CSSWG Gamut Mapping Breakout

Christopher Cameron
2024-03-27

High level positions

Providing a (hue, chroma, luma) parameter space for specifying and manipulating colors is an extremely useful feature that we should deliver.

Color matching between CSS, images, video, and canvas is a critical feature.

High level positions

Providing a (hue, chroma, luma) parameter space for specifying and manipulating colors is an extremely useful feature that we should deliver.

Color matching between CSS, images, video, and canvas is a critical feature.

Use of imaginary and extremely-out-of-gamut colors is a dangerous and unstable foundation important functionality, and should be discouraged.

Current spec

Attempts to deliver safe (hue, chroma, luma) space via oklch space

- This is not what oklch is designed for
- This results in imaginary and extremely-out-of-gamut colors
- This is “fixed up” by imposing a meaning on oklch via CSS gamut mapping

Current spec

Attempts to deliver safe (hue, chroma, luma) space via oklch space

- This is not what oklch is designed for
- This results in imaginary and extremely-out-of-gamut colors
- This is “fixed up” by imposing a meaning on oklch via CSS gamut mapping

CSS gamut mapping has side-effects

- It extremely violates the meaning of color values
- It breaks color matching
- It encourages “future-dangerous” content
- It is only useful for imaginary or extremely-out-of-gamut content

This is not the way to deliver the desired functionality

It extremely violates the meaning of color values

Violating the meaning of color values

CSS gamut mapping forces oklch L=1 plane to white and L=0 plane to black.

This is simply not what those planes mean.

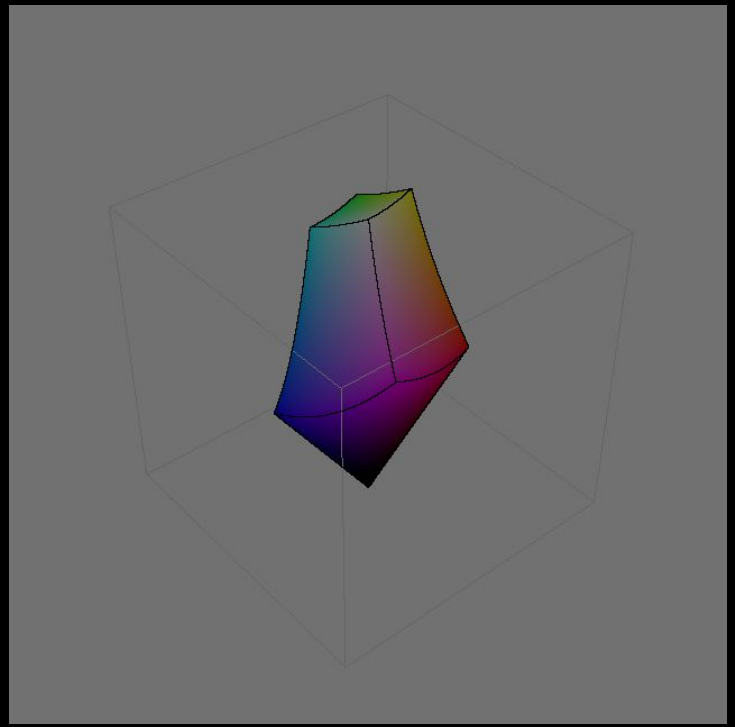
L=0 is entirely imaginary (physically non-existent colors).

L=1 is very colorful.

