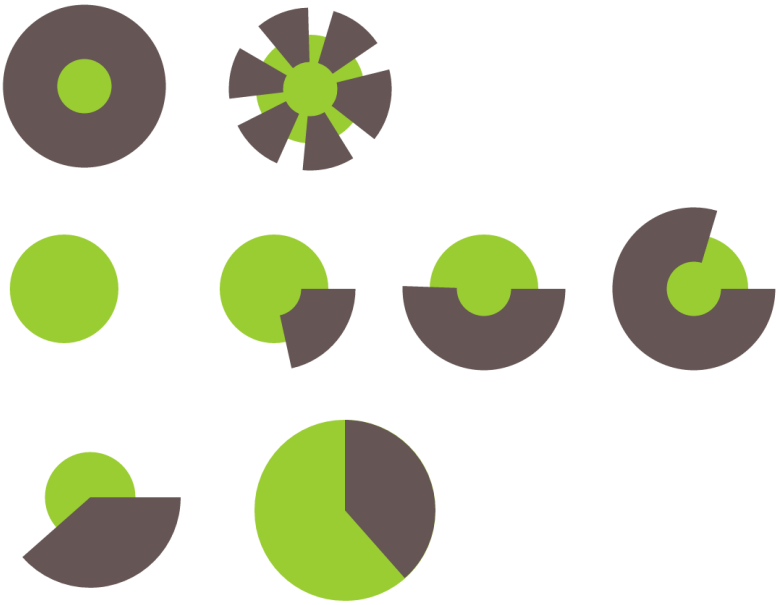
CSS vs SVG: The Final Round(up)



is the last article in the series of article comparing common CSS techniques to their SVG nterparts. This article is a roundup of several CSS and SVG solutions, as opposed to being article comparing one solution that can be achieved using either CSS and SVG. There are unch of excellent articles out there that cover the details for each of these solutions, so we

News

Collections

For You

Topics

Icon fonts are monochrome and can't be multi-coloured—that is, unless you use a workaround which requires you to stack multiple elements of an icon on top of each other to separate the "shapes" making that icon up, and then color each shape individually to make up the complete icon. This solution, though clever, is overkill, in my opinion. But if this idea interested you, you can learn all about it in this article by Parker Bennett.

> ılso have weird failures many browsers, have anti-aliasing issues resulting from the fact that ɔnsider them as text and as such apply anti-aliasing to them, making them look not as sharp ɹpect them to, and they can be frustrating to position.
>
> ɛr has written an excellent and thorough cagematch-style article over at CSS-Tricks that ɔros and cons of each icon system (icon font vs SVG icons). You should definitely check it
>
> ɯe of the advantage of SVG icons, summed up in a few points:
>
> are more semantic. An SVG is an image and the icon is thus marked up as one.
>
> icons look crisp and clear on all screen resolutions.
>
> icons provide you with more control over SVG icon styles (multi-color icons). Individual shapes can be selected and styled using CSS. Sure, there are some limitations depending on ɹay you embed the SVG, but this is going to change in the future. (I wrote an article about

News

Collections

For You

Topics

recommend you read more about them and give them a try.

Charts, charts, charts!

These are one of the interesting UI elements that is getting even more interesting with the introduction of new technologies to create them. Let's take a deeper look.

harts

used to create simple charts that require more "regular" shapes such as rectangles and

s great at creating rectangular shapes, vertical (or horizontal) bar graphs are a great example r charts.

wrote about and demo'd the concept of creating vertical bar graphs using CSS long ago. In he explains how to create bar charts using CSS to style and position elements of an HTML which is used to mark up the graph's data. Eric's technique is great because the data is in a semantic way (tables are the perfect way to store data like that) and the data can be d manipulated using a server-side language like PHP, for example.

lle forked Eric's technique and created a new version of a bar graph using the same

