CONTAINER QUERIES

- What is it?
- Can I eat it?
- What took them so long?

Hello

- I’m Mathias Ober
- Frontend Developer > 10 years
- Working at valantic
- X / Twitter: **@_omat** 👨‍💻 or **@SkyKartoffel** 🎮
- Hobbies: 🎮 ⛵ 🏸 ⛷️ 🎤 🎸

Somewhen around 2010 AD:

Web devs: **We want to do responsive stuff dependant on screen size!**

Browser devs: ***Sure, here we got you media queries.***

Web devs: **Noice, cool, thx bro.**

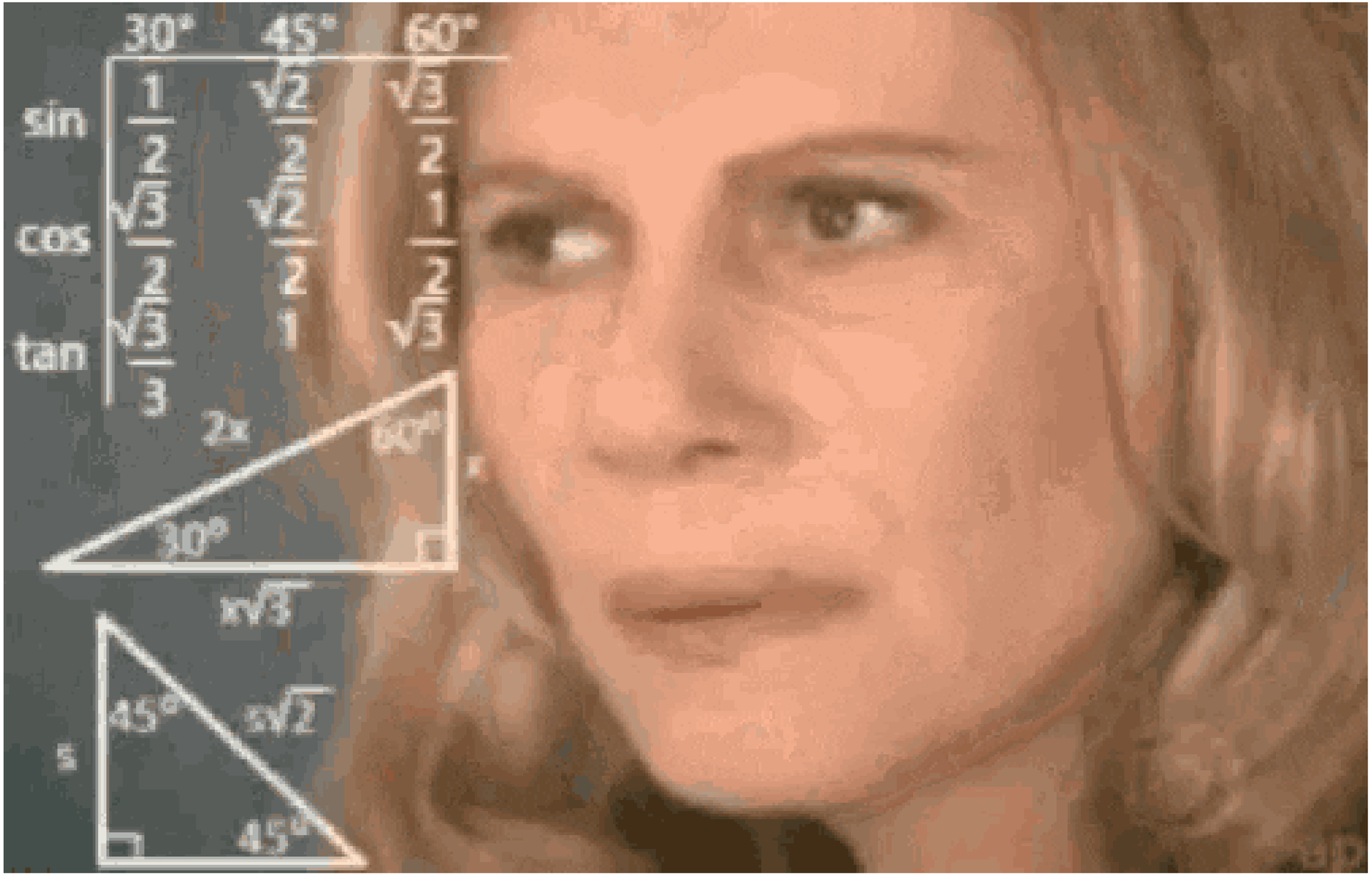
Coding Time 01

Designer enters the game

We need a sidebar. Just use the existing mobile styling.

