Paper SAS339-2014

Advanced Mobile Reporting with the ODS EPUB3 Destination

David W. Kelley, SAS Institute Inc., Cary, NC

ABSTRACT

The Base SAS® 9.4 Output Delivery System (ODS) EPUB destination enables users to deliver SAS® reports as e-books on Apple mobile devices. The first maintenance release of SAS® 9.4 adds the ODS EPUB3 destination, which offers powerful new multimedia and presentation features to report writers.

This paper shows you how to include images, audio, and video in your ODS EPUB3 e-book reports. You learn how to use publishing presentation techniques such as sidebars and multicolumn layouts. You become familiar with best practices for accessibility when employing these new features in your reports.

This paper provides advanced instruction for writing e-books with ODS EPUB. Please bring your iPad, iPhone, or iPod to the presentation so that you can download and read the examples.

INTRODUCTION

The SAS 9.4 ODS EPUB destination enables you to create SAS reports as e-books that look great and travel well on your smartphone or tablet. ODS EPUB is a part of Base SAS, and runs everywhere that Base SAS runs. ODS EPUB generates e-books that work optimally with the Apple iBooks e-book reader on iPad, iPhone, and iPod. Results will vary with other e-book readers. If you are not familiar with ODS EPUB, please read the SAS Global Forum paper on that subject (Kelley, Langston, and Summers, 2013).

The ODS EPUB destination suits well for traditional SAS reports containing tables, graph, and text, which render nicely in iBooks. However, iBooks is capable of providing a much richer e-reading experience. iBooks implements much of the 2011 EPUB 3 specification, which is based in part on HTML 5 and CSS 3. Those technology standards allow for audio and video, as well as advanced presentation features such as multicolumn layout. iBooks supports those EPUB 3 features and more.

The ODS EPUB destination is based on the EPUB 2 specifications, which were approved in 2006 and 2007. EPUB 2 is defined with the XHTML 1.1 and CSS 2 standards, which do not allow for multimedia or multicolumn layout. The problem is that what ODS EPUB can produce falls short of what iBooks can consume. How can customers create e-books with the ease of ODS EPUB that take full advantage of the multimedia capabilities of their mobile devices?

With the first maintenance release of SAS 9.4 ODS EPUB3 destination, you can harness the multimedia and advanced presentation features of iBooks to produce engaging, content-rich mobile reports as e-books. Like ODS EPUB, ODS EPUB3 is a part of Base SAS. Note that on z/OS, ODS EPUB3 requires the HFS file system.

After reading the paper or viewing the presentation, all users should be able to view the example e-books on an iPad, iPhone, or iPod. Novice users should be able to run the example SAS jobs to generate and view multimedia e-books. Experienced users should be able to write a multimedia e-book from scratch. The desired result is that you will employ creative uses of multimedia and advanced presentation features in your e-books.

50 STATES OR BUST

For personal enjoyment and to test the ODS EPUB3 destination as I developed it for the first maintenance release of SAS 9.4, I wrote an e-book with it. The e-book, unimaginatively titled *Visit 50 States*, is about my family's pursuit of the goal of visiting all 50 states in the United States. I originally wrote *Visit 50 States* using the ODS EPUB destination, and then upgraded to the ODS EPUB3 destination to take advantage of its new capabilities. ODS EPUB3 can do everything ODS EPUB can do plus more.

In some ways the original version of *Visit 50 States* presented like traditional SAS output. It contained familiar reporting elements, including:

- Text defining and illustrating our family's criteria for a state visit
- Choropleth map of states visited
- Crosstabular summary report of states visited
- Appendix of web links to memorable attractions in each state visited

In other ways *Visit 50 States* presented like the e-book that it was. It had these e-book structures:

- Custom cover with title and author information
- Custom interactive table of contents
- Multiple chapters

The fun challenge for me was to improve both the content and the structure of *Visit 50 States* using the features in ODS EPUB3 and iBooks. Looking for the biggest bang for the buck, I turned my attention first to multimedia support, which is where iBooks shines, especially on iPad.

MULTIMEDIA

Broadly considered, multimedia describes the combination of content types in an e-book or other electronic publication. Content types include text, still images, animated images, audio, and video.

ODS' support for multimedia varies. Text has been in SAS reports in the form of titles and footnotes forever. Text as a first-class content type (i.e. ODS output object) is a recent development, having initially appeared in production form in SAS 9.4. Images, both still and animated, were originally supported by ODS HTML in SAS® 8.0 and today are broadly supported across ODS destinations, although not as output objects. ODS EPUB3 is currently the only destination that supports audio and video. Like images, audio and video are not output objects.

Because images, audio, and video are not output objects, they cannot be selected or excluded, or stored to and retrieved from an ODS Document item store. Another consequence is that you cannot directly specify images, audio, and video with SAS statements. SAS procedures and ODS statements do not know about them.

How then does ODS EPUB3 support images, audio, and video? It supports them with destination-specific inline formatting functions. This means that in order to include images, audio, and video in your e-book, you need to specify them as part of SAS language text statements. This might seem odd and contrived at first, but you will find that it's a natural way to do things. That's because by nature multimedia combines text with images, audio, and video.

Let's see how this works in ODS EPUB3. We'll start retracing my multimedia makeover of the *Visit 50 States* e-book by adding an image to it. To simplify the discussion, let's pretend as if I am doing it for the first time. You'll know that it's pretend because I won't have any false starts!

PICTURE THIS

I want to add a title page to my e-book. It will be nothing fancy – just a title, image, and my name. iBooks already presents the title and author information in its user interface, but I want to duplicate it in the e-book's contents. Doing so will provide a guaranteed means for visually challenged readers to be able to obtain the information.

To ensure that such readers also have a guaranteed opportunity to read the title page, I want iBooks initially to open to the title page. No reader will be able to miss it!

I will structure the title page as a separate chapter. This will allow me to assign *semantics* to it. Semantics are an EPUB 3 innovation that enables the author to specify structural information about the chapter. E-book readers can take advantage of semantic information to offer context-sensitive behaviors. iBooks currently has limited support for semantics, but it's good practice to define them in order to future-proof your e-book.

With the exposition out of the way, let's go directly to the SAS code:

```
title;
footnote;
ods epub3 file="visit_50_states.epub" title="Visit 50 States" style=ebook
          options(creator="David W. Kelley" cover=”multimedia/cover.jpg”
                  chapter_type=”frontmatter titlepage”);
ods escapechar=’^’;
ods proclabel=”Visit 50 States”;
proc odstext contents=””;
    p “Visit 50 States”/style=systemtitle(just=c);
    p “^{image multimedia/CO_family.jpg?width=75%;alt_desc=Kelley family on
vacation in Colorado in 2011}”/style={just=c};
    p “David W. Kelley”/style={just=c};
run;
```

Remember that we are adding a chapter to an existing e-book. I am not showing the SAS code for the existing chapters. To keep the discussion tight and focused, we are considering only the changes to *Visit 50 States.*

The ODS EPUB3 statement opens the destination for writing to the file *visit_50_states.epub*. The ODS EPUB3 statement STYLE option specifies a custom *EBook* style, which for brevity's sake is not detailed here. The e-book title, author, and cover image are specified as per usual.

The CHAPTER_TYPE option assigns the semantics for the chapter. Valid semantic terms are defined by the EPUB 3 Structural Semantics Vocabulary, http://www.idpf.org/epub/vocab/structure/. You can specify as many terms as you like and in any order. The default semantics for ODS EPUB3 e-book chapters are "bodymatter chapter".

The ODS ESCAPECHAR statement defines the '^' character as the signal to ODS to treat whatever text follows it as an inline formatting function. ODS PROCLABEL customizes the table of contents entry for the ensuing procedure step.

That procedure is PROC ODSTEXT, which is the hook on which we hang our multimedia hat. PROC ODSTEXT allows you to create any number of nested lists and paragraphs. Making the title page for *Visit 50 States* is a straightforward matter of specifying a paragraph (P statement) for each of the title, image, and author. The STYLE option centers the paragraphs, which are left-justified by default. The ODSTEXT statement CONTENTS option suppresses the output object entry in the table of contents.

