



Edit Manage Stats



Ingo Steinke

Posted on Aug 14



5



2



3



3

CSS max-width after transform + scale vs. pseudo elements

#webdev #css

Another one of those situations where order of declarations matters (it often does in CSS anyway) and the don't even seem to be handled consistently across browsers.

Ensure max-width after scale transformation

A simple use case: restrict the maximum width of an element after transformation. Maybe this problem only applies to pseudo elements, but we'll see.

The actual case adds another aspect that might cause the problem: a translation on the Z-axis which will result in an additional perceived scale transformation when applied to a 2-dimensional browser window.

First, I apply a z-translation in order to set up a [parallax perspective effect](#), then I apply a 2d scale to compensate for the perceived downsizing.

This worked fine so far (at least in Chrome / Chromium and Firefox, and only in my specific project setup, see below ...)

```
.container::after {  
  content: ""; /* add more visual styles here */  
  transform-origin: top left;  
  transform: translateZ(-1px) scale(2);  
}
```

Now I want to add an additional restriction to a maximum width on wide screens, so I add

```
width: auto;  
max-width: 100%;
```

which also worked as intended on Chrome and Firefox. It did not even matter if I put the width before or after the translation, it still got applied to the resulting width - but not in Safari!

Safari consistently used another calculation order, no matter where I put the `max-width` in my code, it will never apply to the element's width after transformation.

Avoid this combination if you can

This important difference is either documented poorly or just hard to find. It might be another rare edge case that you should avoid if you can, or maybe "max-width", "scale", "transform", "order", and "Safari" are too broad and general terms to yield any helpful result apart from the usual go-to resources covering best practices for beginners.

Don't use `transform:translateZ` together with `scale` and `max-width` if you can avoid it.

But if you do ...

I got so frustrated about wasting my time on cross-browser issues again, that I was tempted to use a [CSS browser hack](#) to work around the issue. But this always feels like making it work without knowing why, beyond the perceived browser behavior.

What I actually want to do is state explicitly that the `max-width` must be applied to the pseudo element after the transition, not before.

Now it seems I finally rephrased my problem in a way where I start getting helpful results after all.

Search region: Germany ▾ Any time ▾

[https://stackoverflow.com](https://stackoverflow.com/questions/66128302) › [questions](#) › 66128302

Can you define a maximum width/height to CSS transform scale?

Depending on what your actual situation is, you can use `width: 50%; max-width: 50px; margin: 0 auto;` to achieve better results and lose the transform and transform-origin: `.outer {width: 200px; height: 100px; background-color: yellow; position: relative} .inner {width: 50%; max-`

[https://developer.mozilla.org](https://developer.mozilla.org/en-US/docs/Web/CSS/max-width) › [en-US](#) › [docs](#) › [Web](#) › [CSS](#) › [max-width](#)

max-width - CSS: Cascading Style Sheets | MDN - MDN Web Docs

Syntax `css max-width: 3.5em; max-width: 75%; max-width: none; max-width: max-content; max-width: min-content; max-width: fit-content(20em); /* Global values */ max-width: inherit; max-width: initial; max-width: revert; max-width: revert-layer; max-width: unset; Values...`

[https://stackoverflow.com](https://stackoverflow.com/questions/15507941) › [questions](#) › 15507941

Position AFTER transform in CSS? - Stack Overflow

CSS: `#outer { border: solid 1px red; width:600px; height: 600px; position: relative; } #text { transform: rotate (90deg); position: absolute; top: 0; right: 0; }` In Firefox 19.0.2 on OS X 10.6.8, it fails. This appears to be because, despite the order in which the CSS properties were given...

People also asked:

- [Can you define a maximum width/height to CSS transform scale?](#)
- [Position AFTER transform in CSS?](#)
- [How to limit a transform scale to the width and height of the parent?](#)

I did not even read the other two, because as I prepared a codepen to reproduce the problem in an isolated way, I found it hard to make it behave as I described.

In short, transformation combined with `max-width` does not work well, sometimes not at all, on a pseudo element (like `.container::after` but it does work well on a regular DOM child.

So if you do ...

Avoid using the problematic combination on pseudo elements and make sure to use regular child nodes instead.

Maximum width will then be applied to the calculated width after the transformation has been applied.

Here is the [codepen: Transform + max-width on child nodes vs. pseudo elements](#)

HTML

CSS

Result

EDIT ON

Here is a parent container (blue).

Here is an actual child element in the DOM (green, because it works as intended)

after the container content (red, please avoid if you can)

◀

▶

Resources

1× 0.5× 0.25×

Rerun