It’s easy, right?

Coding Time 02


30°
45°
60°
sin
1/2
√2/2
√3/2
cos
√3/2
√2/2
1/2
tan
√3/3
1
√3
2x
60°
30°
x√3
45°
x√2
x
45°

Web devs: **Quick question, how can make stuff responsive to their container?**

Browser devs: ***… about that… Nope, not gonna happen. Simply not possible.***

Web devs:

But why?

- Basic principles of how browser layout works
- Every box has an intrinsic size
  - The content defines the size of the box
- Some boxes have an extrinsic size
  - The context defines the size of the box
  - User sets width for component

Why not?

```css
.our-element:media(min-width: 500px) {
  width: 499px;
}
```

> *Well, since the query no longer matches, the new width is no longer applied. Since that new width is never applied, the element query would match again, so the new width would be applied, so the query would no longer match, so the new width wouldn't be applied—and so on unto infinity. We've achieved a TARDIS-caliber paradox with just a few lines of CSS, and there's no predictable way for the browser to handle it.*

https://css-tricks.com/container-queries-once-more-unto-the-breach/

... years went by and developers dealt with those limitations, everyone was living his life and people forgot what they really wanted.

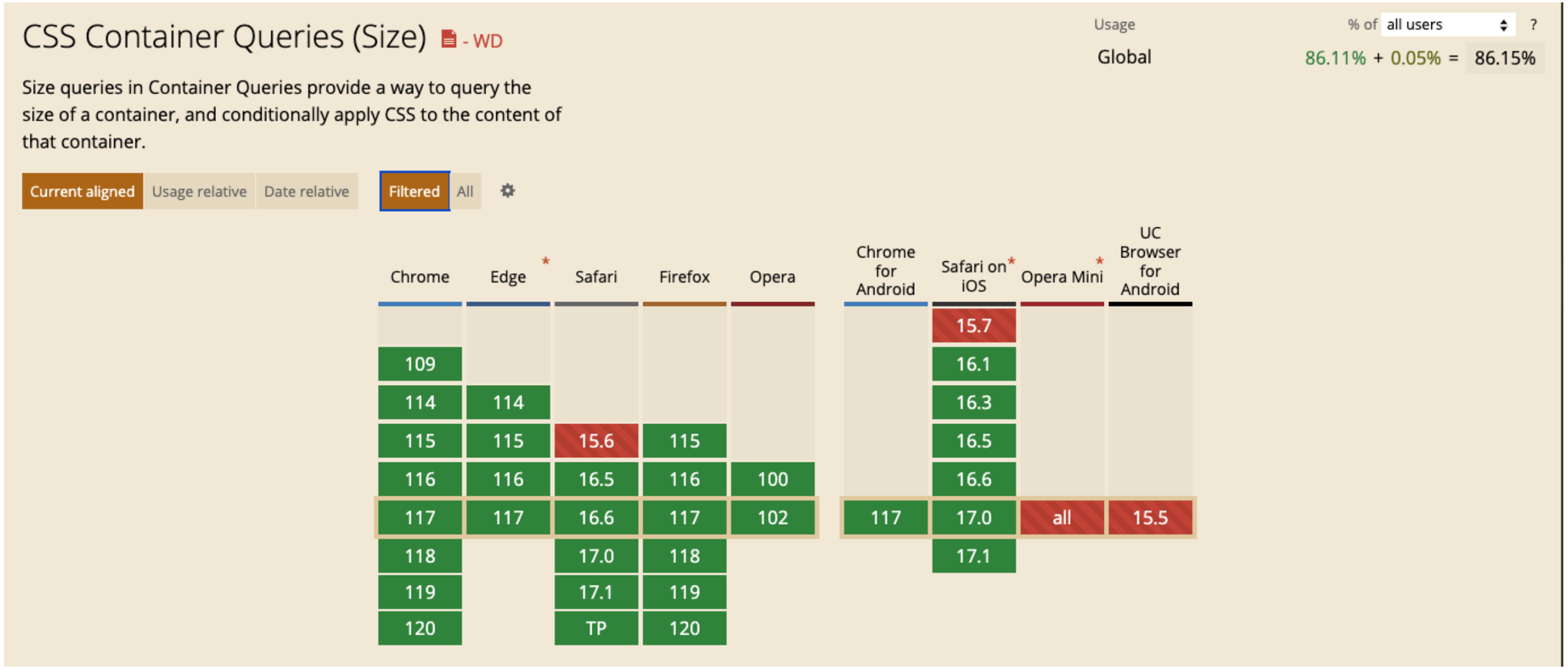
The END

... until 2022 ...

Browser devs: ***Remember about that thing you asked some time ago?***

CONTAINER QUERIES

Coding Time 03



There is a polyfill for older versions of the supported browsers

Did they lie to us?

What happened?

> *... we need to reframe the way we talk about a potential solution. Since a solution can't allow an element to restyle itself, we can build that constraint into the specification: queries attached to an element can only influence the styling of that element's child elements.*

https://css-tricks.com/container-queries-once-more-unto-the-breach/

Limitations

