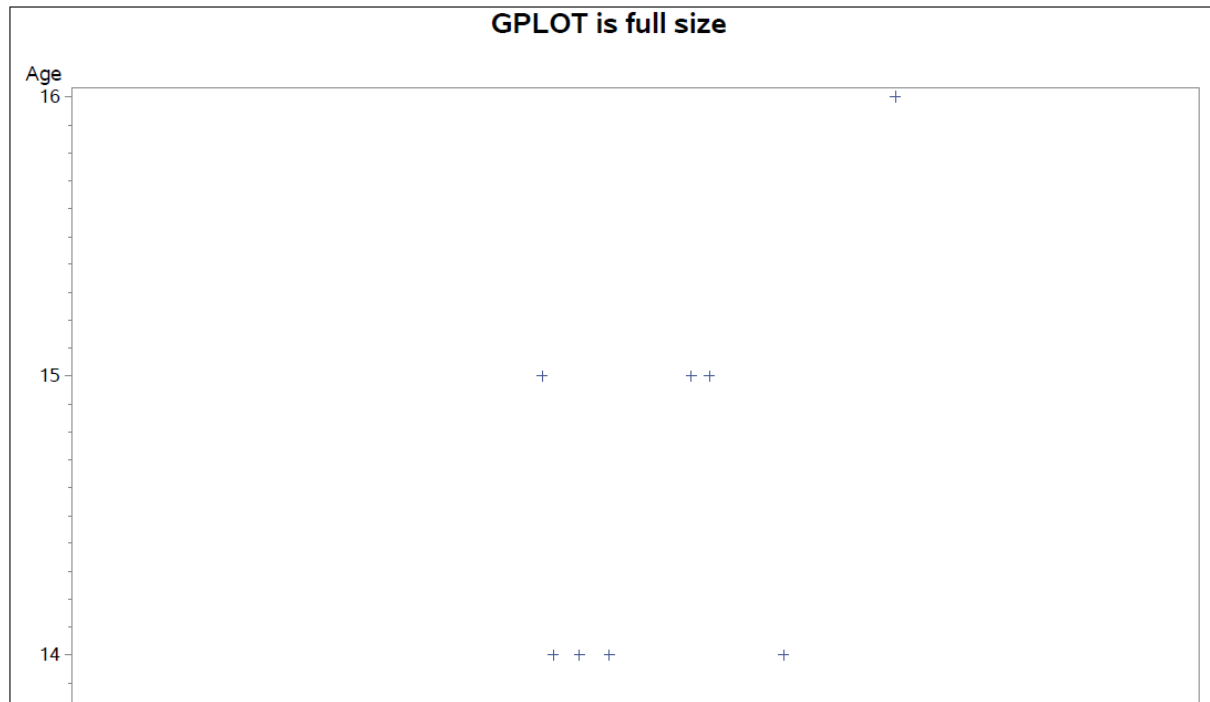


Display 30. Images Displayed within ODS HTML Destination

The image is exactly the same no matter what the image DPI is. HTML does not worry about image DPI. It knows the height and width from the properties of the image. Both images are exactly 500 pixels x 179 pixels. Monitors have PPI (pixels per inch), so the 'per inch' does not consider DPI. It is just displaying the pixels value, which is the same regardless of DPI. If the monitor has a higher PPI, the image might be smaller. The DPI only affects the clarity of the image, not the size.

There are other non-tabular outputs like SAS/GRAPH and ODS GRAPHICS. Again both destinations handle them in different ways. ODS HTML handles graphs very similar in the way it handles images. The graph is saved off in an image format and HTML just references that file when needed. ODS PDF embeds the graph inside the PDF file. The sizing is dependent on the graph sizing options. Now, we can look at SAS/GRAPH and see how it appears in ODS PDF. Here is a simple PROC GPLOT.

```
options reset=all border;
title 'notice graph is full size of page';
ods pdf file="graph.pdf" notoc;
  proc gplot data=sashelp.class; plot age*height; run;
ods pdf close;
```



Display 31. SAS/GRAPH Displayed within ODS PDF Destination

I did not show the whole page but the GPLOT does fit the entire page in ODS PDF. Graph objects are embedded into the PDF. ODS PDF and the GRAPH subsystem communicate about available space before placing down the output. Graph with no size setting will always want a full page for the output. Now we can do another example with a table and graph on same page.

```
options number;
options reset=all border;
title 'GPLOT is full size';
ods pdf file="graph.pdf" notoc startpage=no;
proc print data=sashelp.class(obs=5); run;
  proc gplot data=sashelp.class; plot age*height; run;
ods pdf close;
```

