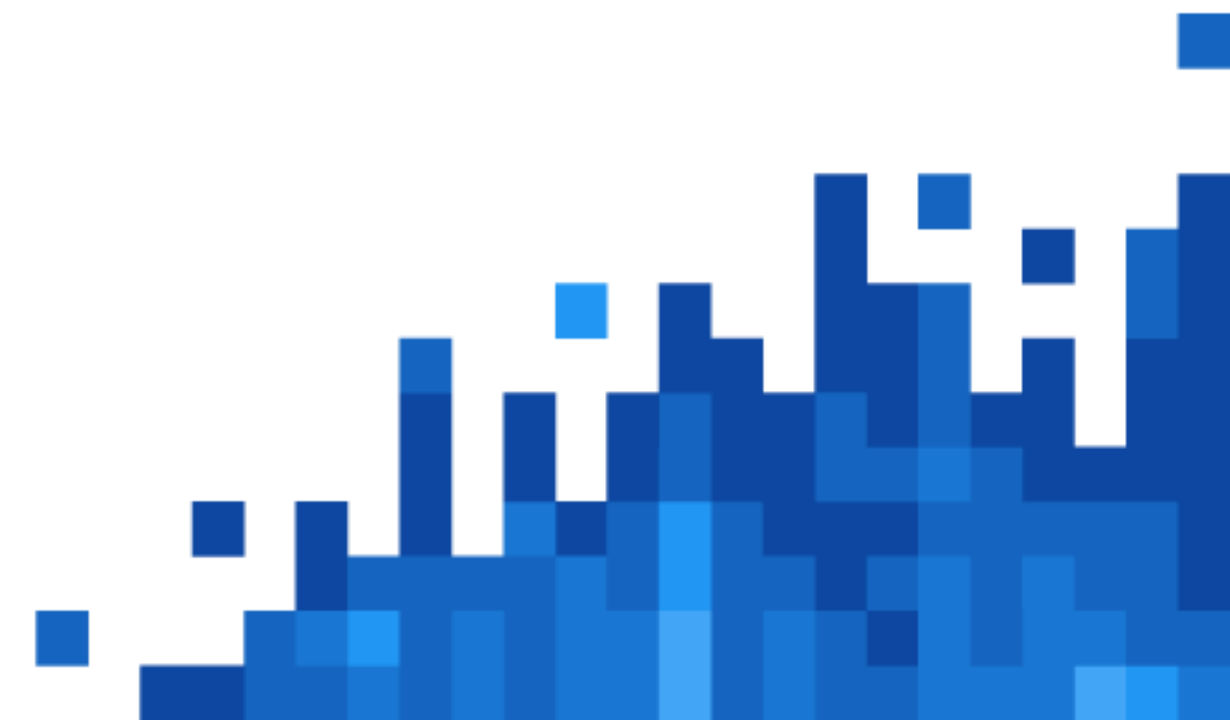# WebVR API

Brandon Jones
@tojiro
Google

Nell Waliczek
@NellWaliczek
Microsoft

10 second WebVR Primer

- JavaScript API
- Provides information about & access to VR devices (Capabilities, Pose, Input, etc.)
- Visuals produced with WebGL
- Not exactly what one would call "easy"