News

Collections

For You

Topics

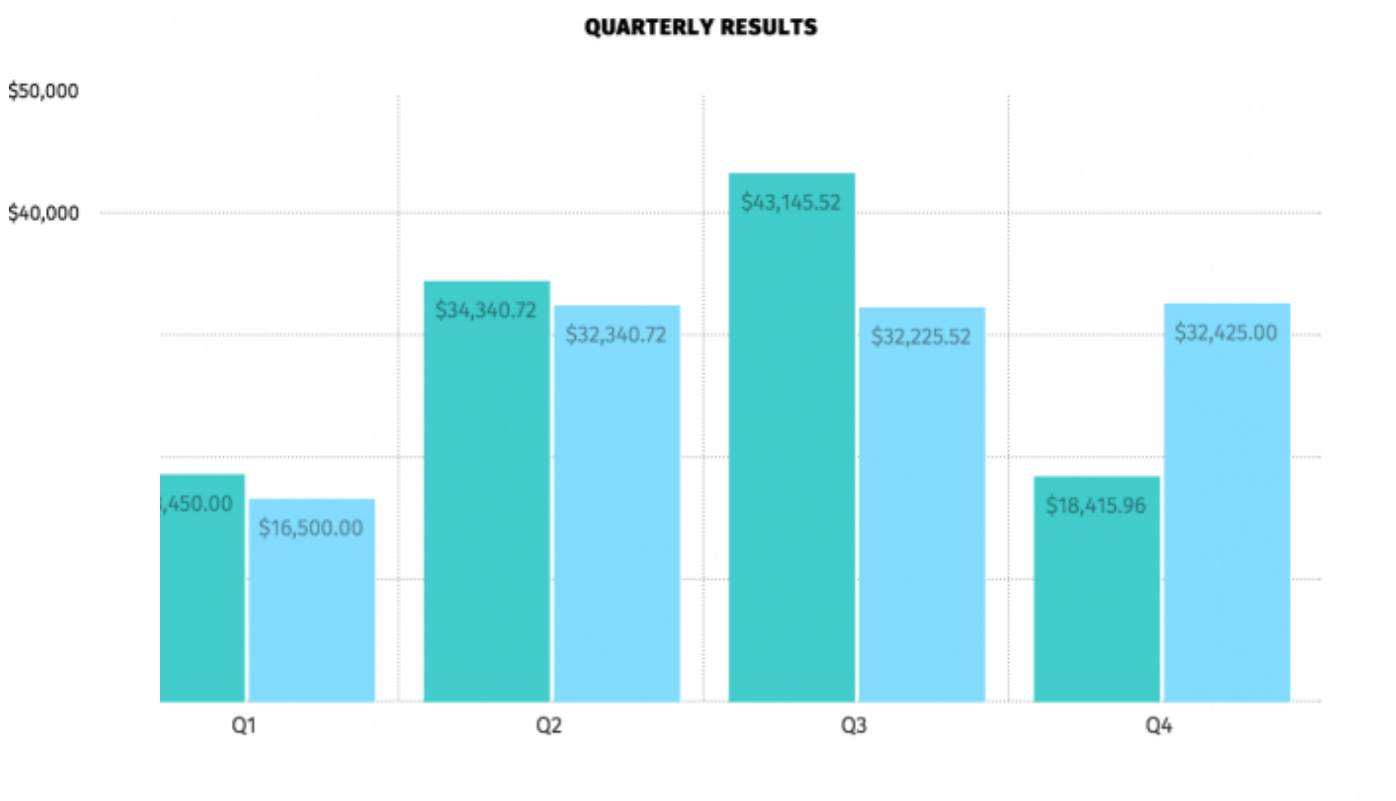
```
        <tr>
         <th></th>
            <th class="sent">Invoiced</th>
            <th class="paid">Collected</th>
        </tr>
    </thead>
    <tbody>
        <tr class="qtr" id="q1">
        <th scope="row">Q1</th>
        <td class="sent bar"><p>$18,450.00</p></td>
        <td class="paid bar"><p>$16,500.00</p></td>
        >
        <tr class="qtr" id="q2">
        <th scope="row">Q2</th>
        <td class="sent bar"><p>$34,340.72</p></td>
        <td class="paid bar"><p>$32,340.72</p></td>
        >
        class="qtr" id="q3">
        <th scope="row">Q3</th>
        <td class="sent bar"><p>$43,145.52</p></td>
        <td class="paid bar"><p>$32,225.52</p></td>
        >
        class="qtr" id="q4">
        <th scope="row">Q4</th>
        <td class="sent bar"><p>$18,415.96</p></td>
        <td class="paid bar"><p>$32,425.00</p></td>
        >
        ody>
```

