

SAS3427-2019

A Practical Guide to Responsive Reactive Design Using SAS® Visual Analytics

Elliot Inman, Olivia Wright, and Mark Malek, SAS Institute Inc., Cary, NC

ABSTRACT

While it may seem as if you need to be an artist to create the kinds of beautiful, interesting, interactive visualizations you see on many commercial websites, you don't. All you need is a basic understanding of how HTML5 works and how human beings process visual information. In this paper, we provide guidelines for using SAS® Visual Analytics to create websites that are responsive and reactive. Responsive design relies on HTML5 technologies to dynamically adjust HTML content to the screen size and orientation of a web-connected device. This enables websites to work well on many different devices, but it can cause problems. We present guidelines that reduce trial-and-error testing and describe common responsive design issues (resized legends, squished graphs, and more). We show how to easily test the responsiveness of a report using web developer views built into Google Chrome and Mozilla Firefox, and we provide warnings for some known issues with different browsers. Reactive design focuses on how a website responds to users' interactions, in particular, the speed and sensibility of response to human input. We describe how to implement a reactive design creating a smooth workflow of finger swipes or mouse clicks, how to use white space and negative space to draw users' attention, how to use color in headers and graphs to associate related content, and other tips for enabling users to maintain a mental map of dynamic content, quickly accessing the information they need to know.

INTRODUCTION

SAS Visual Analytics 8.3 uses HTML5 to render reports. For web developers who code in HTML, write CSS, and use frameworks like Bootstrap to develop web-based applications, HTML5 enables development of highly responsive web content. With HTML5, text, pictures, and other content automatically resize and rescale for whatever interface is being used from desktop monitor to smart phone, tablet, and so on.

Data scientists often have a broad skill set of statistical, programming, and subject matter expertise, but they are not necessarily web developers. One advantage of SAS Visual Analytics for data scientists is that they don't have to become web developers to deliver high-quality professional web-based reporting. By following a few simple guidelines, data scientists and other report developers can create web-based reporting that takes advantage of HTML5 functionality. This paper describes how to transition a set of traditional tab-based reports into a responsive scrolling report that will look and respond like modern websites. All of this functionality is standard in SAS Visual Analytics.

In this paper, the use case is child social services data. The data includes nine years of state child services metrics, including the number of children served by state child services, the number of children adopted and still waiting to be adopted, and some data on financial subsidies for adoptive parents. The purpose of these reports is to fulfill data transparency requirements for federal regulations regarding the collection of state-level data on child

services. We should note that the authors of this paper have created all the reports in this paper, including both the typical “before” and improved “after” reports.

After a description of the data, we show how to modernize a set of traditional reports, including guidelines for focus, flow, standardization, and simplification of disjoint content into a single, flowing responsive report. We describe how users can go beyond the basics and experiment with more unusual ways of creating reactive reporting. We then discuss some of the challenges of working with HTML5.

THE DATA (U.S. FOSTER CARE AND ADOPTION DATA)

In this paper, the data is public data from the United States [Administration on Children and Families](#) (ACF), a division of the United States Department of Health and Human Services. Links to all the raw data are provided in the References section of this paper.



The data includes state-level data for all 50 states, Puerto Rico, and the District of

Columbia for children served by state social services programs. The data collection program is called the Adoption and Foster Care Analysis and Reporting System (AFCARS). AFCARS data includes state-level data on children entering and exiting child services, adopted and waiting children, and other metrics. The AFCARS data is available in Excel format [here](#). As of the publication of this paper, the national data set included state-level data from fiscal years 2008 to 2017. For technical definitions of the variables used in these reports, readers can review the technical notes from ACF in [their data summary](#).

In addition to the historical data set made available, ACF makes available data on subsidies provided to prospective adoptive parents to assist in managing the needs of potential adoptees. This is an important source of support for high needs children for whom adoption might carry a significant financial challenge for parents. This subsidy data was extracted from PDF reports and merged by state and year with the AFCARS data. Subsidy data from 2008 to 2016 is available, but no data is available for 2017. When merged with the historical data set from AFCARS, the final data structure excluded all data from 2017. Thus, the final data set included all AFCARS national data set and subsidy data for the years 2008 to 2016.

Two additional variables were calculated: the Adopted/Waiting ratio and the Exiting/Entering ratio. Each metric is the simple ratio of two counts calculated to provide a comparable measure across states that vary significantly in population. The Adopted/Waiting ratio is the ratio of adopted children to children still waiting for adoption in that fiscal year for that state. The ratio can exceed 1.0, suggesting that there were more children adopted than waiting, but the ratio is the number of children adopted that year relative to the number of children *still waiting* for adoption at the end of that year. More children could have entered the system than exited.

The Exiting/Entering ratio was calculated the same way. This ratio is a broader measure of all children served by that state’s child services division, including those children temporarily removed from a home, reunited with family or other relatives, and otherwise served by various programs for children in the state’s care. Again, this ratio can exceed 1.0 for the same reason previously described.

For SAS Visual Analytics reporting, the wide data structure with years and variables repeated in columns was transformed into a tall data structure with years and variables in

rows. In the following examples, we show data for several specific states. These states were not chosen for any particular reason.

MODERNIZING TRADITIONAL REPORTING

Figure 1 is a typical example of tab-based reporting. There are five tabs of reports; three tabs are shown.