Violating the meaning of color values

P3 gamut in oklab/lch

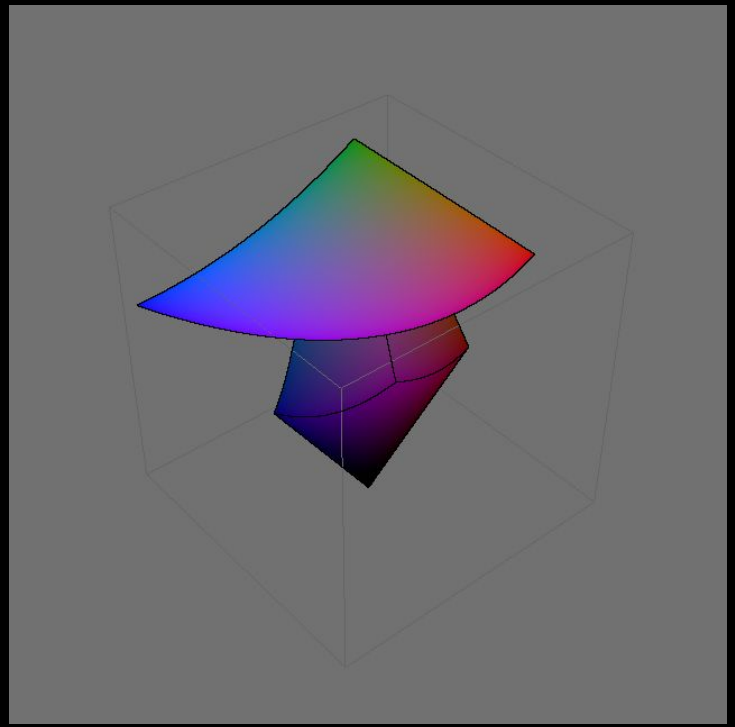
Darkened so that background is #FFFFFF



Violating the meaning of color values

L=1 plane in oklab/oklch

Extremely bright and colorful!



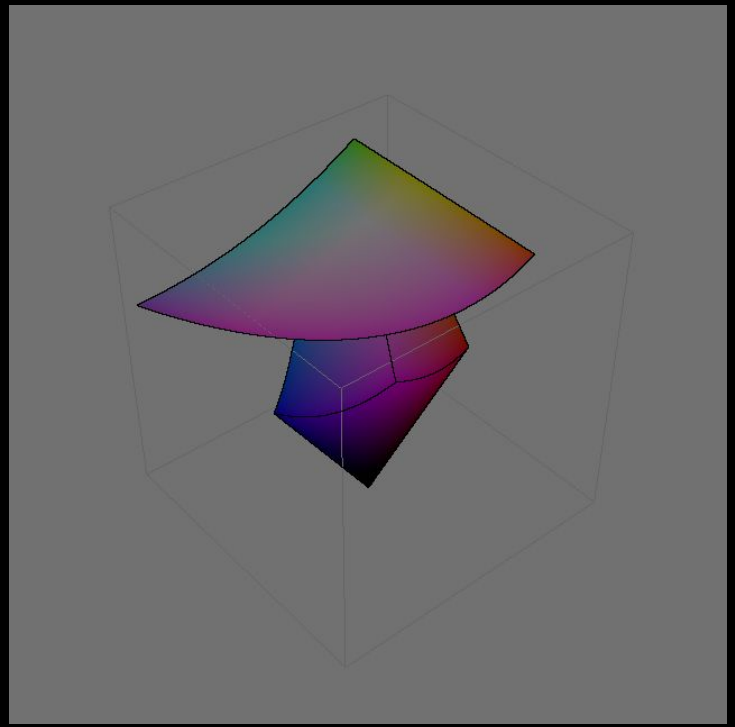
Violating the meaning of color values

L=1 plane in oklab/oklch

With per-channel clamping

- This is “hue shift”
- Not a good representation of original

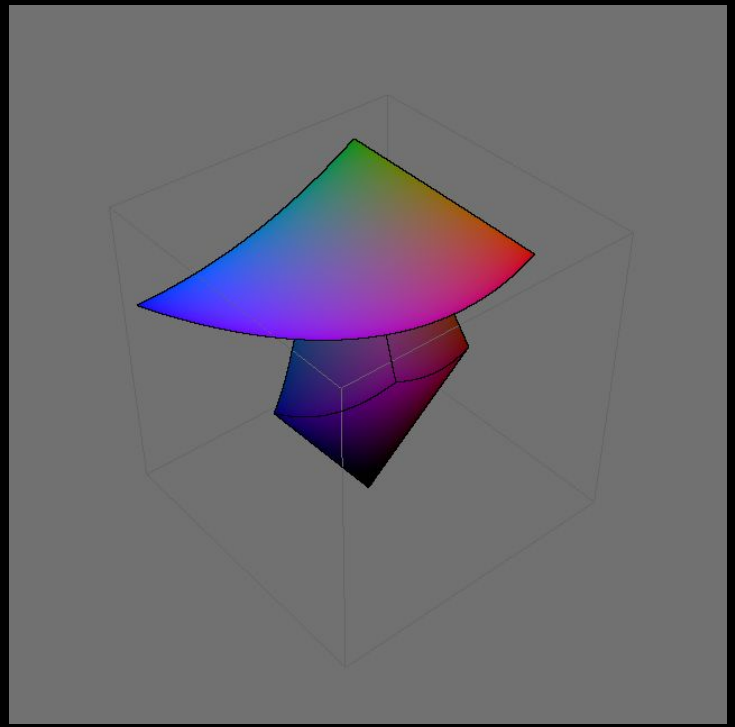
This hue shift only happens when very far out-of-gamut



Violating the meaning of color values

L=1 plane in oklab/oklch

Extremely bright and colorful!