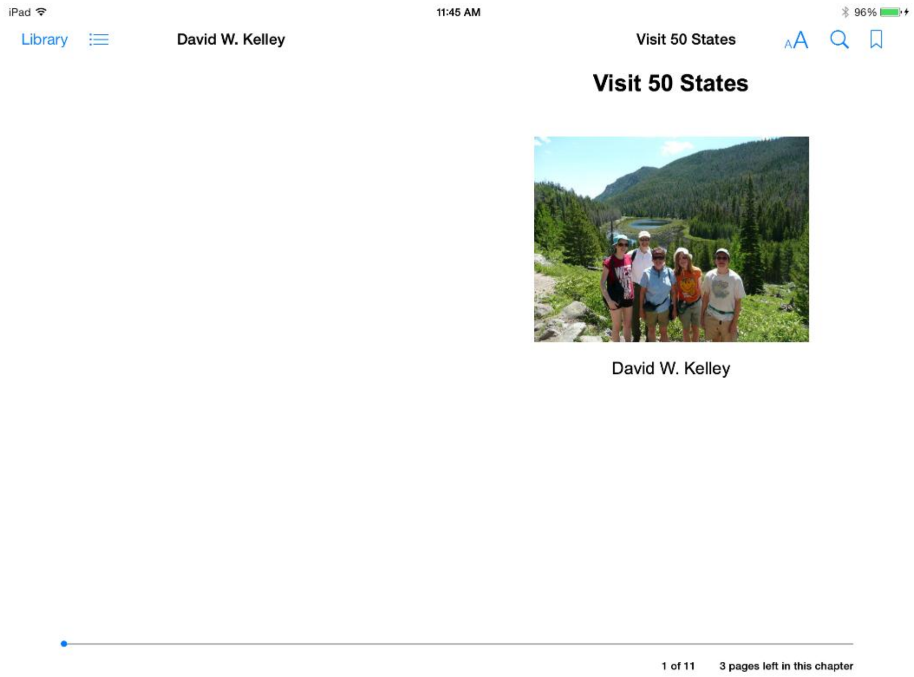
The IMAGE inline formatting function in the second paragraph inserts the supplied image file into the *visit_50_states.epub* ZIP file. It also scripts a reference to the image in the internal HTML5 file that implements the title page chapter. You can specify either a filename or a fileref for the image. SAS expects that relative image filenames are relative to the current working directory.

If the IMAGE function syntax looks familiar to you, it's because it is based on the web query string syntax. If it looks unfamiliar to you, well that's because it's not like any other syntax in SAS or ODS!

Here's what you need to know about images and the IMAGE function:

- You can supply options to the IMAGE function. Append a '?' character followed by the options. Separate the options with a ';' or '&' character.
- The IMAGE function accepts the options WIDTH, HEIGHT, and ALT_DESC, as well as CSS properties such as FLOAT. The function scripts an HTML IMG tag in the internal HTML5 file, so the W3C IMG element documentation is a good resource.
- Use percentages when supplying the image width or height. This will help to ensure a consistent user experience across e-book reader software and hardware platforms.
- Always provide informative alternate text via the ALT_DESC option. The image might not be accessible to all of your readers, so additional context is required.
- ODS EPUB3 supports PNG, GIF, JPEG, and SVG images. Use JPEG for photographs and SVG for small-footprint, scalable performance. Be warned that putting too many SVG images in your e-book will slow down iBooks noticeably, because it has to draw each image based on its vector graphics definition.
- iBooks will zoom an image if you double-tap it. To prevent pixilation, raster images should be at least 1.5 times the intended viewing size up to a maximum of 3.2 million pixels.

Figure 1 shows how iBooks displays the title page of *Visit 50 States*.



Visit 50 States

David W. Kelley

Figure 1: Title Page in iBooks 3.2 on iPad (iOS 7)

Because it is the first page in the e-book, iBooks displays it on the right page of its two-page layout. As instructed, iBooks sizes the image width to be 75% of the page width.

Now that you know how to add images to your ODS EPUB3 e-books, let's move on to audio. Hearing is believing!

NOW HEAR THIS

I want to add a travel-themed song to *Visit 50 States*, and I know which one: the Johnny Cash version of Geoff Mack's "I've Been Everywhere." After purchasing the song from iTunes, I am ready to write the SAS code to insert the song into my e-book. But first I need to step back and consider legalities.

If I add a song that I purchased to an e-book I create for my personal use, then I am not harming anyone. However, if I redistribute my e-book, then I am essentially pirating copyrighted music. I am not going to do that, of course! I say it as fair warning to you to be mindful of how you employ copyrighted media in your e-books. I say it also to explain why any version of *Visit 50 States* that I redistribute will not contain the song I purchased. Probably I will substitute a public domain recording of it or another travel-themed song, such as "This Land is Your Land" by Woody Guthrie.

With legal ramifications duly noted, I am itching to write the code. But first I need to decide how to present the song in my e-book. The fact is that there is nothing to *see* – its music after all. And the song has nothing to do with my family vacations per se -- it's just an entertaining add-on. For those reasons I do not want to place the song in the linear flow of the e-book's content. Instead, I will put it in the non-linear section of the e-book and link to it from the "SMART Traveling" chapter. I will also add an entry for the song to the table of contents so that it can be readily discovered by the reader. For simplicity's sake I will show how to set up only the table of contents link.

Finally, we are clear to write the code:

```
ods epub3 newchapter=now options(nonlinear="chapter");
title "^{style [url='http://en.wikipedia.org/wiki/I''ve_Been_Everywhere']I've Been
Everywhere}";
title2 "Written by Geoff Mack";
title3 "Sung by Johnny Cash";
proc odstext contents="";
   p "^{image multimedia/albumartwork.jpg?width=15%;alt_desc=Johnny Cash album
artwork}"/style={just=c};
   p "^{audio multimedia/ivebeeneverywhere.m4a?controls=controls}"/style={just=c};
run;
title;
ods epub3 options(nonlinear="none");
ods epub3 event=branch(start label="I've Been Everywhere (Audio)"
url="chapter2.html");
ods epub3 event=branch(finish);
```

The first ODS EPUB3 statement starts a new nonlinear chapter. Once again PROC ODSTEXT handles the multimedia. The IMAGE inline formatting function in the first paragraph inserts an image of the Johnny Cash album artwork into the *visit_50_states.epub* ZIP file. The AUDIO inline formatting function in the second paragraph inserts the M4A audio file for the song I purchased into the ZIP file. I specified a file, but a fileref would have worked just as well.

As with the IMAGE function, you can specify options to the AUDIO function. The options include:

- AUTOPLAY: The audio will automatically start playing without stopping.
- CONTROLS: The audio controls should be displayed.
- LOOP: The audio will start over again, every time it is finished.
- MUTED: The audio output should be muted.
- PRELOAD: Specifies if and how the audio should be loaded.

Refer to the W3C documentation for the HTML5 AUDIO element for more information.

ODS EPUB3 supports the MP3, MP4, M4A, and WAV audio formats. Apple employs the M4A format in its products and services, so you can expect an excellent user experience with M4A audio in iBooks.

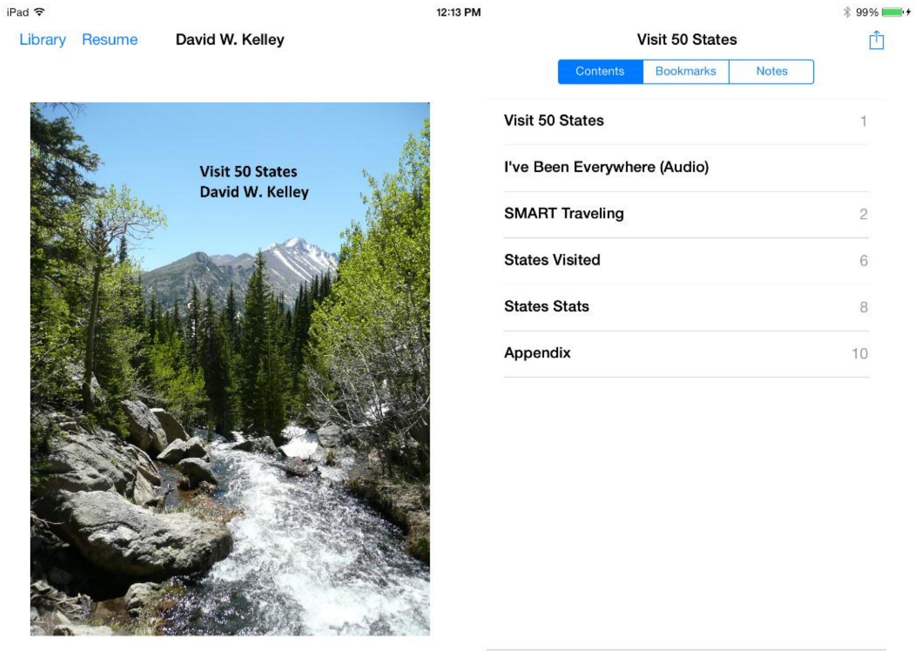
After the PROC ODSTEXT step come three ODS EPUB3 statements. They all relate to the fact that the current chapter is nonlinear. The first ODS EPUB3 statement establishes that subsequent output will appear in the linear flow of the e-book. The next two ODS EPUB3 statements specify the BRANCH event to create an entry for the chapter in the table of contents. ODS EPUB3 does not automatically create table of contents entries for nonlinear output, so we need to do it ourselves.

