How to use z-index in svg elements?

I'm using the svg circles in my project like this,

```
<svg xmlns="http://www.w3.org/2000/svg" viewBox="0 0 160 120">
    <g>
        <g id="one">
            <circle fill="green" cx="100" cy="105" r="20" />
        </g>
        <g id="two">
            <circle fill="orange" cx="100" cy="95" r="20" />
        </g>
    </g>
</svg>
```

and i'm using the z-index in the **'g'** tag to show the elements the first.In my project i need use only z-index value, but i can't use the z-index to my svg elements. i have googled a lot but i didn't find anything relatively. So please help me to use z-index in my svg.

Here is the DEMO.

javascript jquery svg z-index

asked Jul 22 '13 at 11:38
karthik
1,475 2 13 38

6 Answers

Specification

In the SVG specification version 1.1 the rendering order is based on the document order:

```
first element -> "painted" first
```

Reference to the specification: SVG 1.1 Rendering Order

> 3.3 Rendering Order
>
> Elements in an SVG document fragment have an implicit drawing order, with the first elements in the SVG document fragment getting "painted" first. Subsequent elements are painted on top of previously painted elements.

Solution (cleaner-faster)

You should put the green circle as the latest object to be drawn.

```
<svg xmlns="http://www.w3.org/2000/svg" viewBox="30 70 160 120">
   <!-- First draw the orange circle -->
   <circle fill="orange" cx="100" cy="95" r="20"/>

   <!-- Then draw the green circle over the current canvas -->
   <circle fill="green" cx="100" cy="105" r="20"/>
</svg>
```

Run code snippet Expand snippet

Here the fork of your jsFiddle.

Solution (alternative)

The tag `use` with the attribute `xlink:href` and as value the id of the element. Keep in mind that might not be the best solution even if the result seems fine. Having a bit of time, here the link of the specification SVG 1.1 "use" Element.

> Purpose:
>
> To avoid requiring authors to modify the referenced document to add an ID to the root element.

```
<svg xmlns="http://www.w3.org/2000/svg" viewBox="30 70 160 120">
    <!-- First draw the green circle -->
    <circle id="one" fill="green" cx="100" cy="105" r="20" />

    <!-- Then draw the orange circle over the current canvas -->
    <circle id="two" fill="orange" cx="100" cy="95" r="20" />

    <!-- Finally draw again the green circle over the current canvas -->
    <use xlink:href="#one"/>
</svg>
```

Run code snippet Expand snippet

Notes on SVG 2

SVG 2 is the next version of the standard https://www.w3.org/TR/SVG2/.

> 3.4 Rendering Order
>
> Elements in SVG are positioned in three dimensions. In addition to their position on the x and y axis of the SVG viewport, SVG elements are also positioned on the z axis. The position on the z-axis defines the order that they are painted.
>
> ...
>
> 3.4.1. Controlling element rendering order: the ‘z-index’ property
>
> ...

- SVG 2 Support Mozilla - Painting
- How do I know if my browser supports svg 2.0
- Can I use SVG

edited yesterday

answered Aug 14 '14 at 7:23
Maicolpt
521 4 5

There is also an old draft about overriding the rendering order, but it is a feature not available. Draft Reference – Maicolpt Aug 14 '14 at 7:24

1 Great Answer, and reference to the spec! :) – Jonathan Marzullo Sep 24 '15 at 0:09

1 yikes! it isn't always easy to draw the elements in the order you want them painted, especially if the objects are being generated programatically and may appear nested (e.g. it appears **g** can't contain a,b, such that a is below g sibling c but b is above it) – Michael Sep 29 '15 at 1:24

@Michael: In your scenario, first I will try to understand if really the elements have to be grouped. – Maicolpt Oct 2 '15 at 23:27

Try to invert `#one` and `#two` . Have a look to this fiddle : http://jsfiddle.net/hu2pk/3/

`Update`

In SVG, **z-index is defined by the order the element appears in the document**. You can have a look to this page too if you want : http://stackoverflow.com/a/482147/1932751

edited Jul 22 '13 at 11:57

answered Jul 22 '13 at 11:44
Lucas Willems
3,822 1 12 31

1 Thanks but i need to the element based the z-index value. – karthik Jul 22 '13 at 11:53

Ok. And you want #one is on #two or the opposite ? – Lucas Willems Jul 22 '13 at 11:55

ya if i said z-index value as -1 for #one means it will show at top level. – karthik Jul 22 '13 at 11:57

8 There is no z-index property in any of the SVG specifications. The only way to define which elements appear on top and which appear to the bottom is by using DOM ordering – Nicholas Kyriakides Jul 23 '13 at 13:44

7 `d3.selection.prototype.moveToFront = function() { return this.each(function() { this.parentNode.appendChild(this); }); };` And then you can say `selection.moveToFront()` via stackoverflow.com/questions/14167863/… – mb21 Sep 24 '14 at 13:49

You can use use.

```
<svg xmlns="http://www.w3.org/2000/svg" viewBox="0 0 160 120">
    <g>
        <g id="one">
            <circle fill="green" cx="100" cy="105" r="20" />
        </g>
        <g id="two">
            <circle fill="orange" cx="100" cy="95" r="20" />
        </g>
    </g>
    <use xlink:href="#one" />
</svg>
```

The green circle appears on top.
jsFiddle

answered Oct 9 '14 at 11:36
Jose Rui Santos
9,264 3 28 44

2 Does this draw #one twice? – mareoraft Jul 24 '15 at 3:49

@mareoraft Yes, `#one` is drawn twice. But if you want, you can hide the first instance through CSS. `use` has the same effect as cloning the referred DOM element – Jose Rui Santos Jul 24 '15 at 6:03

As others here have said, z-index is defined by the order the element appears in the DOM. If manually reordering your html isn't an option or would be difficult, you can use D3 to reorder SVG groups/objects.