News

Collections

For You

Topics



d the live demo for the bar graph here.

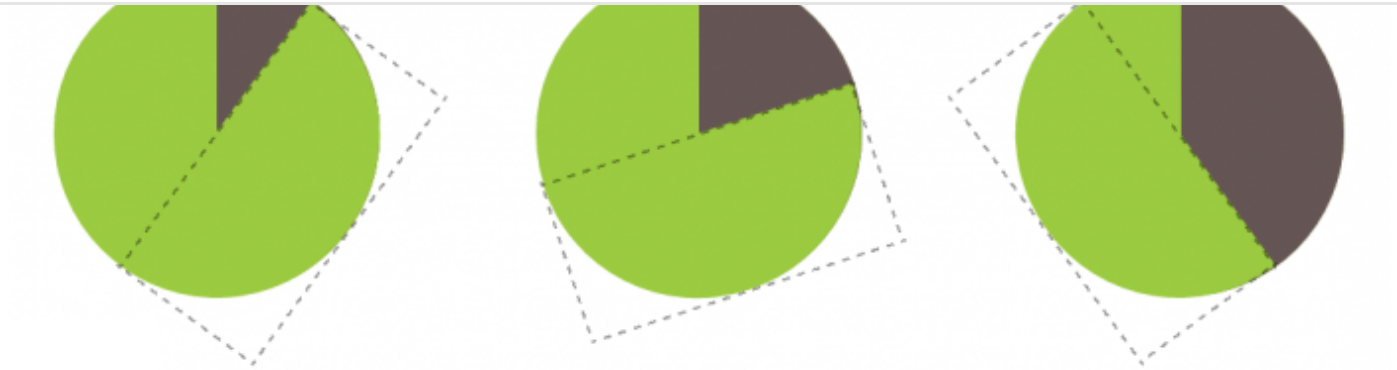
e, Robin also looks at how to create spark lines also using CSS, and also following the same ıd leveraging CSS's natural ability to create and style rectangular elements. Check his article ıils.

News

Collections

For You

Topics



ique enables you to create two-color pie charts as the one shown in the above example. dding more colours is not quite as simple.

re *conic gradients* would fill in. An example of a conical gradient is shown in the following owed from Lea's introductory article:

> able to control the percentage with a simple HTML attribute:
>
> ```
> background: conic-gradient(#655 attr(data-value %), yellow green 0);
> ```
>
> This also makes it incredibly easy to add a third color. For example, for a pie chart like the one shown on the pie chart above, we would just add two more color stops:
>
> ```
> background: conic-gradient(deeppink 20%, #fb3 0, #fb3 30%, yellowgreen 0);
> ```
>
> —Lea Verou

h conical gradients are not supported yet, Lea has created a polypill which allows you to them today, so definitely check it out if you're interested in experimenting with them today.

h pure CSS line graphs using image sprites are also possible, I highly recommend against se they're neither accessible nor are a good way to represent data.

News

Collections

For You

Topics

an HTML5 Canvas would require double the amount of maintenance.

With SVG, you get semantics and accessibility as well as interactivity with javaScript out of the box.

Bar charts can easily be created using simple SVG shapes such as `<rect>` s and `<line>` s. The numbers representing the data are also accessible since they are actual `<text>`, not an image of text.

> ou're likely to not want to create these charts manually, especially if they are complex charts
> data and that need to be both dynamic and interactive. This is where frameworks like d3.js
> ndy.