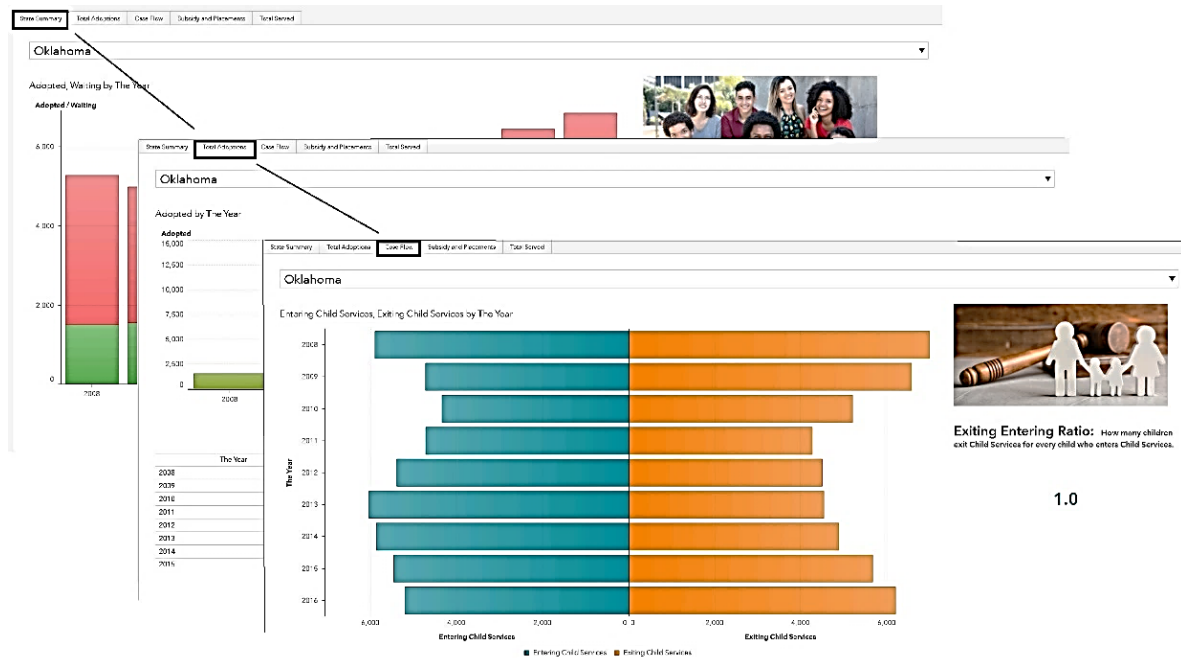


Figure 1. Traditional Reporting

There is nothing fundamentally wrong with this structure. It works. Users know to skip from tab to tab to view individual reports, each of which has a descriptive unique name. There is some notable redundancy over the reports, including repeated state drop-down selections and duplicate tables, but the reports could be linked to allow for a single state selector that filters all reports and a single data view tab for the data. That could be done to improve the usefulness of these reports. However, if you compare this reporting to many modern websites, you will see two obvious differences.

The first difference is how modern websites capture *focus* with the shape and positioning of the content. Most desktop monitors are no longer square in shape; they are rectangular. But many modern websites don't actually fill that rectangular space from left to right. Substantial empty "white space" margins draw the content into the center. This is, in fact, exactly where most users are looking. Users do not read a rectangular monitor as they once read a printed book or newspaper starting top left and working down. For many data scientists, thinking about the shape of reporting might seem strange, but after you compare your reporting to popular modern websites, you realize how much web-based reporting has changed.

The second difference is the *flow* of the content. In the early days of software development when the computer mouse was first being used in conjunction with a keyboard, software

developers were cautioned to minimize the number of physical leaps required to jump from two-hands typing to one hand on a mouse. Only the most poorly designed interface would require a user to constantly switch back and forth. In the modern age, the equivalent advice might be to minimize the number of times a user has to click versus scroll using a mouse. Modern websites flow downward with additional content becoming available via a scrolling mouse or, in the case of a smart phone, a swipe of the finger or thumb.

Figure 2 shows a set of tab-based reports on the left and the single scrolling report on the right. Each of the tab-based reports requires a state selection at the top. There is duplication of one type of graph for two different purposes. A waterfall plot is used for Children Adopted and, on another tab, for Children Served. Following tab order, the reports seem to bounce from a more detailed metric (Percent Subsidized) to a more general one (Children Served). And some information is redundant, for example, the first tab includes Children Adopted and Children Waiting in the stacked bar chart, but the first waterfall plot shows Children Adopted.