Why is it valuable

- Lots of need for simpler media (360 video, etc)
- App model discourages short form content
- Some content is not store-friendly
- Security model of the web is a big plus
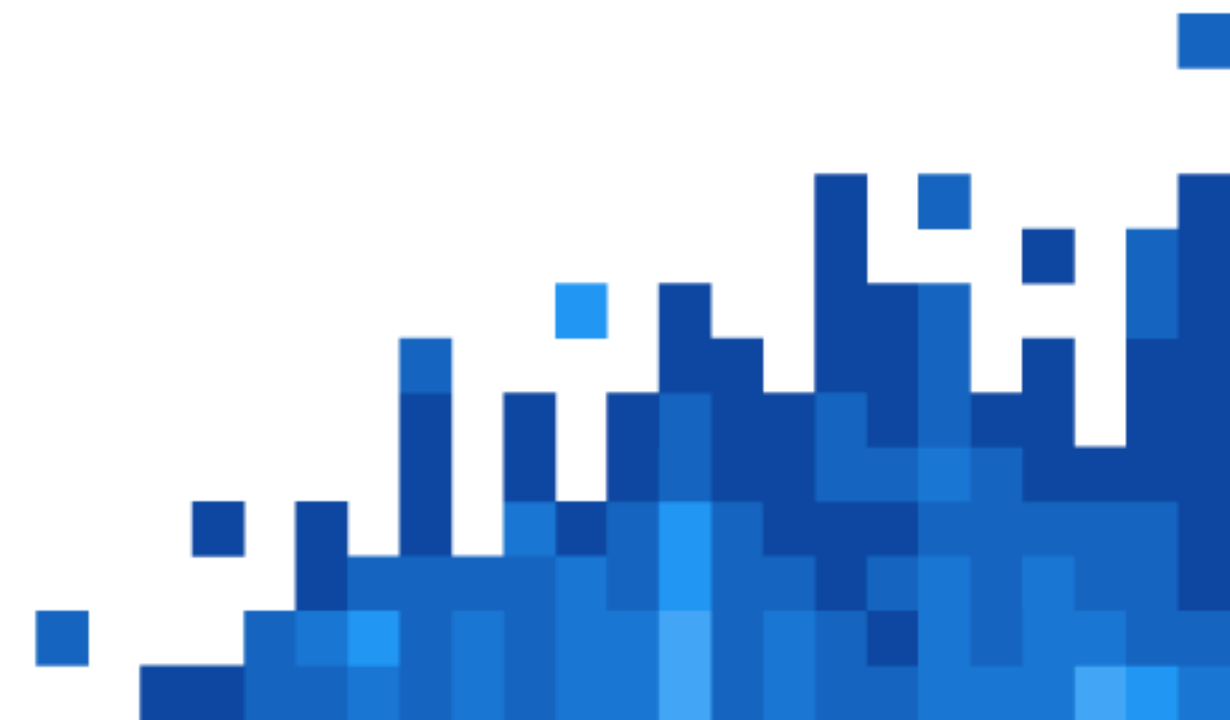- Web welcomes experimentation

Video Link!

How do I embed YouTube in Keynote?!?

WebVR is available today! (sort of)

- WebVR "1.1" API
- Accessible as an Origin Trial in Chrome
- Enabled by default in Edge, Firefox,
  and some Mobile VR browsers
- Support for every major headset except PSVR

Not without challenges

- VR is a new tech with evolving best practices
- More demanding than other, similar tech
- Kind of insane for JavaScript

What's next?