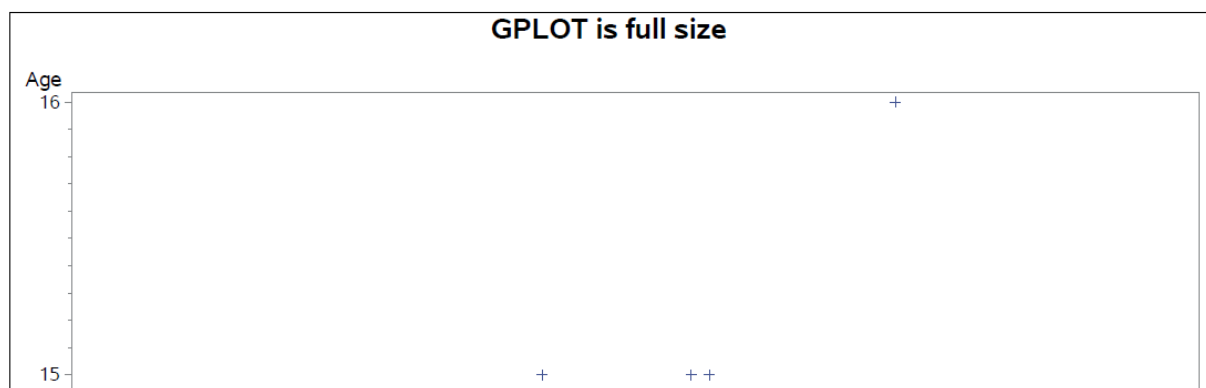
GPLOT is full size

1

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

Display 32. SAS/GRAPH Displayed within ODS PDF Destination (page 1 of 2)

Page 1 is complete with a small table with 5 observations. Notice since the graph wanted a full page it is positioned it on the next page even though I said "startpage=no" in the ODS PDF statement.



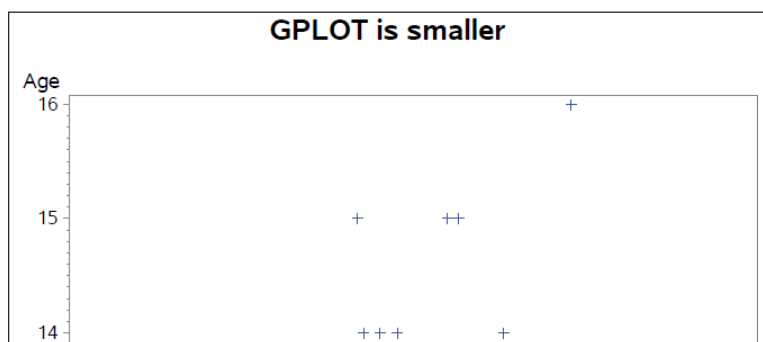
Display 33. SAS/GRAPH Displayed within ODS PDF Destination (page 2 of 2)

Most people like to maximize space in their output and want the graph to be on the same page as the table. GOPTIONS is what we need to solve this problem. GOPTIONS are graphical options that can control your graphical output. We can set two size attributes (hsize and vsize) to make the graph fit in a smaller space. Now we can try it one more time with GOPTIONS set.

```
options number;
goptions reset=all border hsize=5in vsize=5in;
title 'GPLOT is smaller';
ods pdf file="graph.pdf" notoc startpage=no;
proc print data=sashelp.class(obs=5); run;
proc gplot data=sashelp.class; plot age*height; run;
ods pdf close;
```