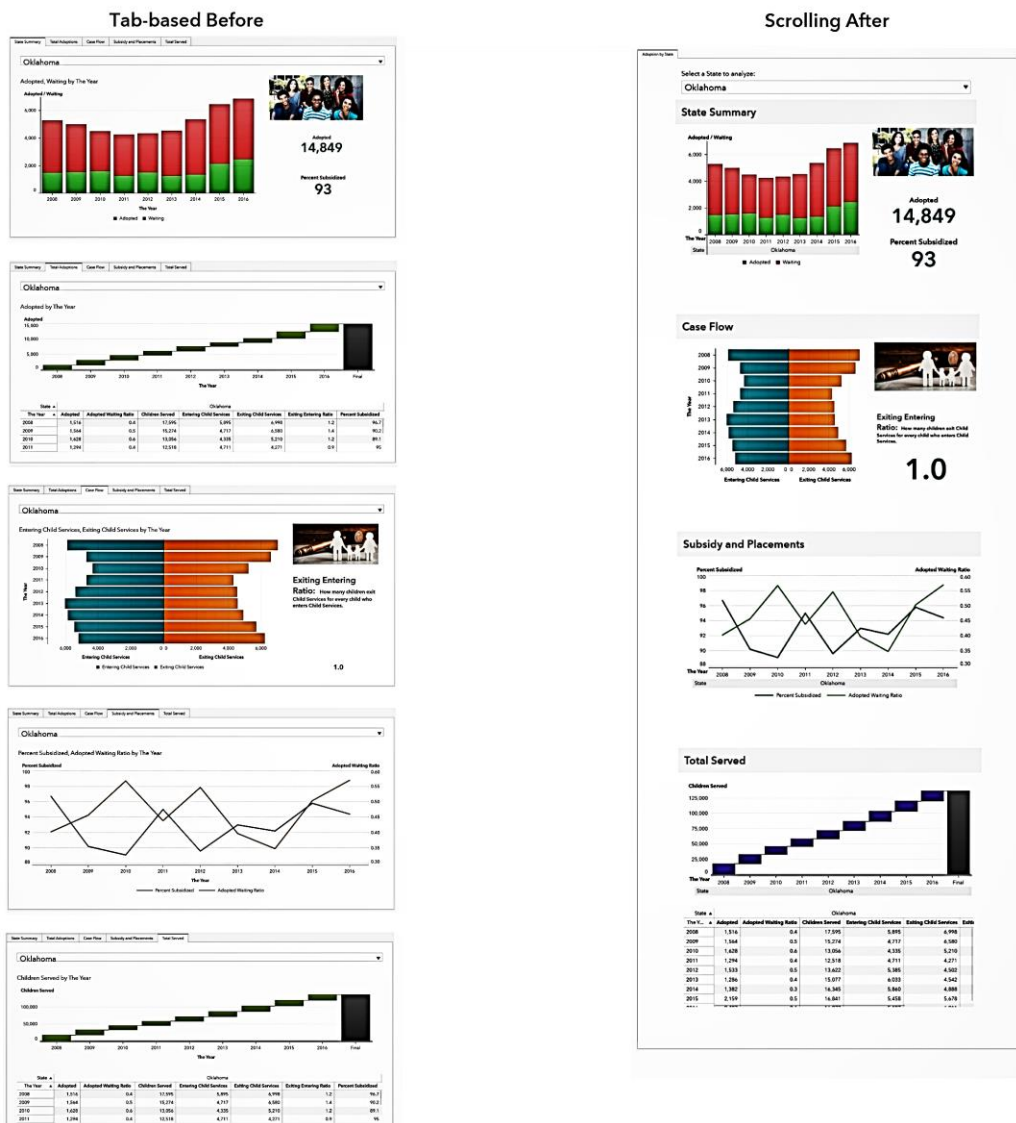


Figure 2. Before and After Reporting

Figure 3 shows some of the major changes made in restructuring the reporting into a more responsive, reactive design.

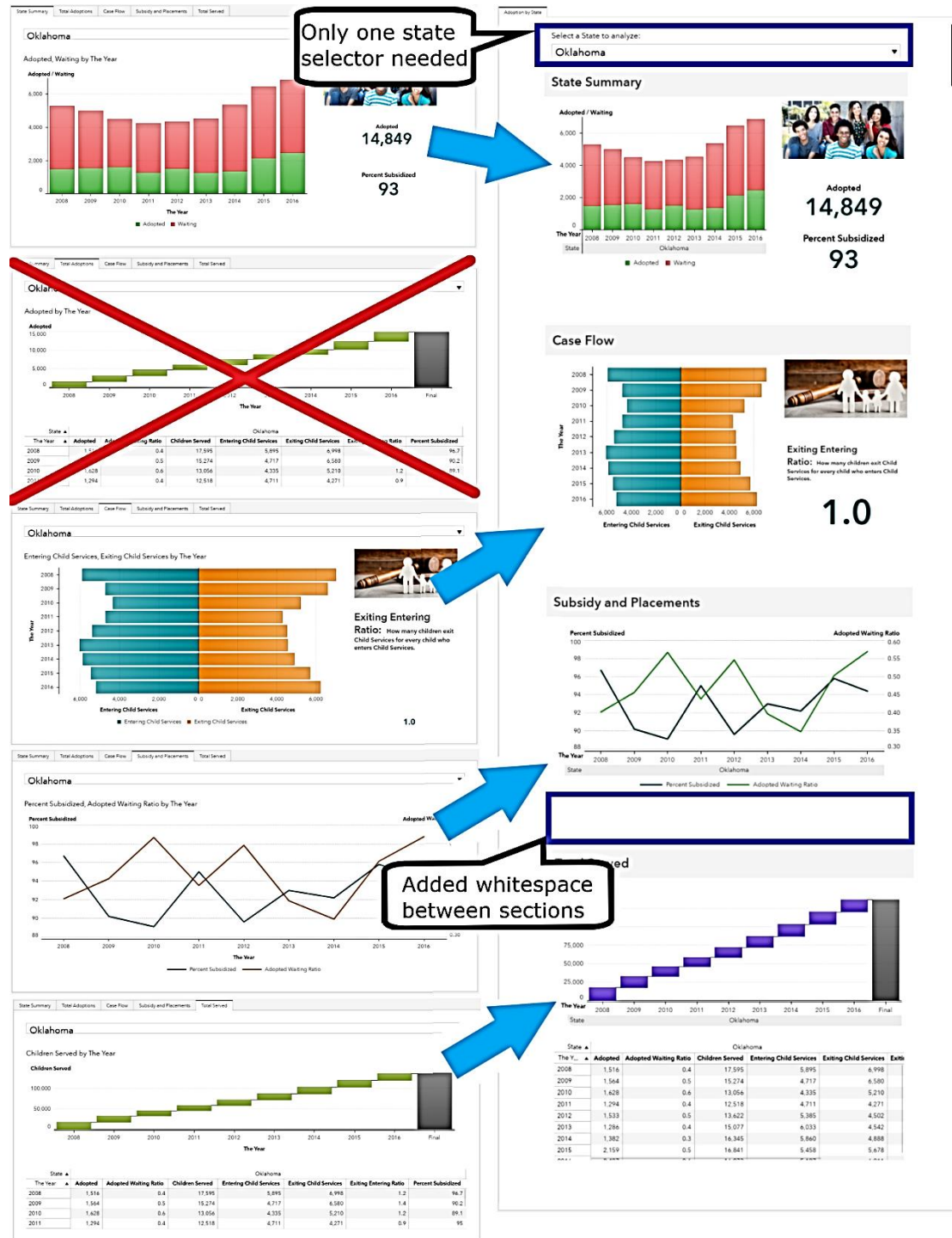


Figure 3. Major Design Changes

The following guidelines describe the steps to create the report in SAS Visual Analytics.