BRANCH events can be nested to build nested table of contents entries. Match up the START and FINISH options to the desired depth. Our chapter needs just a single entry, so two BRANCH events suffice.

Specify the label and URL for the table of contents entry with the LABEL and URL options, respectively. iBooks will display the label in the table of contents. If the reader selects it, iBooks will navigate to the specified chapter.

This raises the question: How do you know what the URL of this or any chapter is? Its simple -- the first chapter is “chapter1.html”, the second chapter is “chapter2.html”, and so on. It could be argued that the title page, for example, is not logically a chapter and should not be named “chapter1.html”. Do not confuse the purpose of an e-book section with how it is represented physically in an ODS EPUB3 *.epub* ZIP file. The motivation for the naming scheme is that the predictable chapter names make it possible to write SAS code to reliably link to the chapters from the table of contents or the chapters themselves.

Figure 2 displays the updated iBooks table of contents.



Visit 50 States 1

I've Been Everywhere (Audio)

SMART Traveling 2

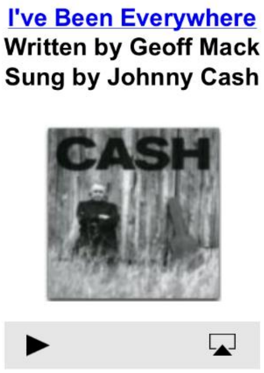
States Visited 6

States Stats 8

Appendix 10

Figure 2: Table of Contents in iBooks

The first entry (“Visit 50 States”) links to the title page. The second entry (“I’ve Been Everywhere (Audio)”) is the one we just created with the ODS EPUB3 BRANCH events. Notice that it has no associated page number. That’s because the chapter is nonlinear. Selecting “I’ve Been Everywhere (Audio)” takes us to the Johnny Cash album artwork and song; see Figure 3. For the sake of space, I cropped the screenshot.



I've Been Everywhere
Written by Geoff Mack
Sung by Johnny Cash

Figure 3: "I've Been Everywhere" Chapter in iBooks

If you select the link “I’ve Been Everywhere”, iBooks will launch the Safari web browser, which will display a Wikipedia entry about the song. From the user experience perspective it’s a nice touch, although hardly necessary.

Select the play button on the audio control and iBooks will play the song. That’s all there is to it!

Now that you know how to add audio to an e-book, let’s turn our attention to video.

VIVA VIDEO

My wife takes the photographs in our family. She documents our vacations with numerous photographs and the occasional video. While driving on the Trail Ridge Road in Colorado in 2011, she filmed a marmot (large mountain-dwelling ground squirrel) gnawing a piece of wood. Being that we live in central North Carolina, the marmot was a curiosity, and it was incredibly cute besides. When it came time to add a video to my *Visiting 50 States* e-book, I knew that it had to be the marmot video.

Unlike with the Johnny Cash song, I want to include the marmot video in the linear flow of the e-book. Specifically I want to add it to the end of the “SMART Traveling” chapter, which is the main chapter of the e-book. I see the video as a fun reward and send-off for readers who make it through the e-book. Like the song, I do not consider the video to be an integral part of the e-book. The e-book can be enjoyed without them. That is an important requirement for accessibility.

Here is the SAS code to add the video:

```
ods proclabel = "Marmot Sighting (Video)";
proc odstext contents="";
   p "^{video
multimedia/marmot.m4v?controls=controls;poster=multimedia/marmot.jpg;float=left;
margintop=0;marginright=.5em;width=240px;height=135px}
      No vacation is complete without home video to capture highlights of the
journey.  If you enjoy wildlife, check out the marmot^{noteref large mountain-
dwelling ground squirrel} we saw while driving the
      ^{style [url='http://en.wikipedia.org/wiki/Trail_Ridge_Road']Trail Ridge
Road}.";
run;
```

The major new feature here is the VIDEO inline formatting function. It requires a video file, which can be specified by filename or fileref. The VIDEO function accepts the same core options as the AUDIO function, plus one more: POSTER. POSTER is for specifying a poster image that will be displayed by the video control prior to playback. iBooks requires a poster image. Commonly the poster image is created from a frame from the associated video.

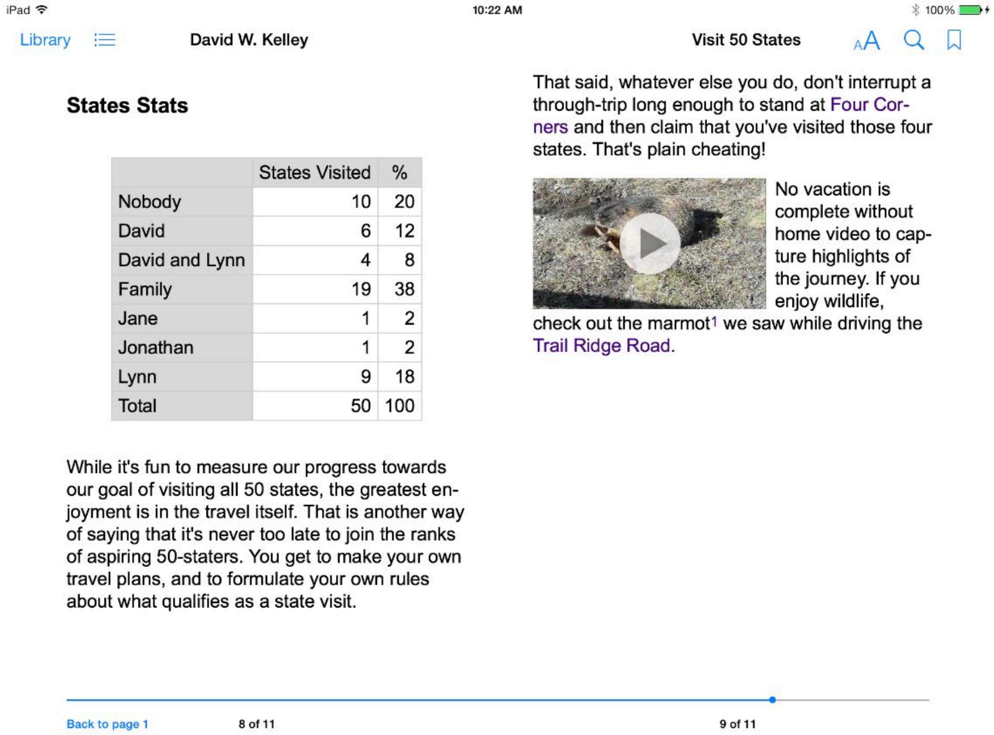
In addition to the poster image, I’ve specified the CSS FLOAT, MARGINTOP, and MARGINRIGHT attributes in order to float the poster image to the left of and top-aligned with the paragraph text.

Previously I recommended defining a percentage width for images. But for the poster image I’ve defined width and height in pixels. Why is that? If you do not specify dimensions that have the same aspect ratio as the image, then iBooks will draw unsightly black borders around the image in the video control. The poster image dimensions are 480 by 270 pixels, so I’ve defined the displayed dimensions to be 240 by 135 pixels. This yields an appropriately sized image with no borders.

To learn more about VIDEO options, see the W3C documentation for the HTML5 VIDEO element.

ODS EPUB3 supports the M4V, MP4, and QuickTime video formats. Apple developed both M4V and QuickTime, so you cannot go wrong with either format in iBooks.

Figure 4 shows how iBooks renders the video control in the chapter.



States Stats

	States Visited	%
Nobody	10	20
David	6	12
David and Lynn	4	8
Family	19	38
Jane	1	2
Jonathan	1	2
Lynn	9	18
Total	50	100

While it's fun to measure our progress towards our goal of visiting all 50 states, the greatest enjoyment is in the travel itself. That is another way of saying that it's never too late to join the ranks of aspiring 50-staters. You get to make your own travel plans, and to formulate your own rules about what qualifies as a state visit.