GPLOT is smaller

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



Display 34. SAS/GRAPH Displayed within ODS PDF Destination All on 1 Page

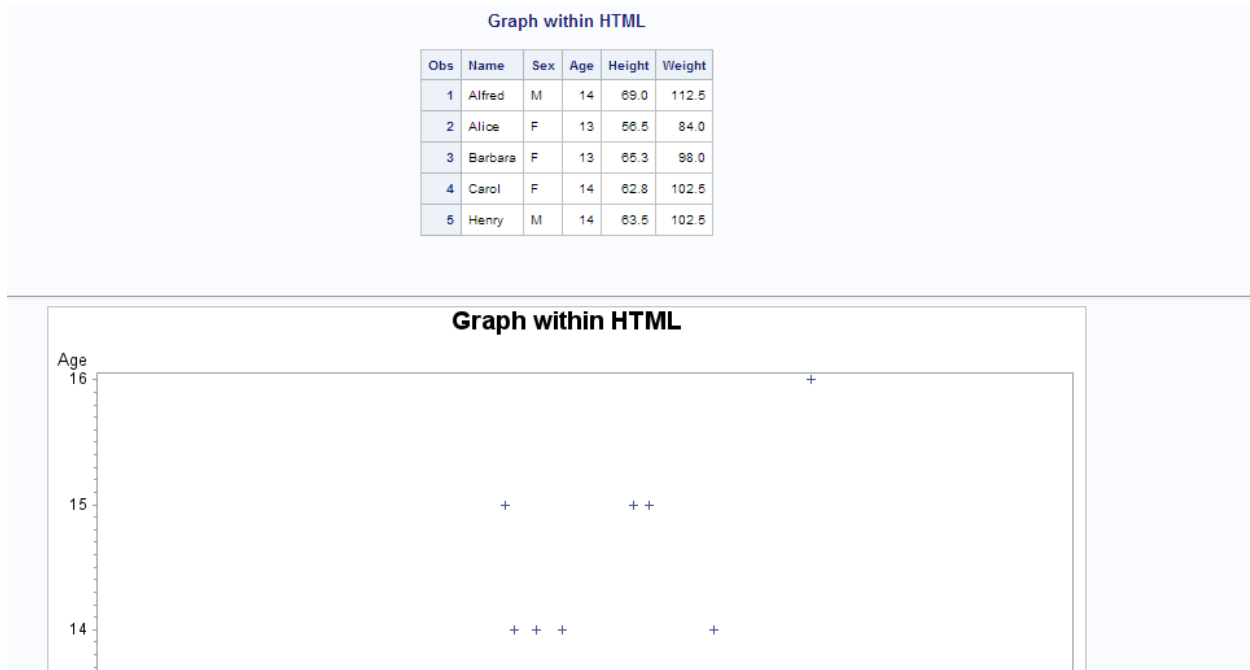
So using GOPTIONS the graph is telling ODS PDF that it only needs 5 inches of both height and width. ODS PDF determines that space is available on page 1. Therefore, it places it below the table.

As I said before HTML just references the file that SAS/GRAPH creates and displays it on the page.

```

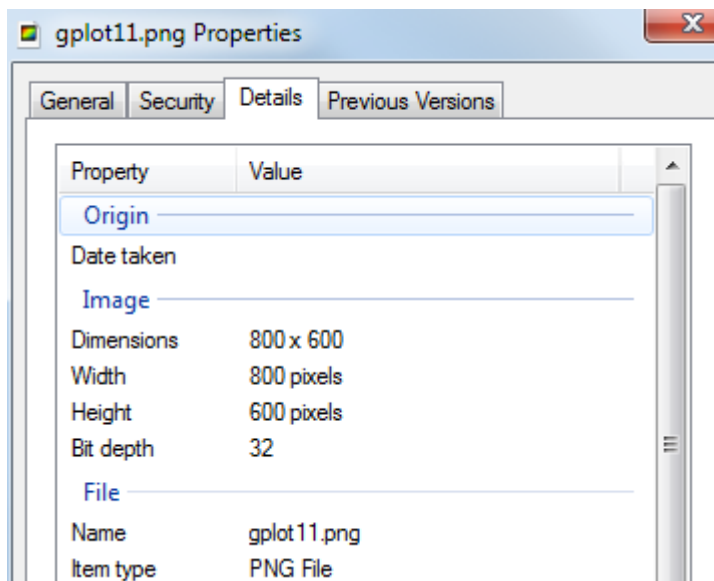
options reset=all border;
title 'Graph within HTML';
ods html file="graph.html";
proc print data=sashelp.class(obs=5);run;
  proc gplot data=sashelp.class; plot age*height; run;
ods html close;

```



Display 35. SAS/GRAPH Displayed within ODS HTML Destination

Remember I said HTML just references the file and uses the size in the file's properties. Here is the file that was created in my current working directory by SAS/GRAPH.