OPTIMIZE FOCUS AND FLOW

Guideline	Implementation in SAS Visual Analytics 8.3 (and later)
Enable Scrolling	Create a new blank tab. In Report > Options > Layout, uncheck the Avoid Scroll Bars check box. This one step will enable you to stack one report on top of another into a scrolling structure.
Create Graphic Organizers	In the scrolling report, users no longer have the benefit of tab titles. For each tab in the old report, create a heading in the new report. For ours, we have used text boxes with a gray background and a relatively large 22pt font. Use the same consistent size and style for each "page break" in your scrolling report. Note how "State Summary," "Case Flow," "Subsidy and Placements," and so on provide a dynamic type of section break.
Build Margins	Create margins by adding empty text boxes. For the text box, under Options > Layout > Width, we used 15% for the specified width. Select the Extend width if available and Shrink width if necessary options to allow HTML5 to dynamically adjust the sizing. The new report now draws the user's focus to the center of the screen with white space margins on the side.
Increase Whitespace	Add empty text boxes to separate content. For these reports, under Options > Layout, we used 10% height. The width of the text box was set to 70%, which is the width minus the margins. When scrolling through the report, users need some type of visual cue of a transition. The gray headers and white space between visualizations help distinguish the topic for each visualization.
Order the Reporting	On the left side of the SAS Visual Analytics report development window, there are three icons: Data, Objects, and Outline. The Outline View enables you to insert the previously developed tab-based objects in the new desired order by dragging and dropping them into the new page.
Maximize the View	Although the tab reporting structure is gone, you want users to be able to view each visualization in the greatest possible detail all at once, filling the monitor screen even as they scroll. Most of the visualizations here have a height set at 70%, leaving some room for the header and white space separating reports. The height specification ensures that the rendering of the report will make enough room for the visualization, regardless of the screen.

Table 1. Guidelines for Focus and Flow

The previous steps will transform a set of tab-based reports into a single flowing scrollable report. However, simply combining the individual reports can create some expected problems: repeated selections, content that appears out-of-place, and report design inconsistencies that might not have been noticed before. The following guidelines should help to reduce these issues.

STANDARDIZE AND SIMPLIFY

Guideline	Implementation
De-duplicate	Remove duplicate visualizations that convey the same information. The "Total Adoptions" tab is not needed as adoptions are on the previous graph. Remove duplicate prompts if possible. One state selector at the top filters all reports.
Delete Unnecessary Content	Remove any titles and legends that are not necessary. Look at the bottom of the "Children Served" report . The original tab-based report included the graph title "Children Served by the Year," but that title is omitted in the new design. The content of the visualization should be obvious from the X-axis and Y-axis labels and, without that title, there is more room for the data. When rendering the report, browsers will no longer have to make room for that title.
Validate Interactions	Ensure that tab-specific interactions still work and are reasonable for that visualization. A single scrolling report can create a set of complex interactions. For this report, the graphs still provide interactive content, but there is only one main selection at the top of the report for state. That one selector filters all visualizations in the scrolling report.
Leave a Trail	One of the benefits of an initial selection is that the user's choice is immediately active in working memory. The user just made a choice and expects a filtered report. However, users returning to the same report might not realize that the state selection affects all reporting. For that purpose, we have added lattice columns where possible. The "State Summary" and "Subsidy and Placements" sections use the lattice. There will never be more than one state on the lattice, but it is helpful to remind users that the state filter is on.

Guideline	Implementation
Minimize Hidden Information	Suppose a user never interacts with any aspect of the report after making an initial selection. Is all of the information required to understand the visualizations actually available? There is a tendency of some report developers to rely too heavily on hover-overs and other interactive features to present critical information. Hover-over functionality should be used only to answer two questions: What is this entity and what is the value? If additional information is critical for the user, it should be available elsewhere on the report.
Standardize Color	Tab-based reports might have reused colors to represent certain variables or conditions. This should be avoided in any reporting process, but it is easily overlooked, especially with a large number of tab-based reports. In a scrolling report, this problem exacerbates potential confusion. The "Children Served" section was changed to a more unique color to avoid confusion with other greens in the report.
Scaffold General to Specific	Visualizations that summarize trends should appear at the top of the scrolling report. Detailed information like text explanations and data tables should appear at the bottom.

Table 2. Guidelines to Standardize and Simplify

By using the method described, data scientists with little or no formal training in web development or HTML5 coding can create reports that look much more like modern, responsive, reactive websites. However, the shift from a traditional to modern flow and focus is just the start. Data scientists interested in going further can experiment with additional ways of implementing even more unusual reporting.

EXPERIMENTING WITH REACTIVE DESIGNS

Any user would agree that simply tabulating all of the subsidy data for every state every year in a single PDF would result in a very hard-to-digest report. The facts would be accurate and reporting requirements could be fulfilled, but it would be a report not easily understood. Figure 4 shows how some of this data is currently reported.

for at-risk youth, but there is nothing about longitude and latitude or geographic proximity that has any meaning here. In fact, the reason we are using *rates* of subsidized adoption is because *counts* of subsidized children would show California, Texas, and Florida different from other states based only on population size. Counts would be misleading in the same way that the sheer square area of land is not meaningful. The map is convenient and perhaps even expected, but it is not very useful and might actually distort the meaning of the metric.

The second weakness is the fact that the additional data is presented at both too high and too low a level to be easily understood. The bar chart provides the data for all states combined or one state at a time, but the chart provides no opportunity to compare states over time. Conversely, the data table provides so much detailed information that it would be difficult to glean any insight from it. Years are in rows, but states are in columns, which makes comparisons impossible.