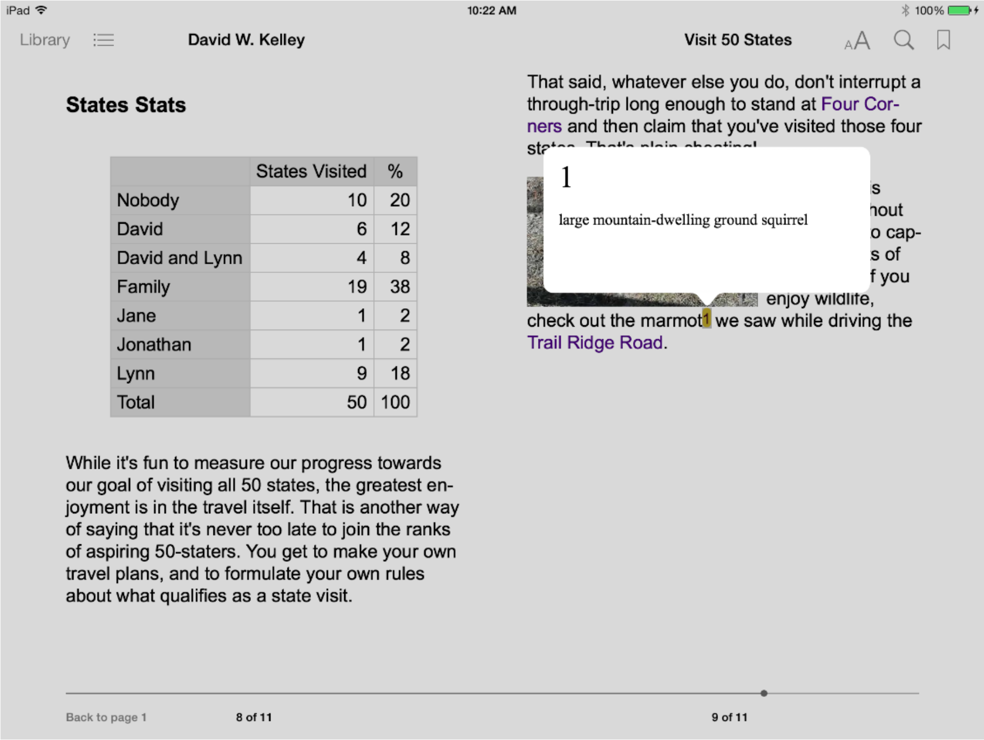
That said, whatever else you do, don't interrupt a through-trip long enough to stand at Four Corners and then claim that you've visited those four states. That's plain cheating!

No vacation is complete without home video to capture highlights of the journey. If you enjoy wildlife, check out the marmot[1] we saw while driving the Trail Ridge Road.

Figure 4: "SMART Traveling" Chapter in iBooks

The SAS code premieres another ODS EPUB3 feature: the NOTEREF inline formatting function. The NOTEREF function declares text that iBooks will display in a pop-up footnote. By default the footnote marker is “1”, “2”, and so on. You can customize the marker with the TEXT option.

Just like I felt the need to tell you what a marmot is, I feel the need to tell my e-book readers the same thing. The pop-up footnote is a handy way to present such parenthetical information. When you select the footnote marker “1” in the paragraph text, iBooks pops-up a footnote with the information. See Figure 5.



States Stats

	States Visited	%
Nobody	10	20
David	6	12
David and Lynn	4	8
Family	19	38
Jane	1	2
Jonathan	1	2
Lynn	9	18
Total	50	100

While it's fun to measure our progress towards our goal of visiting all 50 states, the greatest enjoyment is in the travel itself. That is another way of saying that it's never too late to join the ranks of aspiring 50-staters. You get to make your own travel plans, and to formulate your own rules about what qualifies as a state visit.

That said, whatever else you do, don't interrupt a through-trip long enough to stand at Four Corners and then claim that you've visited those four st

1

large mountain-dwelling ground squirrel

s
hout
o cap-
s of
f you
enjoy wildlife, check out the marmot1 we saw while driving the Trail Ridge Road.

Figure 5: Pop-up Footnote in iBooks

REMOTE AUDIO AND VIDEO

Creating mobile output is ODS EPUB3's calling card. I'm disinclined to advocate for uses that trade off this feature. However, sometimes it makes sense to do so.

Consider video file size – the 16-second marmot video in *Visit 50 States* is 12.5M. Imagine the size of a 15-minute promotional video or of a corporate webcast. It would easily exceed Apple's recommended maximum e-book size of 500M. (The maximum size of an *.epub* ZIP file is 2G.) If you want to make heavy use of video in your e-book, then you need a way to get around the size restriction.

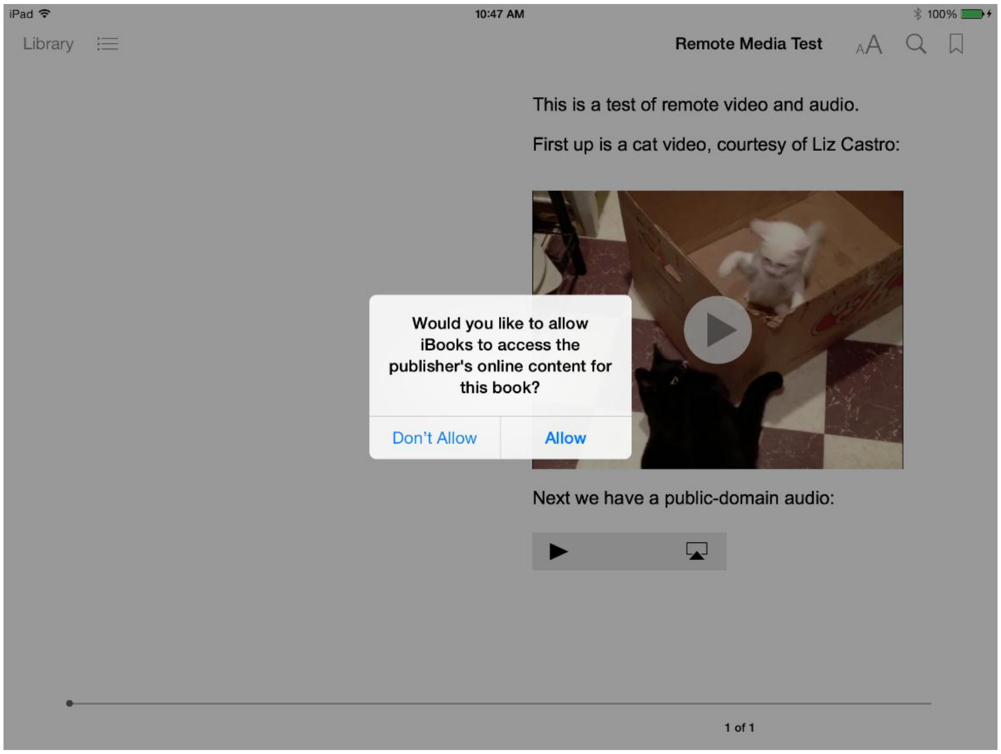
The EPUB3 3 specification writers recognized the need for an exception to the requirement that EPUB 3 e-books be fully portable in the cases of audio and video. For just these media types, you can specify that they be remote (Internet-served) rather than local (included directly in the *.epub* file). This setup enables you to play audio and video segments of any length. What's more, you can update the content of the media as needed, as long as you continue to serve it from the same remote URL.

Because I'm not going to add remote media to *Visit 50 States*, I will not be able to draw on it for an example. Instead, I'll show you an example I created during ODS EPUB3 development. See below.

```
title;
ods epub3 file="remote_media.epub" title="Remote Media Test";
proc odstext contents=””;
   p "This is a test of remote video and audio.";
   p "First up is a cat video, courtesy of Liz Castro:";
   p "^{video
http://www.elizabethcastro.com/epub/examples/catbox.mp4?controls=controls;poster=mu
ltimedia/poster.jpg;width=382px;height=287px}";
   p "Next we have a public-domain audio:";
   p "^{audio http://www.albertopettarin.it/misc/testextaudio.mp3}";
run;
ods epub3 close;
```

The only thing different about remote media is the URL specification. Note that you still need a poster image for remote video, and that image must be local. iBooks presents local and remote media in the same way, except if the remote media is unavailable (for example, when there's no Internet connection).

iBooks will prompt you to allow access to remote media the first time you attempt to consume it. See Figure 6.



This is a test of remote video and audio.

First up is a cat video, courtesy of Liz Castro:

Would you like to allow iBooks to access the publisher's online content for this book?

Next we have a public-domain audio:

Figure 6: Access to Remote Media in iBooks

Ideally, if you employ remote media in your e-books, it should be supplementary content. Your readers should be able to benefit from reading your e-book even if no Internet connection is available. Preserve mobility!