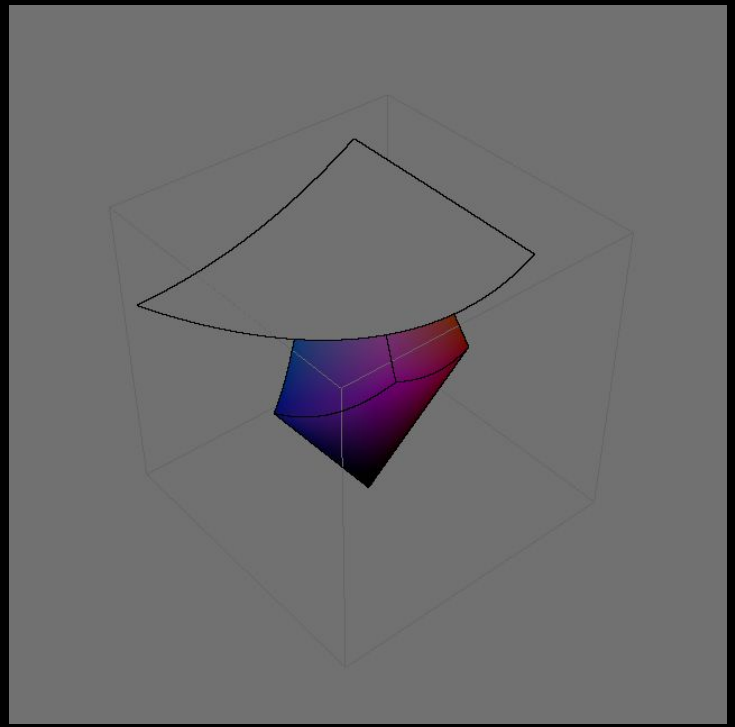


Violating the meaning of color values

L=1 plane in oklab/oklch

With CSS gamut mapping

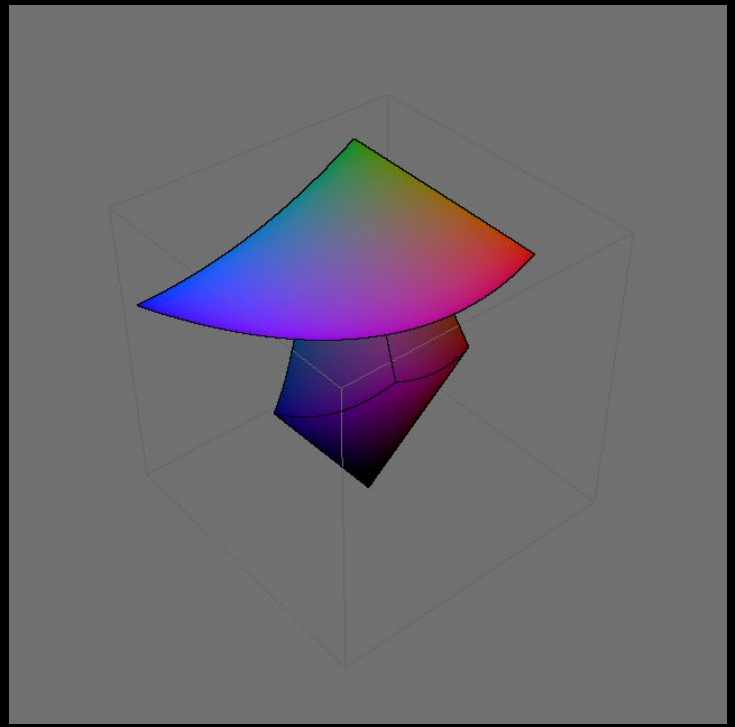
- No hue shift …
- Not a good representation of the original



Violating the meaning of color values

L=1 plane in oklab/oklch

Extremely bright and colorful!