Beyond those problems, the subdued green coloring does little to interest any user. You could argue that color choice reflects the seriousness of the topic, but the subdued color does not catch the attention of a viewer. Again, the report is not wrong, just not very helpful and definitely not very interesting.

Figure 6 is a radically different version of the same report. This report uses a time slider at the top to allow users to select any period of time between 2008 and 2016. Despite the fact that the bars below the time slider are purple and not a black-and-white ASCIImatic, they are indeed a box plot. And below that is a word cloud with the data in row order. In the same amount of space and using the same data as the traditional report, we have a completely different design.

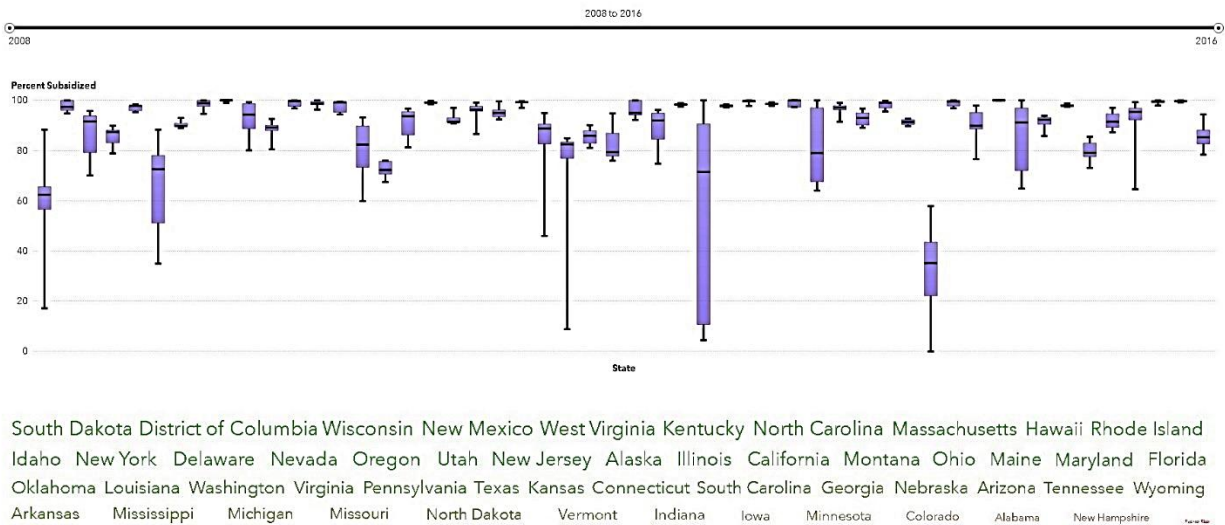
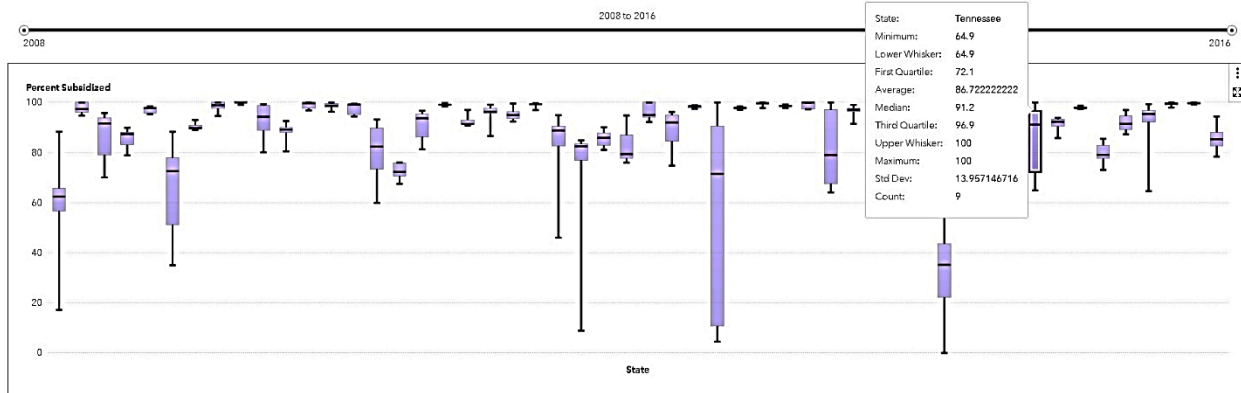


Figure 6. Reactive Report

The map is gone, as are the state name tick marks on the box plot. In the word cloud, size and color were both mapped to the Percent Subsidized variable such that a larger state name in green indicates a higher percent of subsidized adoptions. No formal training is needed to understand this report as it is completely interactive.

As Figure 7 shows, selecting a bar selects the state and provides the box plot data associated with it.



Tennessee

Figure 7. Top Down: Selecting a Bar

Figure 8 shows how selecting a state from the word cloud below selects the state in the box plot.