News

Collections

For You

Topics

(D3.js homepage)

D3 is possibly the most popular data visualisation library out there, and there have been lots of articles and books written on how to use it. Here are a couple of useful resources:

- D3 tutorials by Scott Murray

  ıctive Data Visualization for the Web book, also by Scott Murray

  mes to creating simple pie charts with SVG, you don't need to use a library, especially if your simple and static, not dynamic.

  shing Magazine article, Lea compares the CSS technique for creating pie charts with a clever que she also came up with.

  strokes and stroke dashes, you can fake the pie "slices" using heavy/thick strokes on an SVG

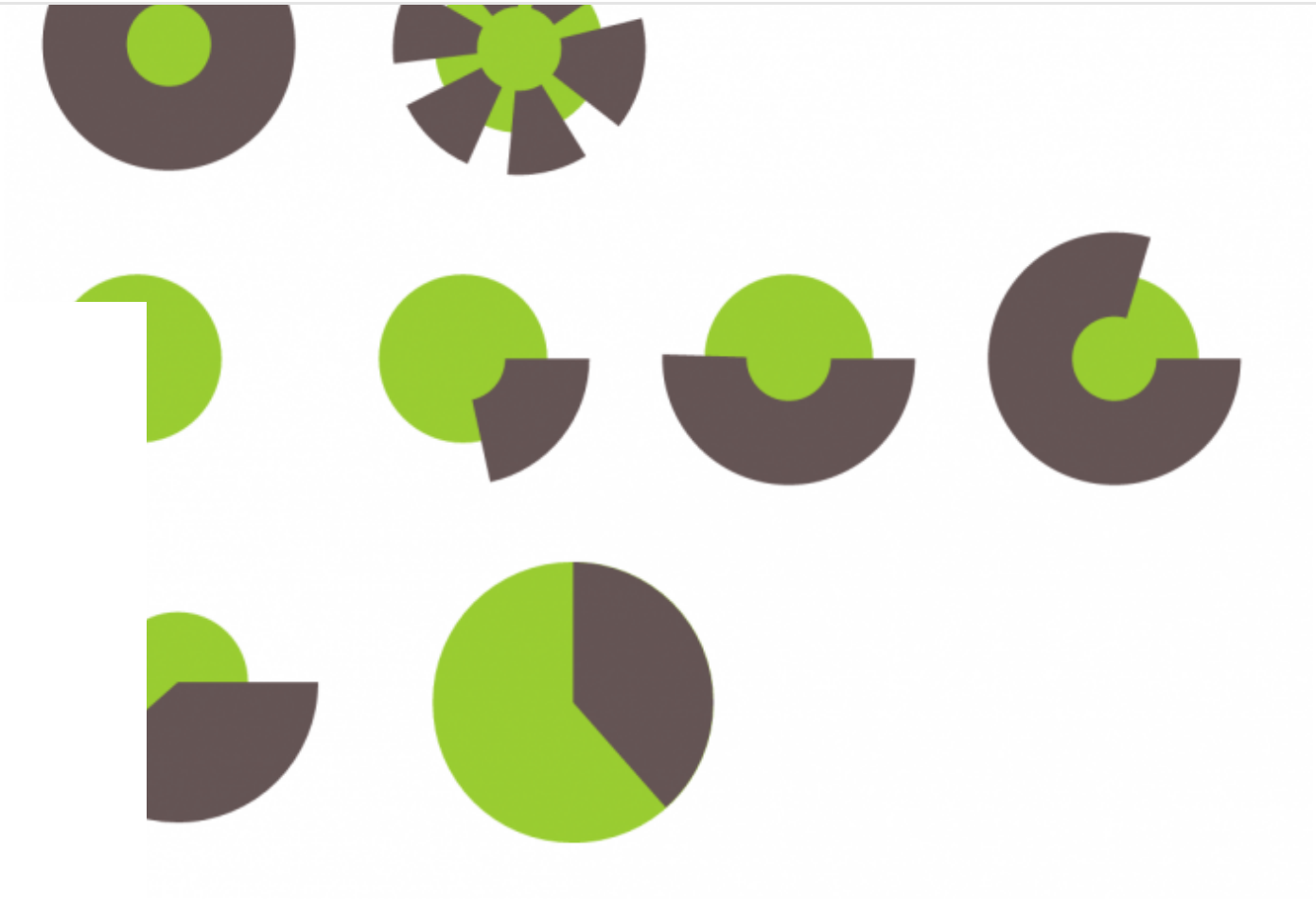
  ng images borrowed from Lea's article show the green circle and the "slices" which are hick stroke on the circle, divided into dashes using the `stroke` and `stroke-dasharray`