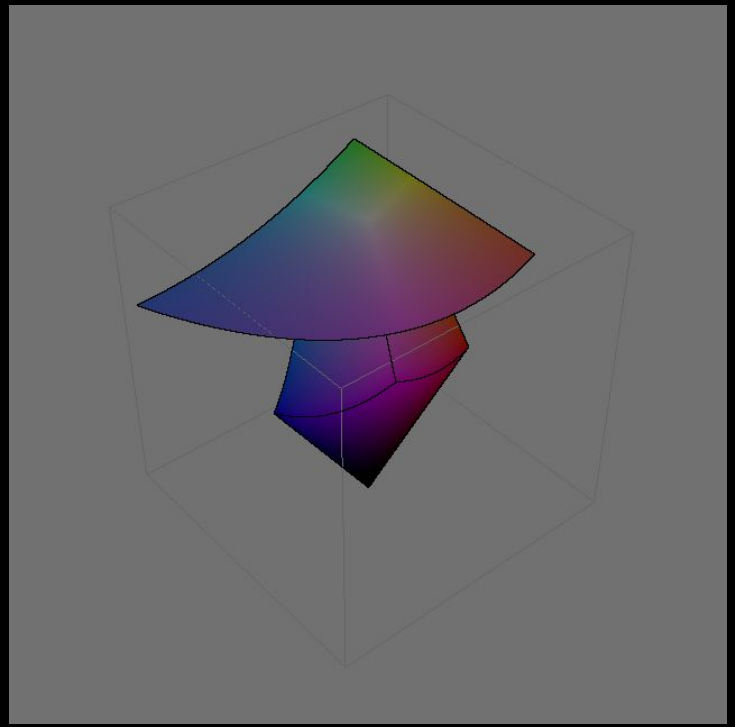


Violating the meaning of color values

L=1 plane in oklab/oklch

Something more reasonable (ad-hoc)

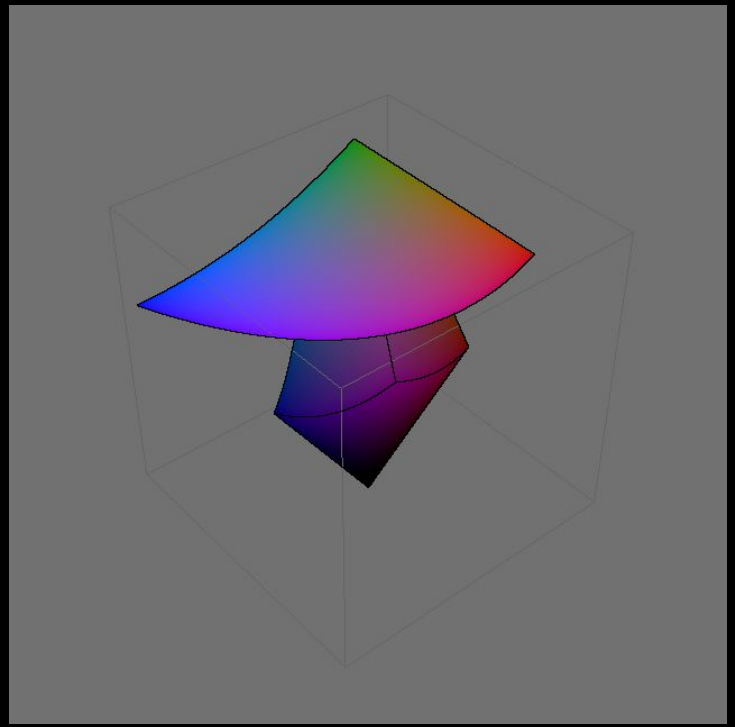
This is what we should aim for!



Violating the meaning of color values

L=1 plane in oklab/oklch

Extremely bright and colorful!



It breaks color matching

Breaking color matching

Gamut mapping is applied only to CSS colors

Not to image, video, or canvas.

It is impossible to ensure a CSS color and an image will look the same.

Breaking color matching

It is impossible to ensure a CSS color and an image will look the same.

- color(display-p3 0 1 0) becomes
- rgb(0% 98.5% 16.0%) in CSS and
- rgb(0% 100% 0%) in images and canvases.

Breaking color matching

Every partner requesting CSS Color Level 4 from us wanted this.

It features prominently in WebKit's "Improving Color on the Web" announcement:

> "WebKit is very excited to provide improved color features to developers through color matching and color gamut detection"
>
> "If you serve wide-gamut images to users not on a wide-gamut display, WebKit will color-match the images and show them in the sRGB space. However, this conversion into sRGB can be done a few ways and isn't guaranteed to happen identically on other browsers or platforms."

Breaking color matching

Color matching being broken is a red flag that the underlying approach is inconsistent and prone to more problems.