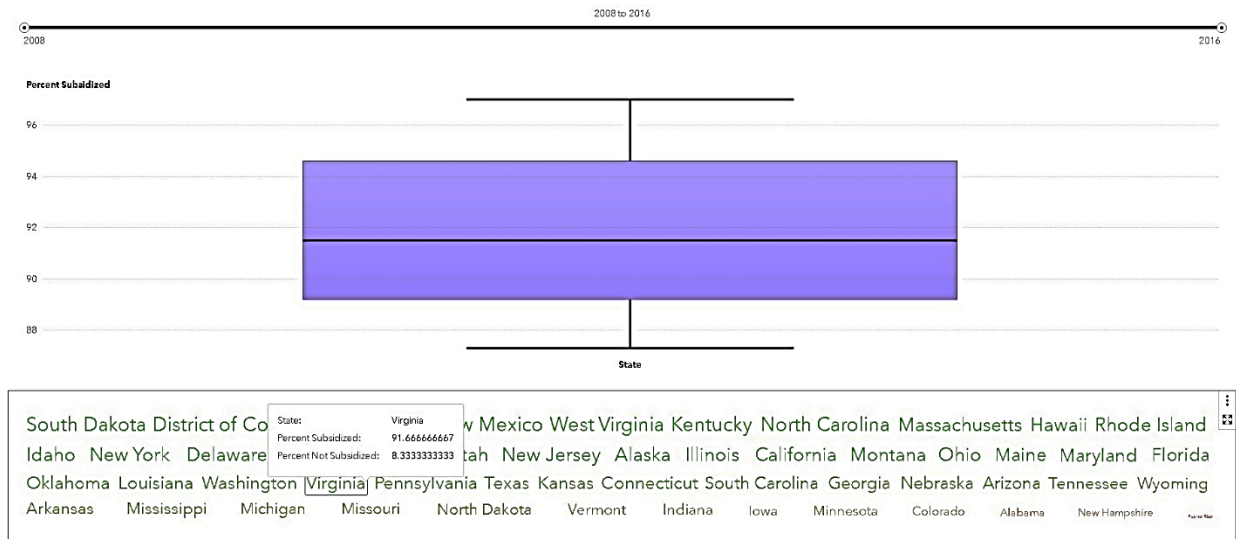


Figure 8. Bottom Up: Selecting a State

While this report occupies a single screen, a few additional aesthetic changes can make the report even better. For example, by adding a background color to the time slider and a

color border around the box plot and word cloud, it becomes absolutely clear that the three objects are related. In the case of the time slider, the background color is a significant improvement and keeps that single straight line from becoming lost at the top of the screen.

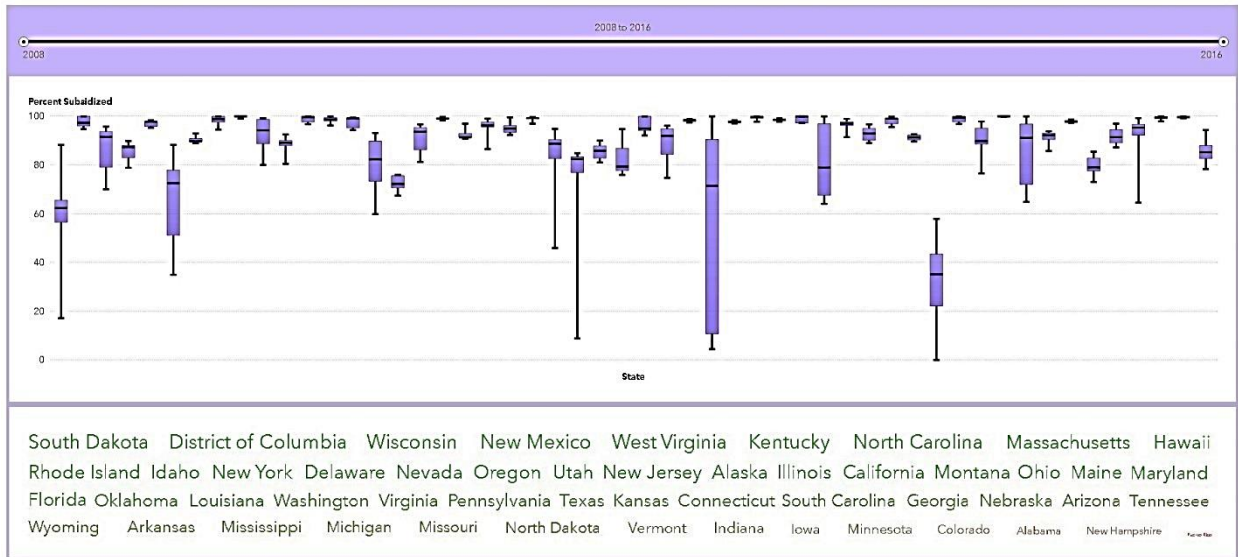


Figure 9. A Color Coordinated Report

While creating a report like this cannot be summarized in a set of next-do-this steps, there are a few guiding principles that can assist in inspiring your experimentation.

Suggestion	Implementation
Eschew the Mundane	Do not repeat visualizations out of mere convenience or habit. The data in these reports is state-level data easily mapped, but there is virtually nothing about the geospatial location of the data that has an effect on the outcomes for foster care children. Why map the data?
Delete, Delete, Delete	Experiment with a stripped-down version of visualizations that do not include titles or axis labels. The states are not listed on the X axis in this visualization, but when you select a bar, you see the state in the hover-over and in the word cloud below.
Misuse the Software	In SAS Visual Analytics, we distinguish “Graphs” from “Controls,” but interactivity means that most graphs can be used as a control. Instead of being used to demonstrate a finding, a word cloud can be used as a control.

Table 3. Suggestions for Experimentation

WARNINGS AND GOTCHAS

SAS Visual Analytics reporting using HTML5 provides a modern platform for data visualization. But the responsive nature of HTML5 and the deployment of reporting on many different shapes and sizes of screens can create some problems.

HTML5 IS RESPONSIVE (IN REAL TIME, ALL THE TIME)

HTML5 will automatically adjust any SAS Visual Analytics reporting based on screen size, user input, and certain aspects of the report, as it tries to fit content onto a screen.

The rise in popularity of rectangular screens for desktop computers prompted the development of web-reporting with wide white space margins to draw users' attention back to the center of the screen. Report developers should, for now, assume a rectangular horizontal screen as the default desktop view. But there are large number of significantly different-sized screens: jumbo wide view screens in conference rooms, tiny smart phone screens held vertically, and a large array of differently proportioned tablet screens. Suffice to say, HTML5 will attempt to adjust reporting to fit those screens, and some aberrations will occur. The best advice is to design on the most common screen you expect users to have and follow the previous guidelines to avoid some pitfalls.