Great developer feedback

- Weird edge cases
- Unexpected API "abuses"
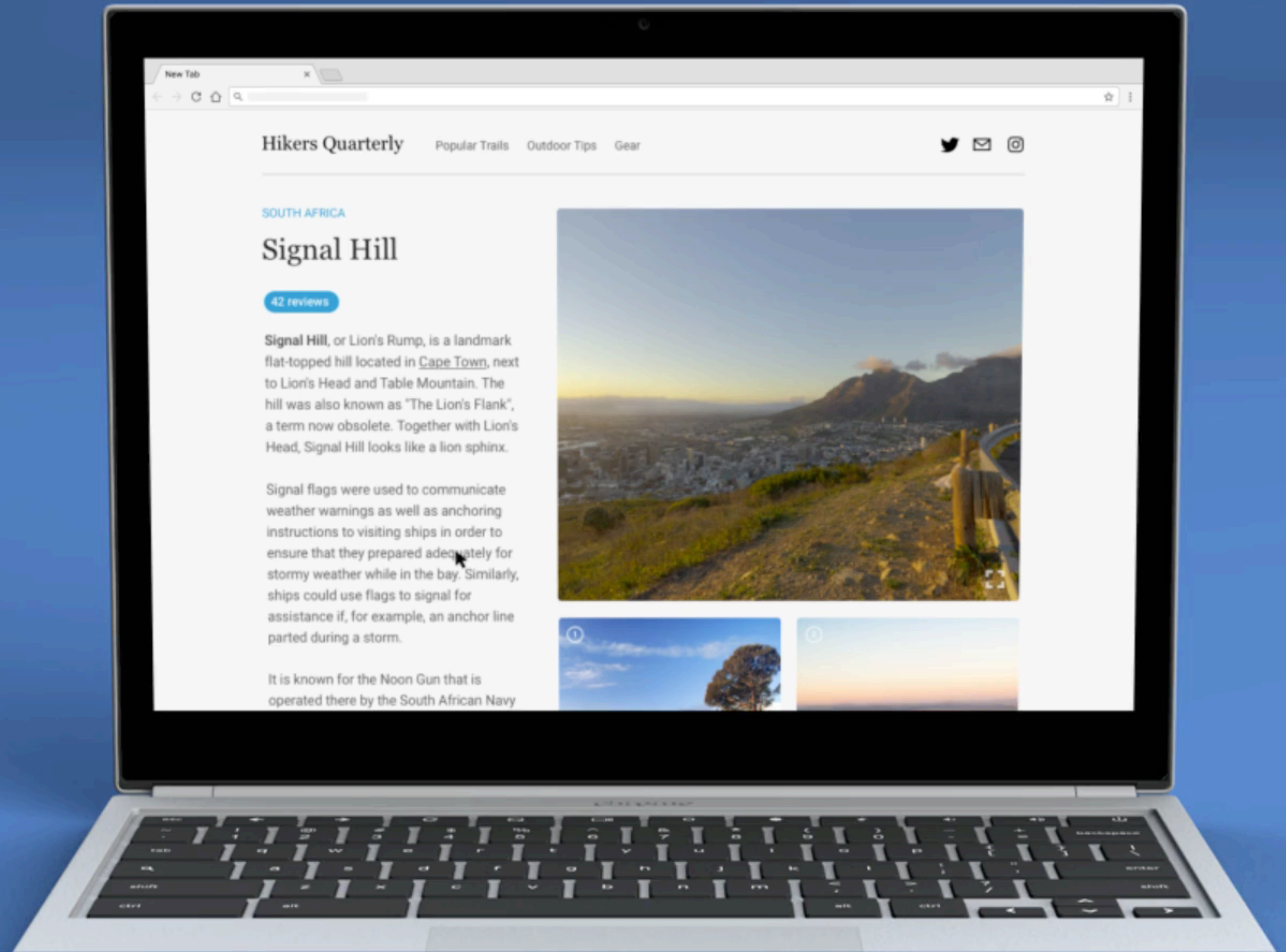- Hard-to-hit best practices

WebVR "2.0"

- Enshrines known best practices
- Explainer available today. Spec incoming.
- All browser vendors committed to the transition
- Designed as a foundation for VR and AR needs