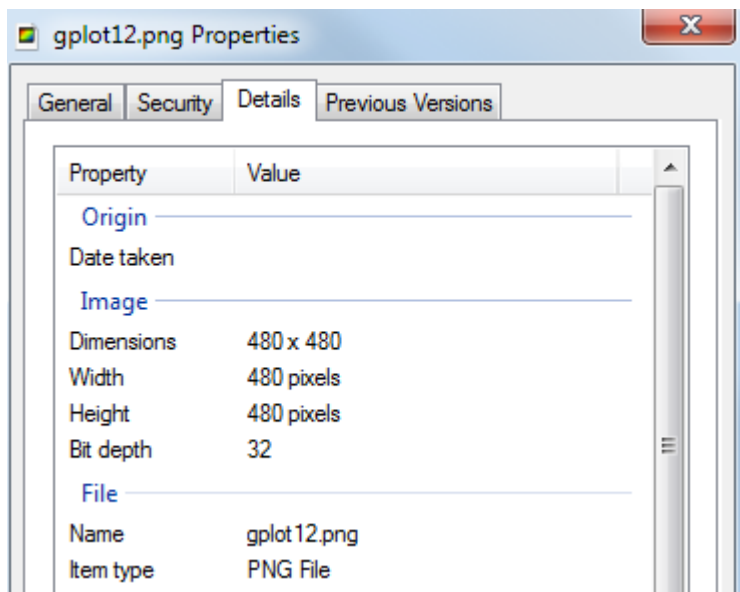
Display 36. Property File of GPLOT (initial size)

If I want to use GOPTIONS to set my size of the gplot, the PNG will have different dimensions and again HTML just plugs the file into the output. Here are the properties if I set hsize and vsize on the GPTIONS.

```

options reset=all border hsize=5in vsize=5in;

```

Display 37. Property File of GPLOT (size set by GOPTIONS statement)

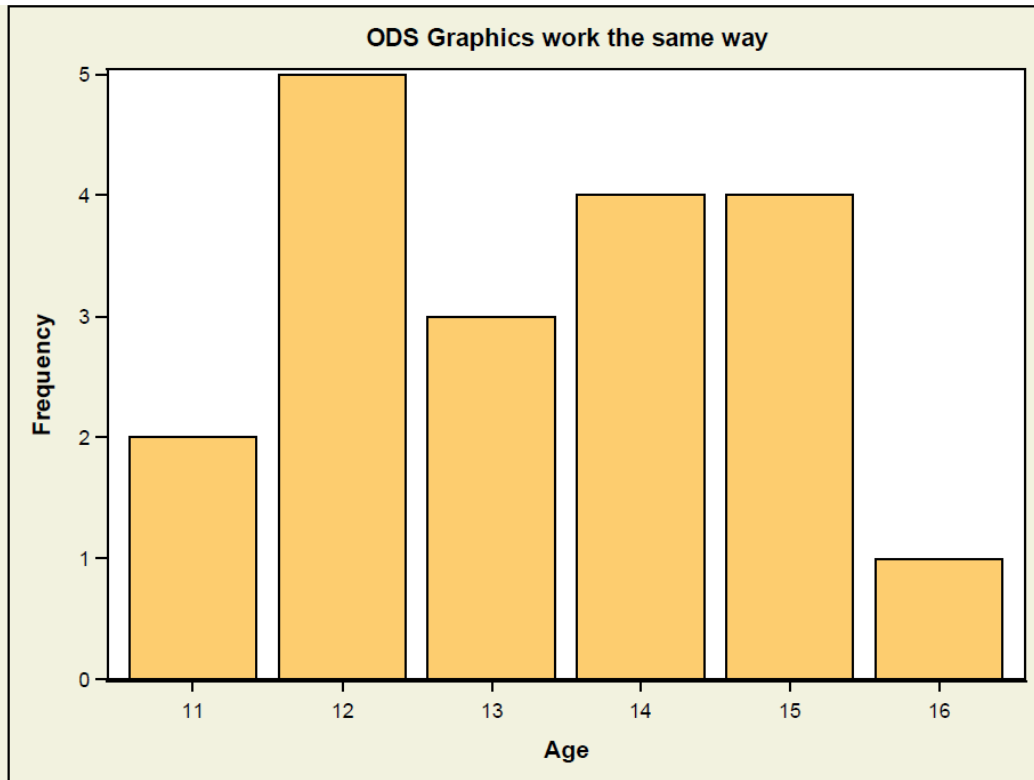
Notice how the width and height changed. SAS/GRAPH uses 96 DPI when creating the graphs. Using simple math again we see the $480/96 = 5$. We set the vsize and hsize to 5 inches in our SAS program. Now we can see how SAS/GRAPH honored our settings and created the graph to our specifications.