Using the Chrome browser, if you select **Developer Tools**, you will have the option to view whatever web page is loaded in the browser dimensions you choose, as well as on many common devices like smart phones and tablets. This should give you a chance to see how reports under development will respond when rendered with HTML5.

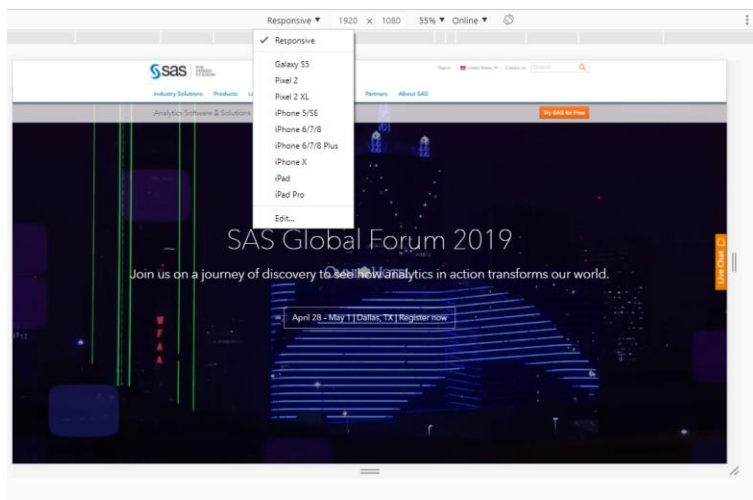
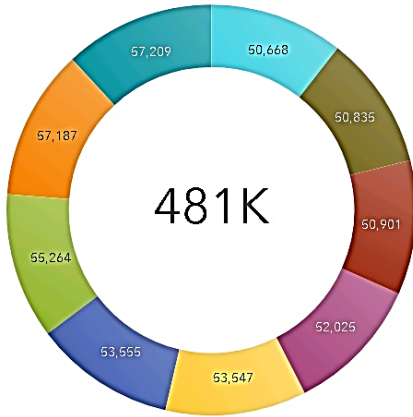


Figure 10. Chrome Emulation of Various Screen Dimensions

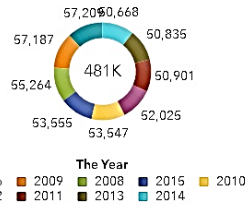
In adjusting for screen size, SAS Visual Analytics attempts to preserve as much meaningful content as possible, but certain content is often lost. Legends for graphs are often abandoned. Pie charts in particular, which have issues of interpretability due to misreading of slice sizes, require significant additional context (titles, callouts, legends) and a lot of additional space around the chart. Figure 11 illustrates the same pie chart rendered in increasingly smaller size and the problems that result. Following the previous guidelines to reduce unnecessary titling and text will help to keep the visualization front-and-center on any screen. Nevertheless, some visualizations like pie charts should be used sparingly.

Adopted by The Year
Adopted



The Year
2016 2009 2008 2015 2010
2012 2011 2013 2014

Adopted by The Year
Adopted



Adopted by The Year



Figure 11. The Incredible Shrinking Pie

NOT ALL BROWSERS ARE ALIKE

One of the first major advances in desktop publishing was the advent of WYSIWIG word processing software in the late 1970s, commercialized in the 1980s. WYSIWIG – what you see is what you get – meant that documents would print exactly as they looked on the screen. HTML5 is definitely not WYSIWIG.

With the rise of the internet, HTML (“hypertext markup language”) provided a common standard for web-based page layout, but dynamic content often rendered inconsistently. The Java Runtime Engine offered the promise of write-once-run-anywhere and provided a common engine for internet multimedia. Later, Adobe Flash/Flex provided an even more adventurous, yet safe space for dynamic visualizations, video, and sound. The current rise of HTML5 offers much of the same promise of these past technologies: HTML5 tags will be processed the same by different browsers and the rendered content will be presented as intended to users. Of course, this is not always true. What you expect to see is not always what appears on the screen.

If your organization has standardized to a particular browser (for example, all desktops have Chrome) and you know users will use that browser, develop and test using Chrome. The same for Firefox, Edge, Safari, and others. The problem is more complex if you are serving content such as public reporting where end users can use whatever they prefer, and this does not begin to account for tablets and mobile devices. The best way to assess browser usage is to audit weblogs of access to your reports, but that often requires more advanced web analytics and IT skills. Report developers who want a quick estimate of the most HTML5 compliant browsers should check [HTML5TEST](#). SAS regularly publishes the results of our own browser testing at [support.sas.com](#).

OTHER CONSIDERATIONS

The majority of this paper has focused on modernizing reporting with very little attention to two issues of importance: usefulness and appropriateness. Those are important questions worth considering. First, do all of the changes suggested here actually make the reporting more useful? Second, are there some situations in which this type of modern reporting is simply not appropriate?

USEFULNESS

Current federal law requires these data to be collected. Gathering the data does involve costs to states to collect, validate, and submit these metrics. Whether that effort is worth the cost is often under reconsideration, as noted in [this 2018 notice](#) to the Federal Register regarding the AFCARS data.

As data scientists, we believe in the importance of gathering, analyzing, and communicating findings from data. Analyzing AFCARS data advances our understanding of how best to meet the needs of at-risk children receiving state social services. But to fulfill the goals of data science, we must report the data in a manner that the general public and researchers can use it. That means the data will be available on the internet. That means that reporting must allow users to interact with the data, not just read a PDF or download a file. Users must be able to ask questions and interrogate the data. A *useful* reporting design allows users to do so.

Why, then, do we need to implement a modern, responsive, reactive design? Because users expect it. Users expect that the data will be supported by reporting that makes the data seem relevant and important to people today.