WebVR 2.0
Magic window

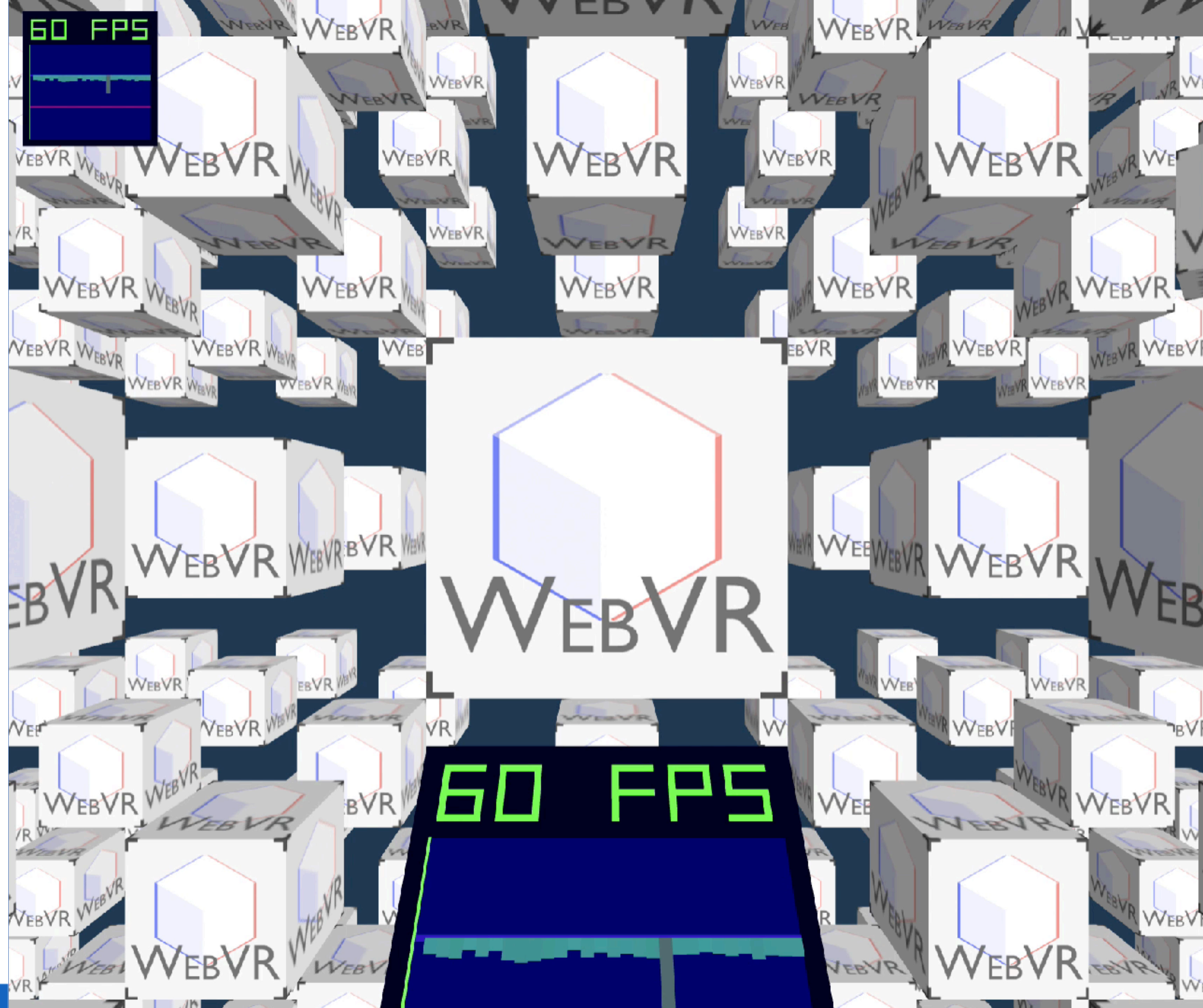Now a first-class part of the API. Uses the same render/input loop as headset presentation.