Use D3 to Update DOM Order and Mimic Z-Index Functionality

Updating SVG Element Z-Index With D3

At the most basic level (and if you aren't using IDs for anything else), you can use element IDs as a stand-in for z-index and reorder with those. Beyond that you can pretty much let your imagination run wild.

Examples in code snippet

```
var circles = d3.selectAll('circle')
var label = d3.select('svg').append('text')
                                                .attr('transform', 'translate(' +
[5,100] + ')')
```

```
var zOrders = {
        IDs: circles[0].map(function(cv){ return cv.id; }),
    xPos: circles[0].map(function(cv){ return cv.cx.baseVal.value; }),
    yPos: circles[0].map(function(cv){ return cv.cy.baseVal.value; }),
    radii: circles[0].map(function(cv){ return cv.r.baseVal.value; }),
    customOrder: [3, 4, 1, 2, 5]
}

var setOrderBy = 'IDs';
var setOrder = d3.descending;

label.text(setOrderBy);
circles.data(zOrders[setOrderBy])
circles.sort(setOrder);
```

```
<script src="https://cdnjs.cloudflare.com/ajax/libs/d3/3.4.11/d3.min.js"></script>

<svg xmlns="http://www.w3.org/2000/svg" viewBox="0 0 400 100">
  <circle id="1" fill="green" cx="50" cy="40" r="20"/>
  <circle id="2" fill="orange" cx="60" cy="50" r="18"/>
  <circle id="3" fill="red" cx="40" cy="55" r="10"/>
  <circle id="4" fill="blue" cx="70" cy="20" r="30"/>
  <circle id="5" fill="pink" cx="35" cy="20" r="15"/>
</svg>
```

Run code snippet   Expand snippet

The basic idea is:

1. Use D3 to select the SVG DOM elements.

   ```
   var circles = d3.selectAll('circle')
   ```

2. Create some array of z-indices with a 1:1 relationship with your SVG elements (that you want to reorder). Z-index arrays used in the examples below are IDs, x & y position, radii, etc....

   ```
   var zOrders = {
       IDs: circles[0].map(function(cv){ return cv.id; }),
       xPos: circles[0].map(function(cv){ return cv.cx.baseVal.value; }),
       yPos: circles[0].map(function(cv){ return cv.cy.baseVal.value; }),
       radii: circles[0].map(function(cv){ return cv.r.baseVal.value; }),
       customOrder: [3, 4, 1, 2, 5]
   }
   ```

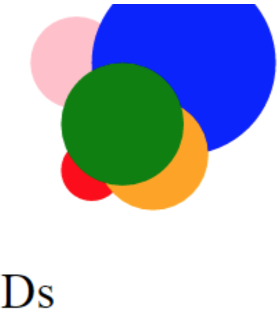
3. Then, use D3 to bind your z-indices to that selection.

   ```
   circles.data(zOrders[setOrderBy]);
   ```

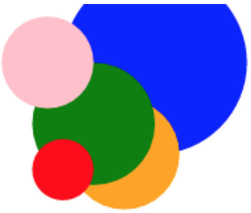
4. Lastly, call D3.sort to reorder the elements in the DOM based on the data.

   ```
   circles.sort(setOrder);
   ```

Examples

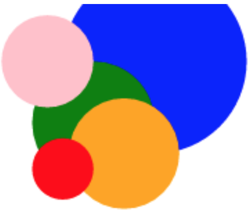


- You can stack by ID



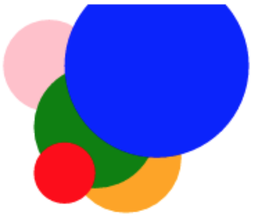
xPos

- With leftmost SVG on top



radii

- Smallest radii on top



customOrder

- Or Specify an array to apply z-index for a specific ordering -- in my example code the array `[3,4,1,2,5]` moves/reorders the 3rd circle (in the original HTML order) to be 1st in the DOM, 4th to be 2nd, 1st to be 3rd, and so on...

answered Apr 20 at 7:01
Steve Ladavich
826 2 17

Another solution would be to use divs, which do use zIndex to contain the SVG elements.As here: https://stackoverflow.com/a/28904640/4552494

answered Mar 6 '15 at 18:15
Roman Rekhler
199 2 3

Using D3:

If you want to add the element in the reverse order to the data use:

```
.insert('g', ":first-child")
```

Instead of .append

Adding an element to top of a group element answered Apr 22 at 11:29

Nathan Hensher
33 5