Now we can look at ODS GRAPHICS. It will be the same as SAS/GRAPH. ODS HTML will look for the created file in the current work directory containing the graph. ODS PDF will check space availability before placing the graph in the output. There are two differences with ODS GRAPHICS:

- (1) It has a different statement to set its height and width. It uses the ODS GRAPHICS statement.
- (2) The default size in PDF will NOT be the whole page. ODS GRAPHICS has a default size set in a template.

Here is an ODS GRAPHICS example in ODS PDF.

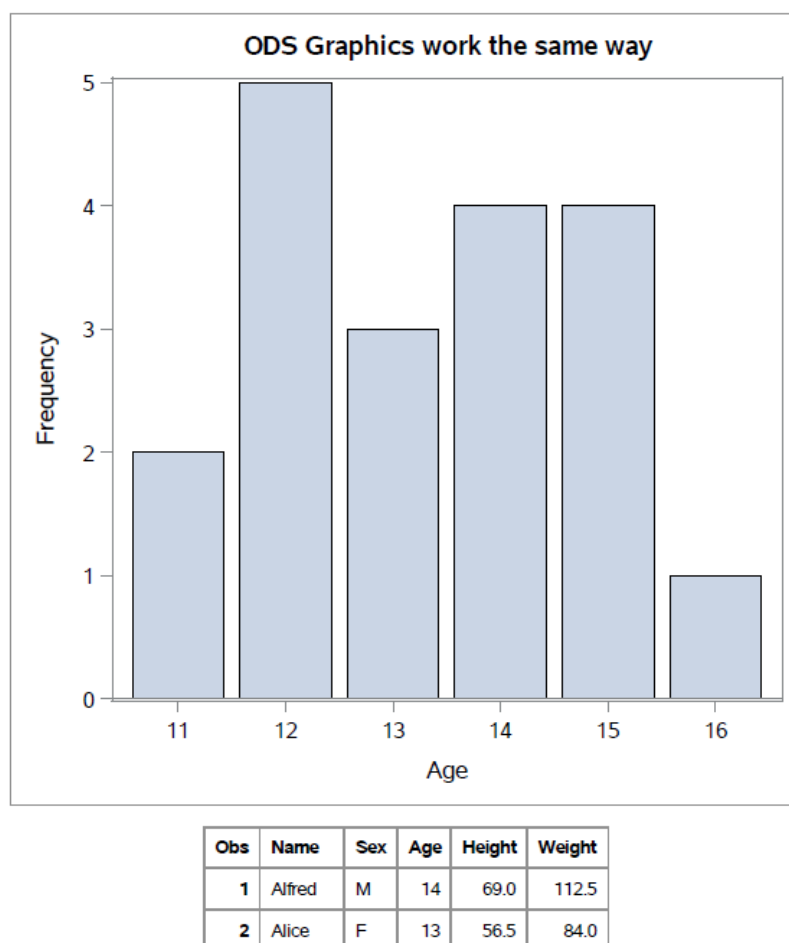
```
ods graphics on / reset=all;
ods pdf file="sgplot.pdf" style=styles.harvest;
title1 'ODS Graphics work the same way';
proc sgplot data=sashelp.class;
vbar age;
run;
ods graphics off;
ods pdf close;
```



Display 38. ODS GRAPHICS Output within ODS PDF Destination

I used a style called harvest to help show that the SGPLOT is not taking up the whole page on page 1. Using the ODS GRAPHICS statement you can set a height and width. As mentioned above, this statement allows you to size your graph and maximize your space in your output.

```
options number;  
ods graphics on / reset=all height=5in width=5in;  
ods pdf file="sgplot1.pdf" startpage=no;  
title1 'ODS Graphics work the same way';  
proc sgplot data=sashelp.class;  
vbar age;  
run;  
proc print data=sashelp.class(obs=5);run;  
ods graphics off;  
ods pdf close;
```



Display 39. ODS GRAPHICS Output within ODS PDF Destination (all on one page)

If we printed out this page, and used a ruler to measure the SGPLOT we would see the graph is 5 inches by 5 inches. It's nice to know that ODS PDF will give you graphs the exact way you specify it. Knowing this allowing you to size your graphs based on your tabular output. You can fit more on page if you know how big the output will be.

Non-Tabular output (images, SAS/GRAPH, and ODS GRAPHICS) can present challenges especially in the ODS PDF destination. The good news is there are ways to resize this output and make it fit as you desire.