CSS PRESENTATION TECHNIQUES

SAS customers have been customizing reports with CSS for years. As old as CSS is, ODS is older. When ODS was invented, CSS did not exist, and the ODS developers necessarily created a proprietary technology, ODS styles, for defining stylistic effects in reports.

With each SAS release over the years, ODS style compatibility with CSS has improved. The two technologies are not and never will be interchangeable, however. CSS can do things that ODS in general doesn't ever need to do. The rub is that CSS can do things that ODS EPUB3 in particular wants to do. How do we get CSS features not in ODS styles into ODS EPUB3?

For the first maintenance release of SAS 9.4, we went back to the future with ODS EPUB3. From ODS HTML4, circa SAS® 9.1, we identified the STYLESHEET option and brought it forward into ODS EPUB3. This is a logical enhancement to ODS EPUB3, which is based in part on HTML 5.

The ODS EPUB3 STYLESHEET option enables you to specify one or more CSS files to be included directly in the *.epub* ZIP file. The specified CSS must adhere to ODS style naming conventions for properties that ODS knows about. Of course we are interested in particular properties that ODS does not know about!

Let's start by using the ODS EPUB3 STYLESHEET option to create a multicolumn layout.

I READ IT IN A MAGAZINE

Multicolumn layout is the standard way of arranging multimedia in the print versions of newspapers and magazines. Readers prefer multiple columns because it's easier to keep your place in long text when it's wrapped in a column rather than spread out across the full width of the page.

ODS has supported multicolumn layout in the ODS PRINTER and ODS RTF destinations for years via the COLUMNS option. In SAS 9.4, ODS PRINTER, ODS HTML, and ODS HTML5 added support for ODS LAYOUT, which can do multicolumn layouts plus more.

ODS EPUB3 goes a different route. Despite its kinship with ODS HTML5, ODS EPUB3 does not support ODS LAYOUT, for two reasons. The first is that ODS HTML5 implements layout with nested tables, which render poorly in some e-book readers, including iBooks. The second is that nested tables intermix the e-book's content with its presentation, which makes the content less accessible.

This is where CSS comes in. CSS separates the content from its presentation, which makes the content more accessible. Because multicolumn layout is defined by CSS3, ODS EPUB3 can support multicolumn layout by permitting the customer to supply custom CSS via the STYLESHEET option.

To demonstrate ODS EPUB3 multicolumn layout, let's create a separate version of the *Visit 50 States* e-book just for that purpose. To uphold my tradition of unimaginative titles, let's call it *Visit 50 States (Multicolumn Layout)*. We'll follow these steps to create it:

1. Create a logically empty e-book and extract its style sheet into an external file, *custom.css*.
2. Append the multicolumn CSS to *custom.css*.
3. Open the ODS EPUB3 destination for *Visit 50 States (Multicolumn Layout)*, specifying *custom.css* as the e-book's style sheet.

We're ready for the SAS code to implement the steps:

```
/*-- create logically empty e-book; we're going to extract its style.css --*/
ods epub3 file="temp.epub" style=ebook;
ods epub3 close;

/*-- extract style.css into external file custom.css --*/
filename myfile ZIP "temp.epub";
data _null_;
   file "custom.css" recfm=n;
   infile myfile(ops/css/style.css) recfm=n;
   input x $char1. @@;
   put x $char1. @@;
run;

/*-- append custom CSS to custom.css --*/
filename custom 'custom.css' mod encoding="utf-8";
data _null_;
   file custom;
   put "@media(orientation: landscape)";
   put "{";
   put ".body";
   put "{";
   put "  -webkit-column-count: 2;";
   put "  column-count: 2;";
   put "  -webkit-column-gap: 1em;";
   put "  column-gap: 1em;";
   put "  -webkit-column-rule: thin solid blue;";
   put "  column-rule: thin solid blue;";
   put "}";
   put "}";
run;

/*-- open ODS EPUB3 destination --*/
ods epub3 file="visit_50_states_multicolumn.epub"
          title="Visit 50 States (Multicolumn Layout)"
          stylesheet=(url="custom.css")
          options(cover_image="multimedia/cover.jpg"
                  creator="David W. Kelley"
                  chapter_type="frontmatter titlepage");
```

Step 1 needs explanation. When you specify a style sheet to ODS EPUB3 with the STYLESHEET option, that style sheet overrides ODS EPUB3's default style sheet. That means that you need to supply a style sheet that does everything the default style sheet does plus whatever customizations you have defined. So you need to acquire the default style sheet, and the way to do that is to extract it from an *.epub* ZIP file. This is the programming idiom for CSS with ODS EPUB3.

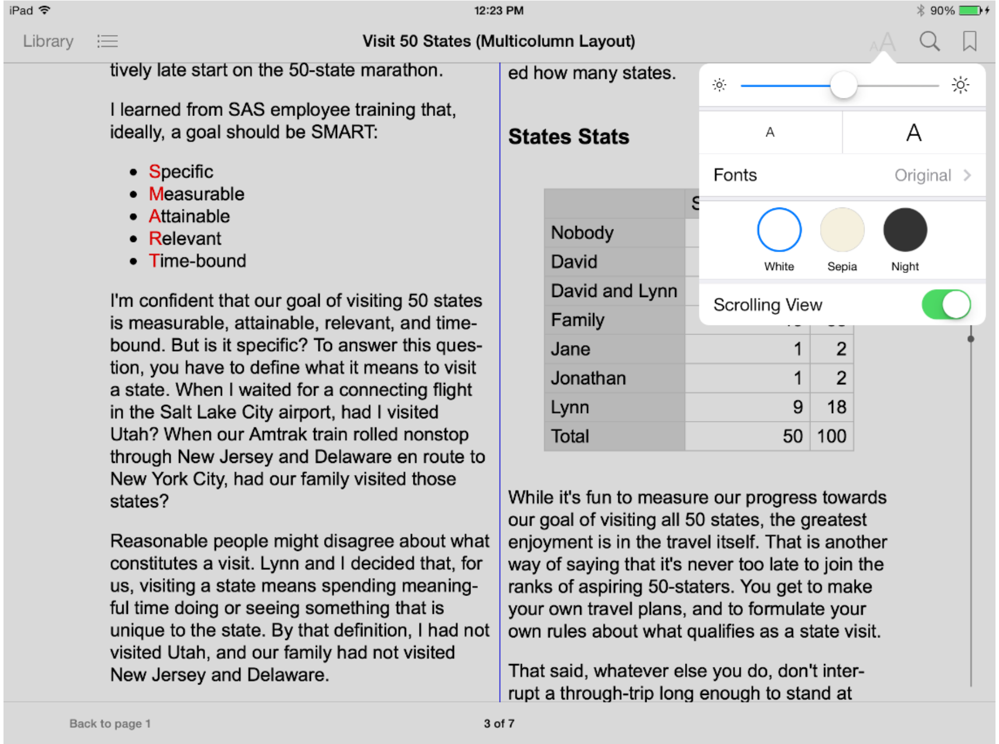
In Step 2 we define our multicolumn layout. Multicolumn layout on iPad is effective when the viewable area is more wide than tall, as is the case with landscape orientation. It's not nearly as helpful for portrait orientation, when the viewable area is more tall than wide. For this reason we'd like to apply multicolumn layout only for landscape orientation.

The EPUB 3 specification defines media queries, which are the CSS mechanism for conditionally applying stylistic effects based on particular media features. Defined features include width, height, color, orientation, and resolution. We specify landscape orientation so that multicolumn layout will take effect only when the viewable area is more wide than tall.

Next, we request two columns divided by a thin solid blue rule. We specify the WebKit-specific CSS properties because iBooks, which is based on WebKit, currently does not honor the standard properties. We nest the properties in the Body class because Body is the ODS-defined class that governs the appearance of the e-book as a whole.

Step 3 requires no additional explanation.

Figure 7 shows how iBooks renders *Visit 50 States (Multicolumn Layout)* in scroll mode.



tively late start on the 50-state marathon.

I learned from SAS employee training that, ideally, a goal should be SMART:

- Specific
- Measurable
- Attainable
- Relevant
- Time-bound

I'm confident that our goal of visiting 50 states is measurable, attainable, relevant, and time-bound. But is it specific? To answer this question, you have to define what it means to visit a state. When I waited for a connecting flight in the Salt Lake City airport, had I visited Utah? When our Amtrak train rolled nonstop through New Jersey and Delaware en route to New York City, had our family visited those states?

Reasonable people might disagree about what constitutes a visit. Lynn and I decided that, for us, visiting a state means spending meaningful time doing or seeing something that is unique to the state. By that definition, I had not visited Utah, and our family had not visited New Jersey and Delaware.

ed how many states.