News

Collections

For You

Topics



JavaScript, you can create these strokes dynamically, and you can create multiple slices by ple SVG elements and strokes too. You can find the script and explanation in Lea's article.

News

Collections

For You

Topics

One area where CSS cannot even compare to SVG is creating irregular UI shapes and animating them. This is also possibly one of SVG's most powerful features.

An example of this is the following slideshow created by Manoela Ilic on Codrops:

The slideshow uses an SVG path and animates that path from one shape to the other and then to another upon clicking the slideshow's right and left buttons.

ng image borrowed from the Codrops tutorial shows the three path shapes that are used in on:

cript, you can define these path shapes:

```
 definitions
; : {
ect : 'M33,0h41c0,0,0,9.871,0,29.871C74,49.871,74,60,74,60H32.666h-
),0,0-10,0-30S6,0,6,0H33',
curve : {
    right :
lc0,0,5,9.871,5,29.871C79,49.871,74,60,74,60H32.666h-0.125H6c0,0,5-
5,0,6,0H33',
    left : 'M33,0h41c0,0-5,9.871-
C69,49.871,74,60,74,60H32.666h-0.125H6c0,0-5-10-5-30S6,0,6,0H33'
```

News

Collections

For You

Topics

other when the slideshow navigation buttons are clicked.

One might argue that the above effect might be possible using CSS. But, even though CSS clip paths come a little closer to creating irregular shapes in CSS (as we saw in the previous article), they still currently rely on SVG to create any complex shape that is beyond polygonal. So, the above path shape would not be possible to create in CSS without an SVG reference, for example, which means that SVG still gets the credit for such an effect.

here are other kinds of elastic shape animations that are simply not possible in CSS. The debar animation, also by Manoela, is an example:

's elastic effect requires you to draw a line and then animate the shape of that line on click can't draw a line in CSS and then animate it to another shape—at least not yet.

demo and similar elastic effects can be found on Codrops, and there are also some examples Codepen for you to play with.

Animated UI Effects and Images

SS has a set of filters that can be used to create certain effects, SVG's built-in filters are more d allow for much more complex effects that can be used to create very appealing UI effects.

er has been leading the way in showing what kind of amazing effects can be achieved with Motion blur is an example shown in this figure borrowed from a tutorial he wrote over at

News

Collections

For You

Topics

Wrapping Up

We've definitely not covered it all—I'm leaving some concepts to be discussed in how-to tutorials in the future, but I hope that this series gave you a clear idea of how SVG can be used to solve or improve solutions for some of our common design problems and/or requirements.

> true that sometimes, some design solutions can be improved using SVG rather than CSS.
> VG was never meant as a *replacement* for CSS. SVG and CSS are a powerful combination
> s us to take our static, rectangular UIs into a completely new level.
>
> enjoyed this series and found it useful.
>
> for reading.

• • •

Topics in this article

DESIGN +

Recommended Articles

Donna Morris, Executive Vice President, Customer and Employee Experience

We've Achieved Global Gender Pay Parity — a Milestone Worth Celebrating!

News

Collections

For You

Topics

Named one of the Best Global Brands of 2018

10-03-2018

9 2

Chris French

How a Scrappy Startup Mentality Helped This Big Tech Company Focus on its Customers

10-22-2018

8 6

News

Collections

For You

Topics