CONCLUSION

Life and our comic books would be boring if all the superheroes were the same. Part of the fun of reading about or watching superheroes is seeing how creative each one of them is when faced with an obstacle. Sure, Superman can just blow a boulder off of a railroad track, but Batman can shift the boulder with gadgets and Wonder Woman might have to use her golden lasso and some scientific leverage. Can ODS PDF be similar to ODS HTML and vice versa? The answer is sometimes.

Just as we are happy about the differences between our superheroes, we need to learn that each destination has things that it does well (like dealing with page numbers and page breaks for PDF or dealing with wide tables like HTML) and other things that it does not do as well. But if you remember that ODS HTML is the output used by web browsers while ODS PDF is measured output used for printing, then you will gradually learn to appreciate each destination for what it does best. People will always ask how to make one destination look like the other. We think the real answer is each destination is perfect for what it does best. No one likes to print their output from a web page. No one thinks a piece of page can hold all their data at once. We have given you several tools within each destination to modify its appearance. You control the output now. You know what to expect and how to use the tools to solve the problems. So there you go, leaping tall destinations with a single bound!

To help you out in your superhero destination adventures, we have included some bonus programs in the ZIP file of programs to illustrate some of the things that each destination does best.

Paper 038-2014

CONCLUSIONS

Table 1 shows the summary of techniques and their usage in each destination:

Technique or Option	PDF	HTML
Papersize, Printer options	+	-
Margin, Orientation, Date, Number Options	+	-
COLUMNS=	+	(not an option for HTML)
Implicit Page Breaks	+	-
Explicit Page Breaks	+	+ (logical page breaks use horizontal rule)
Font-related style change	+	+
Cellpadding style change	+	+
Color-related style changes	+	+
Graph sizing options	+	+

Table 1. Comparison of Techniques

REFERENCES

- Huntley, Scott. 2006. "Let the ODS PRINTER Statement Take Your Output into the Twenty-First Century", *Proceedings of the SAS Users Group International 31 Conference*. Available at: <http://www2.sas.com/proceedings/sugi31/227-31.pdf>
- Huntley, Scott and Lawhorn, Bari. 2010. "Getting the Right Report (Again): Your Compatibility Guide for ODS PDF 9.2", *Proceedings of the SAS Global Forum 2010 Conference*. Available at: <http://support.sas.com/resources/papers/proceedings10/035-2010.pdf>
- Huntley, Scott and Middleton, Woody. 2012. "A Different Point of View with ODS PDF in SAS® 9.3", *Proceedings of the SAS Global Forum 2012 Conference*. Available at: <http://support.sas.com/resources/papers/proceedings12/260-2012.pdf>
- Lawhorn, Bari. 2011. "Let's Give 'Em Something to TOC about: Transforming the Table of Contents of Your PDF File", *Proceedings of the SAS Global Forum 2011 Conference*. Available at: <http://support.sas.com/resources/papers/proceedings11/252-2011.pdf>
- O'Connor, Daniel. 2010. "011-2010: Zoom, Zoom: Get Your Document to Scale on All Paper Sizes", *Proceedings of the SAS Global Forum 2010 Conference*. Available at: <http://support.sas.com/resources/papers/proceedings10/011-2010.pdf>
- Zender, Cynthia. 2007. "Funny ^Stuff~ in My Code: Using ODS ESCAPECHAR". Technical Papers and Presentations. Cary, NC:SAS Institute Inc. Available at: <http://www2.sas.com/proceedings/forum2007/099-2007.pdf>

ACKNOWLEDGMENTS

The authors would like to thank their spouses and children for the time to work on this paper. In addition, thanks are due to the reviewers, Bari Lawhorn, Chevell Parker, Kevin Smith, Allison Crutchfield, and David Kelley and Michele Ensor for making suggestions that improved the paper. And, thanks also to Donna Lanningham who edited this paper and improved the readability by removing grammatical and usage errors.

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the authors at the following address:

Scott Huntley
SAS Institute Inc.
Building S, SAS Campus Drive
Cary, NC 27513
E-mail: Scott.Huntley@sas.com
[http://www.sascommunity.org/wiki/PDF vs. HTML: Can%27t We All Just Get Along%3F](http://www.sascommunity.org/wiki/PDF_vs._HTML:_Can%27t_We_All_Just_Get_Along%3F)

Name: Cynthia Zender
Organization: SAS Education
E-mail: Cynthia.Zender@sas.com

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

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