States Stats

	S	
Nobody		
David		
David and Lynn		
Family		
Jane	1	2
Jonathan	1	2
Lynn	9	18
Total	50	100

While it's fun to measure our progress towards our goal of visiting all 50 states, the greatest enjoyment is in the travel itself. That is another way of saying that it's never too late to join the ranks of aspiring 50-staters. You get to make your own travel plans, and to formulate your own rules about what qualifies as a state visit.

That said, whatever else you do, don't interrupt a through-trip long enough to stand at

Figure 7: Multicolumn Layout in iBooks

Now that you know how to customize ODS EPUB3 e-books with CSS, it's straightforward to apply the technique to other presentation effects in iBooks. Next we consider sidebars.

AND NOW A WORD FROM OUR SPONSOR

A sidebar in a publication is material that is presented separately from the main discussion. Form follows function, because the sidebar material adds context to the main discussion but is not required in order to understand it.

Sidebars are naturally amenable to accessibility techniques. Accessibility is concerned with distinguishing between what must be read and what might be read. Why send a reader wandering off into the weeds if the e-book reader can keep them focused on what's most important?

The EPUB3 specification defines sidebars in a way that capable e-book readers can recognize them and navigate the reader through them or around them as desired. Currently neither ODS EPUB3 nor iBooks implements the specification for sidebars. That doesn't mean that we can't create sidebars with ODS EPUB3 and view them in iBooks. It means that iBooks will not know that the sidebar material is supplementary in nature and could be skipped, if so desired.

With that caveat set forth, let's add a sidebar to the *Visit 50 States (Multicolumn Layout)* e-book. We follow the same steps as in the multicolumn layout example. I won't include all of the SAS code, some of which we can reuse without modification. We'll consider just the changed bits, which are the custom CSS and how it applies in *Visit 50 States (Multicolumn Layout)*. The code to generate the CSS is:

```
filename custom 'custom.css' mod encoding="utf-8";
data _null_;
   file custom;
   put ".sidebar";
   put "{";
   put "  text-align: justify;";
   put "  font-style: italic;";
   put "  background-color: #f5e5e5;";
   put "  border: thin solid black;";
   put "  width: 35%;";
   put "  padding: .5em .7em .5em .7em;";
   put "  margin: .4em !important;";
   put "  float: right;";
   put "}";
run;
```

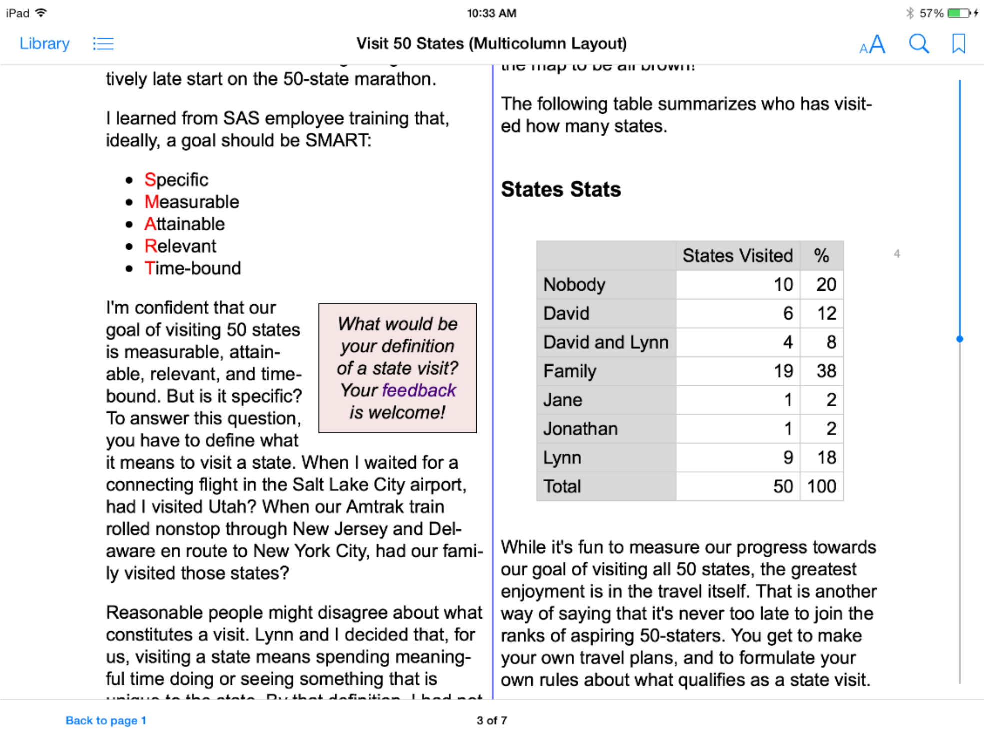
The above code defines the *sidebar* CSS class with the desired presentation properties. The sidebar text is enclosed in a box that floats right of the main text and consumes 35% of the column width.

To apply the *sidebar* CSS to the *Visit 50 States (Multicolumn Layout)* "SMART Traveling" chapter, we turn again to PROC ODSTEXT. To keep things simple, I'll omit most of the ODSTEXT code, leaving what is seen here:

```
proc odstext contents="";
   p "I learned from SAS employee training that, ideally, a goal should be SMART:";
   list;
     item "^{style [foreground=red]S}pecific";
     item "^{style [foreground=red]M}easurable";
     item "^{style [foreground=red]A}ttainable";
     item "^{style [foreground=red]R}elevant";
     item "^{style [foreground=red]T}ime-bound";
   end;
   p "What would be your definition of a state visit?  Your ^{style 
[url='mailto:David.Kelley@sas.com'] feedback} is welcome!"/style={class='sidebar'};
```

The STYLE inline formatting function in the second paragraph embeds my work e-mail address in the solicitation for feedback. The STYLE option after the paragraph text ties the content of the sidebar to its presentation. All we have to do is pass in the name of the CSS class (*sidebar*) to the CLASS option. Whether you use the STYLE option or the STYLE function typically depends on whether you want to style the text in full or in part.

Figure 8 shows how iBooks, when in scroll mode, renders the sidebar in the *Visit 50 States (Multicolumn Layout)* e-book.



tively late start on the 50-state marathon.

I learned from SAS employee training that, ideally, a goal should be SMART:

- Specific
- Measurable
- Attainable
- Relevant
- Time-bound

What would be your definition of a state visit? Your feedback is welcome!

I'm confident that our goal of visiting 50 states is measurable, attainable, relevant, and time-bound. But is it specific? To answer this question, you have to define what it means to visit a state. When I waited for a connecting flight in the Salt Lake City airport, had I visited Utah? When our Amtrak train rolled nonstop through New Jersey and Delaware en route to New York City, had our family visited those states?

Reasonable people might disagree about what constitutes a visit. Lynn and I decided that, for us, visiting a state means spending meaningful time doing or seeing something that is unique to the state. By that definition, I had not

the map to be all brown!

The following table summarizes who has visited how many states.

States Stats

	States Visited	%
Nobody	10	20
David	6	12
David and Lynn	4	8
Family	19	38
Jane	1	2
Jonathan	1	2
Lynn	9	18
Total	50	100

While it's fun to measure our progress towards our goal of visiting all 50 states, the greatest enjoyment is in the travel itself. That is another way of saying that it's never too late to join the ranks of aspiring 50-staters. You get to make your own travel plans, and to formulate your own rules about what qualifies as a state visit.

Figure 8: Sidebar in iBooks

We wrap up the discussion of CSS presentation techniques with a drop cap example.

ONCE UPON A DROP CAP

A drop cap is the first letter of a paragraph, where the letter presents as larger than the rest of the text and dropped down two or more lines. Elaborately illustrated drop caps appear in ancient manuscripts. Drop caps often are seen in novels and fairy tale books today. Whether you regard a drop cap as a nice touch or an affectation depends on your point of view!

I want to employ a drop cap in the first paragraph of the "States Stats" section of *Visit 50 States (Multicolumn Layout)*. Conveniently the first letter in the paragraph is a 'W', a relatively wide character that will present well as a drop cap.

I want the drop cap font to be a serif font – the more ornamentation, the better. I choose Times New Roman to replace the default font, Arial. It's ironic that these old Microsoft fonts feature prominently on cutting-edge Apple mobile devices. That's a legacy of Microsoft Word and Internet Explorer, for good or for bad.

As before, I'm showing just the changes to the SAS code. Below you'll see that I've added code to define the *dropcap* CSS class. I set the FLOAT property so that the drop cap will float to the left of the rest of the paragraph text.