- You need to define what you want to query from a container
- As a developer we need to create explicit containers to measure them
- CSS property `contain` tells browsers how to render specific elements
  - You can set layout, size, paint, content, or none
- `container-type`: `inline-size` or `size`

Things to keep in mind

- Some of the default behavior changes for containers
  - `position: fixed` reacts now to the defined container
  - Changes how content is handled
- Container can’t query themselves
- You can’t change what you query
- Grid: Don’t query the grid container → add wrapper for each element to measure

Container

- Finding container query the nearest ancestor
- container can have a name (or multiple)
- range syntax for container queries (and media queries now)

```
.sidebar {
  container-type: inline-size;
  container-name: sidebar;
}

@container sidebar (50px < width < 380px) {
  .card {
    background-color: red;
  }
}
```

Container Query Units

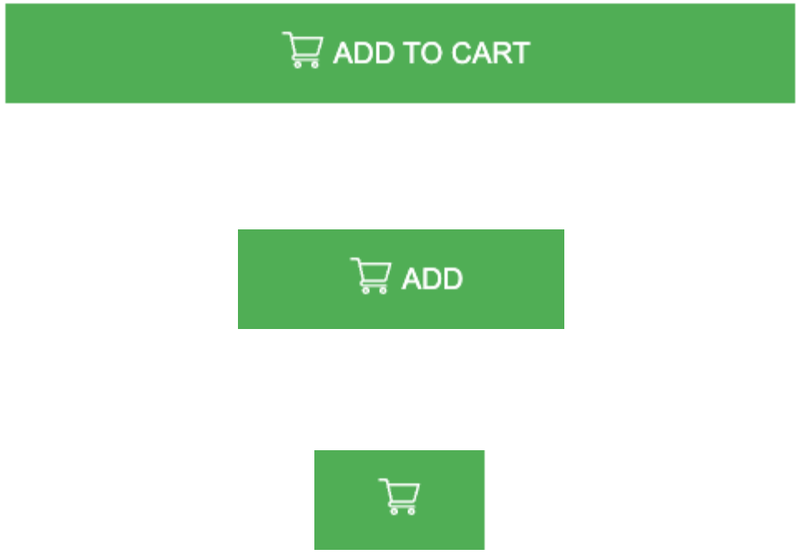
- cqw, cqh, cqi, cqb, cqmin, cqmax → Same as viewport units but for containers.
- Can be used for animations

Examples

```
@container (width > 220px) {
  button span.full-text {
    display: inline;
  }
}

@container (100px < width < 220px) {
  button span.small-text {
    display: inline;
  }

  button span.full-text {
    display: none;
  }
}
```



Coding Time: Button

Coding Time: Grid

What’s next

- Queries on style (custom properties) of the container (chromium only)
  - `@container style(--colors: invert)`
  - No containment required (Every element is a style query by default)
- Hopefully, but not confirmed:
  - Queries on normal style properties
  - Is a sticky element currently snapped?
  - and so on

Thanks to

- Miriam Suzanne for the inspiration
- Patric Eberle for the motivation and help with the talk
- valantic and vlbgWebDev for hosting of the event
- you for listening