position:sticky is back in Chrome

By Paul Kinlan

Paul is a Developer Advocate

Four years ago Eric Bidelman created a rather awesome blog post about the fact that `position: sticky` landed in WebKit, which at the time was the engine that powered Chrome (as well as many other browsers including Safari). One year later, and much to the consternation of web developers we removed `position:sticky` from Chrome because "the current implementation isn't designed in a way that integrates well with the existing scrolling and compositing system".

We've always wanted to get it back in to Chrome as the bug stated, "Once we've got our scrolling and compositing house in order, we should return to `position: sticky` and implement the feature in a way that integrates well with the rest of the engine". The meta bug tracking the implementation has been worked on since 2013.

The great news is that as of Chrome 56 (currently beta as of December 2016, stable in Jan 2017) `position: sticky` is now back in Chrome.

What is position:sticky?

It's taken a little while to get here, so why am I excited about it?

`position:sticky` is a CSS positioning attribute that allows you to fix an element to the viewport (i.e, anchor it to the top of the screen) but only when its parent is visible in the viewport and it is within the threshold value. When it is not fixed to the viewport, the element will act like it is `position: relative`. It is a very nice and simple addition to the platform that removes the need to use JavaScript in an `onscroll` event handler just to lock an element to the top of the viewport.

This is what it looks like on my blog. It allows me to keep the current section's header at the top of the screen whilst you read my rather long and laborious articles :\

To implement this feature specify that the `position` attribute should have the value of `sticky` on the element that you want to be, er, stuck. Additionally, you can also add in the offset at where it needs to be stuck.

```
h3 {
  /* Element will be 'fixed' when it ... */
  position: sticky;
  /* ... is 10px from the top of the viewport */
  top: 10px;
}
```

The previous example will fix the `<h3>` element at 10px from the top of the viewport. To fix it directly to the top of the viewport you would set the `top` attribute as `top: 0px`.

Support for this feature is pretty strong. It is available on Chrome (yay), Firefox and Safari. Here are more details about `position:sticky`:

- Specification
- MDN

Last updated July 2, 2018.