A modern reporting design reinforces the current present usefulness of the data. As new technologies are developed, we regularly transition old content to new formats. Printed books are scanned; paintings are photographed with digital cameras; and vinyl records are transformed into digital sound. But not all content is carried forward. Not all books or paintings or music are transformed to be a part of our modern digital world. In truth, we import into the future only what we value from the past.

The AFCARS data has always been an important source of information about state efforts to serve at-risk children and find permanent homes for children without them. Modernizing such reporting makes clear that this data and the issue, itself, are still important to us. Simply stated, making these reports more modern will make the data more likely to be used and thus, more modern reporting is more useful.

APPROPRIATENESS

The reporting here is based on anonymized aggregated data at the state level, but the data represent the lives of millions of American children, including those who have been adopted, those waiting for adoption, and the hundreds of thousands of children served every year by state child services agencies. The data represents an objective, straightforward accounting of the positive outcomes and continuing risks for millions of young people.

This is a serious story to tell. Is it possible that more traditional reporting like the map-based report with its subdued greens, technical layout, and stuffed-with-facts data table is more “appropriate” than the interactive report in shades of purple and green? That is a challenging question.

In practice, we have seen data scientists and report developers who pursue an aesthetic designed to evoke the seriousness of the subject: black screens filled with rows of gray

gauges or a half-a-dozen complex graphs of all sorts squeezed onto a single screen – all designed to create the impression of a tremendous amount of critical information, all of which must be processed at once. There is nothing fundamentally wrong with that, if that is what users expect. There are limitations to human cognitive capacity in terms of working memory and attention and those designs can tax the resources of any human mind. At the same time, experts in any field often want and need much more detailed information than novices. Experts expect complexity.

On the other hand, the most “appropriate” design may be the one least like what has been done in the past. Human beings have an innate desire for novelty. The desire for new experiences drives a large part of human exploration and learning. It is why people want to take a vacation to a place they have never been or read a new book by an unfamiliar author or learn a new programming language or, in fact, design a different kind of report. If, by implementing a more unusual set of reports with brighter colors, more reactive designs, and somewhat unusual ways of presenting the data, we make the data new again, we may inspire others to more deeply consider these data. If our goal is to encourage the careful consideration of these data, such a novel design is “appropriate.” The design serves the purpose.

CONCLUSION

This paper presented a set of suggestions for how data scientists and report designers can transform traditional tab-based reporting into more modern, responsive, reactive reporting ready for HTML5 compliant browsers. We described how to take a set of disjointed tab-based reports and integrate them into a single scrolling report that will respond well on a variety of screens. We also made some suggestions for how report designers might take this functionality further, designing very novel visualizations that might inspire users to explore the reporting in greater depth. And we noted some of the challenges posed by auto-adjusting HTML5 and how to anticipate potential problems.

For more information on how to modernize your Visual Analytics reporting, please see [Beautiful Reports](#), a SAS community led by Mark Malek that focuses on visual designs that inspire users. In addition to examples of responsive, reactive [scrolling reports](#), it includes additional content on larger issues such as knowing your audience and ensuring accessibility and technical design content on topics such as formatting text and the use of color.

The AFCARS data used is available as open data. Links to the source data are in the References section of this paper. No matter what the design or technologies used, the ultimate value of reporting rests on the value of the data and issues addressed. The treatment of at-risk youth and efforts to find permanent homes for children in foster care is an issue of enormous importance. The collection and reporting of this data to advance our understanding of these issues is, without a doubt, worthwhile. We hope that, in addition to finding useful guidance on report design, readers will be inspired to think of other ways in which these particular data can be made better known, better understood, and more widely used.

REFERENCES

All links below accessed February 2019.

U.S. Department of Health & Human Services. 2019. <https://www.acf.hhs.gov/>

"Adoption and Foster Care Analysis and Reporting System," Federal Register, Vol. 83, No. 51. p. 11449. Accessed at <https://www.govinfo.gov/content/pkg/FR-2018-03-15/pdf/2018-05042.pdf>

"Trends in Foster Care and Adoption." Available
<https://www.acf.hhs.gov/cb/resource/trends-in-foster-care-and-adoption>

"Trends in Foster Care and Adoption (PDF)." Available
https://www.acf.hhs.gov/sites/default/files/cb/trends_fostercare_adoption_08thru17.pdf

"AFCARS Data Set with Adoption Variables (Microsoft Excel)." Available
https://www.acf.hhs.gov/sites/default/files/cb/afcars_state_data_tables_08thru17.xlsx

U.S. Department of Health & Human Services. 2019. Subsidies data. Available at:

https://www.acf.hhs.gov/sites/default/files/cb/adoption_subsidy2012_2016.pdf

https://www.acf.hhs.gov/sites/default/files/cb/adoption_subsidy_2008.pdf

https://www.acf.hhs.gov/sites/default/files/cb/adoption_subsidy_2009.pdf

https://www.acf.hhs.gov/sites/default/files/cb/adoption_subsidy_2010.pdf

https://www.acf.hhs.gov/sites/default/files/cb/adoption_subsidy_2011.pdf

CONTACT INFORMATION

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

Elliot Inman, Ph.D.
Manager, Software Development
SAS Global Hosting and US Professional Services Division
(919) 531-1717
Elliot.Inman@sas.com

Olivia Wright
Software Developer
SAS Global Hosting and US Professional Services Division
(919) 531-3262
Olivia.Wright@sas.com
Twitter: @OliviaJWright
<https://www.linkedin.com/in/oliviajwright/>

Mark Malek
Visual Designer
SAS Business Intelligence Research and Development
(919) 531-1608
Mark.Malek@sas.com
<https://www.sas.com/beautifulreports>
<https://www.linkedin.com/in/mark-malek-b7a16a8b>

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.