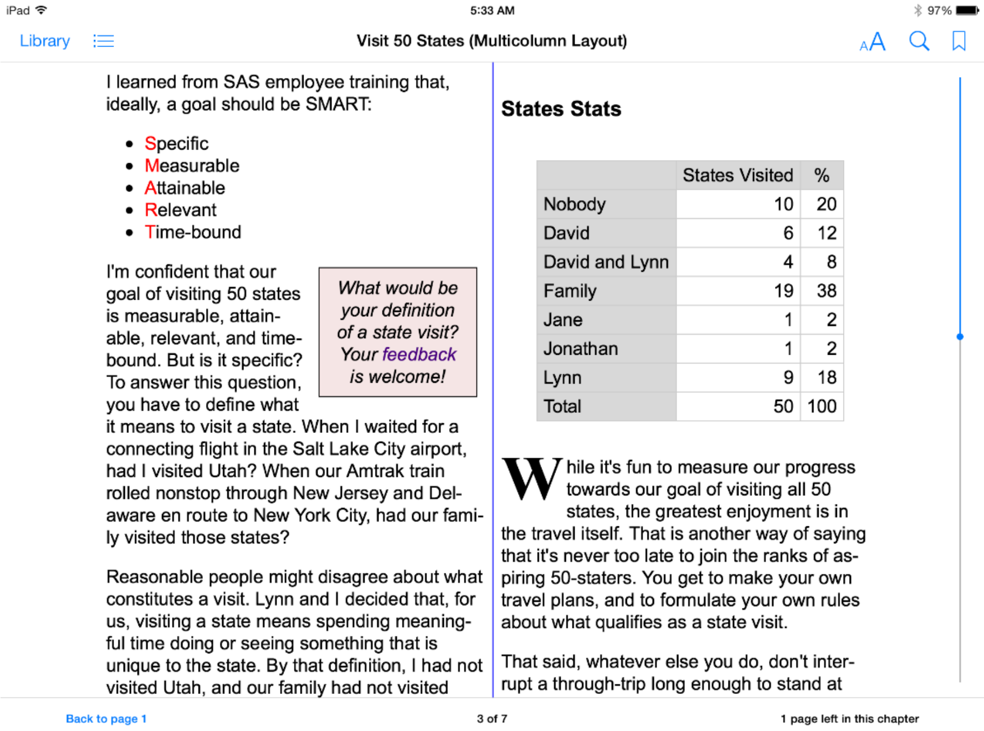
```
data _null_;
   file custom;
   put "p.dropcap:first-letter";
   put "{";
   put "  font-family: Times New Roman;";
   put "  font-weight: bold;";
   put "  float: left;";
   put "  font-size: 3.5em;";
   put "  margin-top: -.25em;";
   put "  margin-right: .05em;";
   put "}";
run;
```

Unsurprisingly, I follow up the CSS code with PROC ODSTEXT code:

```
proc odstext contents="";
```

```
   p "While it's fun to measure our progress towards our goal of visiting all 50
states, the greatest enjoyment is in the travel itself.  That is another way of
saying that it's never too late to join the ranks of aspiring 50-staters. You get
to make your own travel plans, and to formulate your own rules about what qualifies
as a state visit."/style={class='dropcap'};
   run;
```

The STYLE option applies the *dropcap* CSS class to the first letter of the paragraph. Figure 9 shows how iBooks renders the drop cap.


iPad
5:33 AM
97%
Library
Visit 50 States (Multicolumn Layout)
AA
Back to page 1
3 of 7
1 page left in this chapter

I learned from SAS employee training that, ideally, a goal should be SMART:

- Specific
- Measurable
- Attainable
- Relevant
- Time-bound

What would be your definition of a state visit? Your feedback is welcome!

I'm confident that our goal of visiting 50 states is measurable, attainable, relevant, and time-bound. But is it specific? To answer this question, you have to define what it means to visit a state. When I waited for a connecting flight in the Salt Lake City airport, had I visited Utah? When our Amtrak train rolled nonstop through New Jersey and Delaware en route to New York City, had our family visited those states?

Reasonable people might disagree about what constitutes a visit. Lynn and I decided that, for us, visiting a state means spending meaningful time doing or seeing something that is unique to the state. By that definition, I had not visited Utah, and our family had not visited

States Stats

	States Visited	%
Nobody	10	20
David	6	12
David and Lynn	4	8
Family	19	38
Jane	1	2
Jonathan	1	2
Lynn	9	18
Total	50	100

While it's fun to measure our progress towards our goal of visiting all 50 states, the greatest enjoyment is in the travel itself. That is another way of saying that it's never too late to join the ranks of aspiring 50-staters. You get to make your own travel plans, and to formulate your own rules about what qualifies as a state visit.

That said, whatever else you do, don't interrupt a through-trip long enough to stand at

Figure 9: Drop Cap in iBooks

There is more than one way to set up a drop cap with CSS and ODS EPUB3. Another approach would be to use the STYLE inline formatting function to apply the CSS class to the first letter of the paragraph. However, that would have the undesirable effect of making the text less accessible. Whichever way you choose will necessarily involve trial and error in defining the CSS, due to font–specific sizing and placement behaviors. In general working with CSS can be tedious and time-consuming if, like me, you're not an expert at it. But hopefully you've seen that it's well worth the effort.

MEA CULPA

I realized as I was writing this paper that the 3-step approach outlined at the beginning of the discussion is needlessly complicated. Worse, it can cause problems with custom-defined style elements in certain situations. I won't go into details, but I will tell you that I have simplified the approach for the second maintenance release of SAS 9.4 . The first and second steps no longer are required. Simply specify your custom CSS with the ODS EPUB3 STYLESHEET option, and you're done.

We in R&D often find bugs in our software when we're writing documentation about the software. This time is no exception!

AN EMPHASIS ON ACCESSIBILITY

CSS generally aids accessibility in e-books by separating presentation from content. However, if the presentation substitutes for semantic enrichment of the content, then accessibility is impaired. To see what I mean, consider movie dialogue. In keeping with the family travel theme, I've chosen dialogue from the 1983 comedy *Vacation*. To wit:

Car Salesman: Wait a minute, I'll get to the bottom of this, we'll find your car. Davenport!

Davenport: Yes, Mr. Ed.

Car Salesman: Where is Mr. Griswold's Sports Wagon?

Davenport: I don't know sir.

Car Salesman: I know what must have happened. It didn't come in. Now I can get you the Sports Wagon; the only problem is that it may take six weeks. I owe it to myself to tell you that if you're taking the whole tribe cross-country, the Wagon Queen Family Truckster is the way to go. You think you hate it now, but just wait until you drive it.

Clark: Ed, I'm not your average everyday fool. Now I want my super sports wagon now, or I'm gonna take my business elsewhere!

Car Salesman: Davenport! Get Mr. Griswald's car and bring it back here!

[the car is in the process of being smashed. It returns flat, and Clark tries to squat and open the door]

Sighted readers read the dialogue and immediately recognize that bold text signifies the character speaking, normal text signifies the character's speech, and italicized text signifies action. What if the reader is not sighted? If the dialogue text is presented using CSS, then assistive technologies have no opportunity to identify the varying emphases in it. CSS has not done the reader any favors.

To enhance accessibility, when writing this type of text in an ODS EPUB3 e-book, do not use CSS. Instead, take advantage of four inline formatting functions provided specifically for that purpose:

1. EMPHASIS – for vocal stress
2. STRONG – to convey importance
3. ITALIC – to indicate an alternate voice or mode, as well as for semantic uses (for example, identifying names)
4. BOLD – for keywords and other semantics uses, but conveys no voice or mood change

The above definitions come from *EPUB 3 Best Practices* (Garrish and Gyling, 2013).