WebVR 2.0

Prescriptive rendering

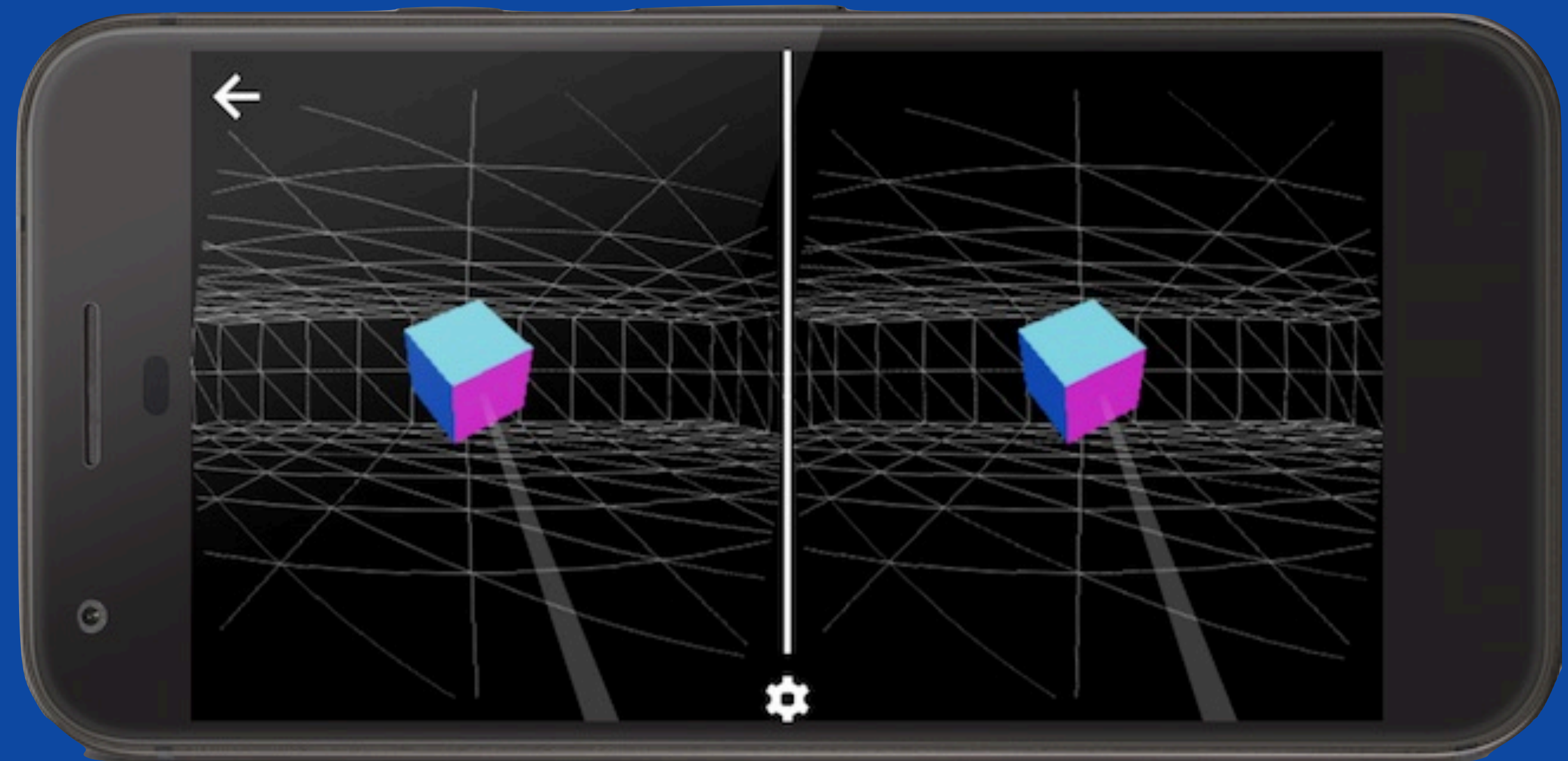Tells content what you need to draw in order to look correct. It provides the render loop with:

- Framebuffers
- Views to render
- Projection/View matrices
- Viewports

WebVR 2.0
Input

Multiple developers
independently
created "ray based"
input systems. Making
that style of input a
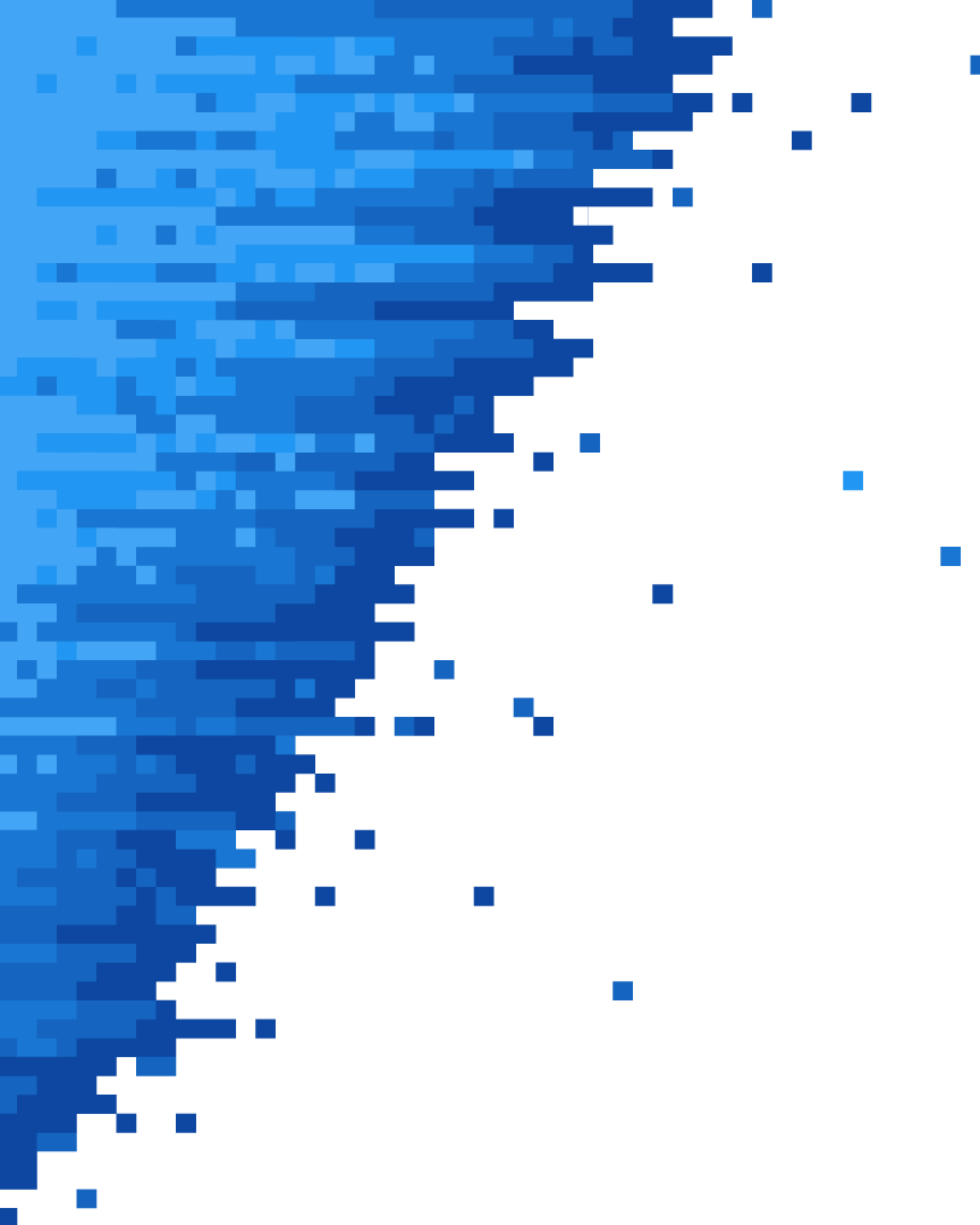core part of the API.

WebVR Ecosystem

Lots of great libraries available for all skill levels

- WebVR Polyfill - Basic support on all platforms
- Three.js, Babylon.js - Easier Javascript
- A-Frame - Declarative
- Play Canvas, Vizor - Visual authoring tools

Thanks!