Here is ODS EPUB3 and PROC ODSTEXT code that applies the BOLD and ITALIC inline functions to the *Vacation* dialogue text:

```
title;
ods epub3 file="dialogue.epub" title="Emphasis in Dialogue";
proc odstext contents="";
   p "^{bold Car Salesman}: Wait a minute, I'll get to the bottom of this, we'll
find your car. Davenport!";
   p "^{bold Davenport}: Yes, Mr. Ed.";
   p "^{bold Car Salesman}: Where is Mr. Griswold's Sports Wagon?";
   p "^{bold Davenport}: I don't know sir.";
   p "^{bold Car Salesman}: I know what must have happened. It didn't come in. Now
I can get you the Sports Wagon; the only problem is that it may take six weeks. I
owe it to myself to tell you that if you're taking the whole tribe cross-country,
the Wagon Queen Family Truckster is the way to go. You think you hate it now, but
just wait until you drive it.";
   p "^{bold Clark}: Ed, I'm not your average everyday fool. Now I want my super
sports wagon now, or I'm gonna take my business elsewhere!";
   p "^{bold Car Salesman}: Davenport! Get Mr. Griswald's car and bring it back
here!";
   p "^{italic [the car is in the process of being smashed. It returns flat, and
Clark tries to squat and open the door]}";
run;
ods epub3 close;
```

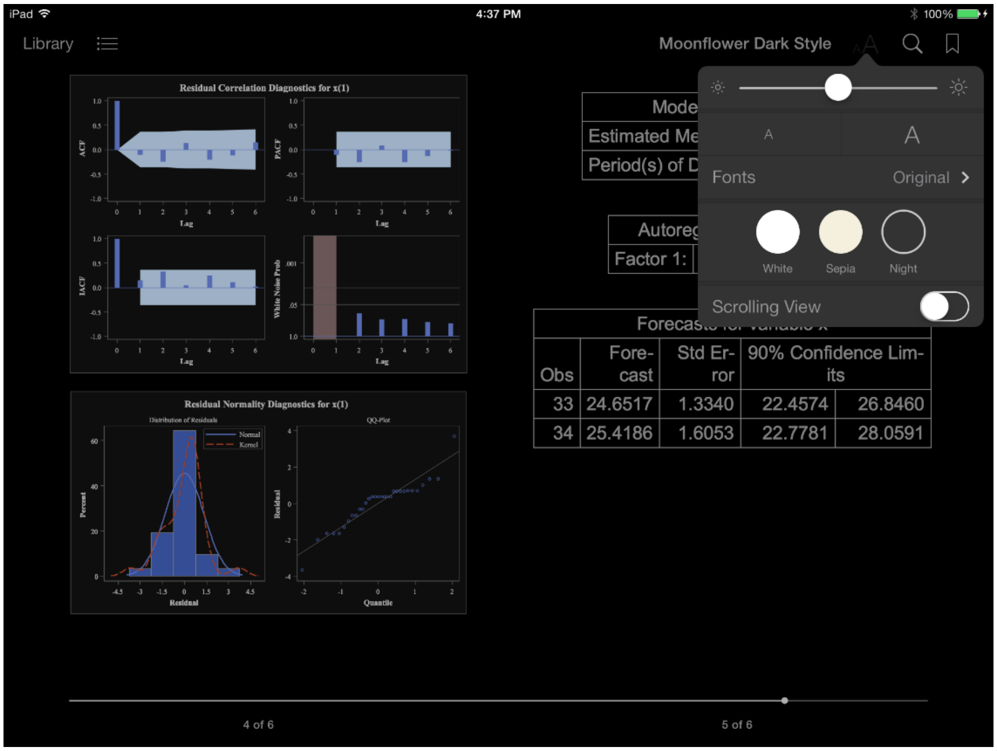
iBooks presents the text in the same way that it would if it had been styled with CSS. The internal HTML5 representation of the text in *dialogue.epub* is different, however. The emphasis information is encoded in the text such that assistive technologies can use it. All readers benefit!

MOBILITY NIGHT AND DAY

The ODS EPUB3 destination is designed with mobility in mind. Although this paper has focused on ODS EPUB3 e-book content and presentation, we cannot forget that the foundational feature of ODS EPUB3 e-books is that they can be read wherever and whenever the reader chooses. This includes low-light situations such as night reading.

iBooks, being up to the challenge, provides a night reading theme called *Night.* *Night* works OK but not great with ODS EPUB3 e-books created with the default style, *Daisy*. Text and tables are presented with an appropriate reverse-video effect. Statistical graphics are the hang-up. Because *Daisy* is designed to be a light style, the graph portion of the style defines images to have a light background. When displayed with the *Night* theme, *Daisy* statistical graphs stick out like a sore thumb.

To provide for an acceptable user experience during low-light reading, for the first maintenance release of SAS 9.4, we added an ODS EPUB3 dark style named *Moonflower*. The graph portion of *Moonflower* defines images to have a dark background, which works great with the *Night* theme. Figure 10 shows what this looks like for PROC ARIMA graphs and tables.



Forecasts for Variable x				
Obs	Fore-cast	Std Er-ror	90% Confidence Lim-its	
33	24.6517	1.3340	22.4574	26.8460
34	25.4186	1.6053	22.7781	28.0591

Figure 10: iBooks Night Theme with ODS EPUB3 Moonflower Style

The ODS EPUB3 and PROC ARIMA code follows:

```
data a;
   input x @@;
cards;
01 02 03 02 03 04 05 04 06 07 06 08 09 10 10 10
09 11 11 13 17 13 14 15 15 16 18 19 20 21 23 24
;
title;
ods epub3 file="moonflower.epub" title="Moonflower Dark Style"
          style=moonflower;
proc arima data=a;
  identify nlag=6 var=x(1) outcov=cov1;
run;
  estimate p=1 outest=est1 outmodel=mod1 outstat=stat1;
run;
  forecast lead=2 out=out1 alpha=0.10;
run;
quit;
ods epub3 close;
```

CONCLUSION

The ODS EPUB3 destination enables you to think about reporting in a new way. You can enrich your content with images, audio, and video. You can lay out your content in multiple columns. You can add publication-style features such as sidebars and drop caps to your output. You can make your report accessible to all of your readers. With ODS EPUB3, your reports can tell a compelling story, like any great e-book does. Put ODS EPUB3 to work for you and write your own masterpiece!

REFERENCES

- Kelley, David W.; Langston, Julianna; Summers, Ed. 2013. "Go Mobile with the ODS EPUB Destination." *Proceedings of the SAS Global Forum 2013 Conference*. Cary, NC: SAS Institute Inc. Available at http://support.sas.com/resources/papers/proceedings13/368-2013.pdf.

- Garrish, Matt; Gylling, Markus. 2013. *EPUB 3 Best Practices*. Sebastopol, CA: O'Reilly Media, Inc.

ACKNOWLEDGMENTS

The author would like to thank SAS coworkers Mike Baker, Robin Crumpton, Tim Hunter, Bari Lawhorn, Susan Owenby, and Chevell Parker for carefully reviewing the paper.

RECOMMENDED READING

- Conboy, Garth, et al. "EPUB 3 Overview." *International Digital Publishing Forum.* October 11, 2011. Available at http://www.idpf.org/epub/30/spec/epub30-overview.html.

- "EPUB 3 Structural Semantics Vocabulary." *International Digital Publishing Forum*. Available at http://www.idpf.org/epub/vocab/structure/.

- Kerscher, George, et al. "Open Publication Structure (OPS) 2.0.1 v1.0.1." *International Digital Publishing Forum.* September 4, 2010. Available at http://www.idpf.org/epub/20/spec/OPS_2.0.1_draft.htm.

- Kerscher, George, et al. "Open Packaging Format (OPF) 2.0.1 v1.0.1." *International Digital Publishing Forum.* September 4, 2010. Available at http://www.idpf.org/epub/20/spec/OPF_2.0.1_draft.htm.

- Kerscher, George, et al. "Open Container Format (OCF) 2.0.1 v1.0.1." *International Digital Publishing Forum.* September 4, 2010. Available at http://www.idpf.org/epub/20/spec/OCF_2.0.1_draft.doc.

- Altheim, Murray, et al. "XHTML 1.1 – Module-based XHTML – Second Edition." *W3C*. November 23, 2010. Available at http://www.w3.org/TR/xhtml11/.

- Bos, Bert, et al. "Cascading Style Sheets Level 2 Revision 1 (CSS 2.1) Specification." *W3C*. June 7, 2011. Available at http://www.w3.org/TR/CSS2/.

- Berjon, Robin, et al. "HTML5." *W3C*. August 6, 2013. Available at http://www.w3.org/TR/html5/.

- Berners-Lee, T., et al. "RFC 3986." *Network Working Group*. January, 2005. Available at http://tools.ietf.org/html/rfc3986.

- Rivoal, Florian, et al. "Media Queries." *W3C*. June 19, 2012. Available at http://www.w3.org/TR/css3-mediaqueries/.

- "Safari CSS Reference." *Apple Inc.* July 23, 2012. Available at https://developer.apple.com/library/safari/documentation/appleapplications/reference/safaricssref/Introduction.html.

CONTACT INFORMATION

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

David W. Kelley
SAS Institute Inc.
100 SAS Campus Drive
Cary, NC 27513-2414, USA
David.Kelley@sas.com
http://www.sascommunity.org/wiki/Advanced_Mobile_Reporting_with_the_ODS_EPUB3_Destination