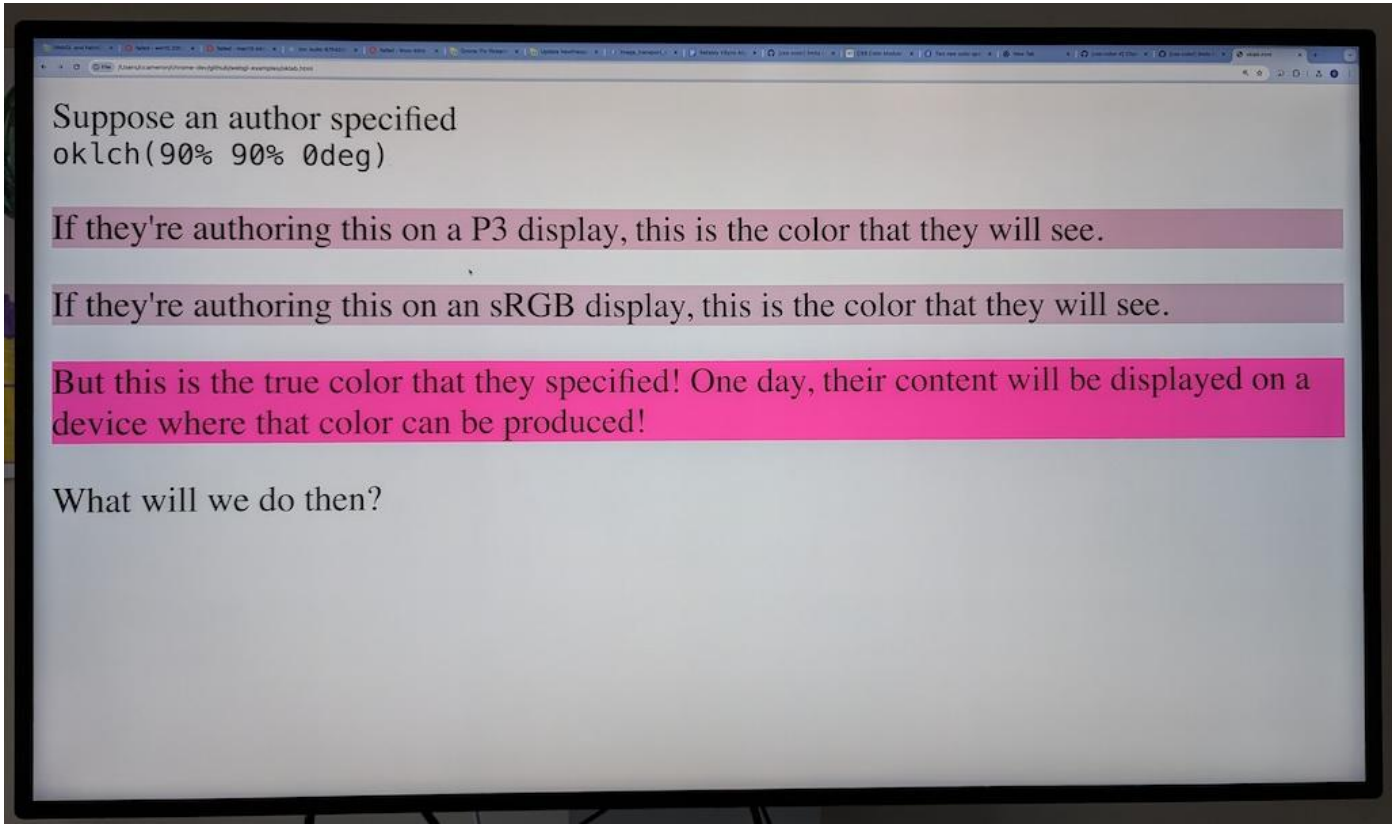
It encourages “future-dangerous” content

Encouraging future-dangerous content

Use of imaginary and extremely-out-of-gamut colors is a dangerous and unstable foundation important functionality, and should be discouraged.

Authors should only specify colors that are in the gamut of the device they are authoring content on.

Encouraging future-dangerous content



It is only useful for imaginary or extremely-out-of-gamut content.

When does gamut mapping improve quality?

Clipping P3 to sRGB produces extremely high quality results, and almost no existing displays are >>P3.

If any content author is specifying colors near the gamut of their display, CSS gamut mapping provides no value.

When does gamut mapping improve quality?

Clipping Rec2020 to sRGB is *unknown* quality.

Evidence suggests that clipping produces good quality here.

Evidence suggests that CSS gamut mapping produces poor quality.

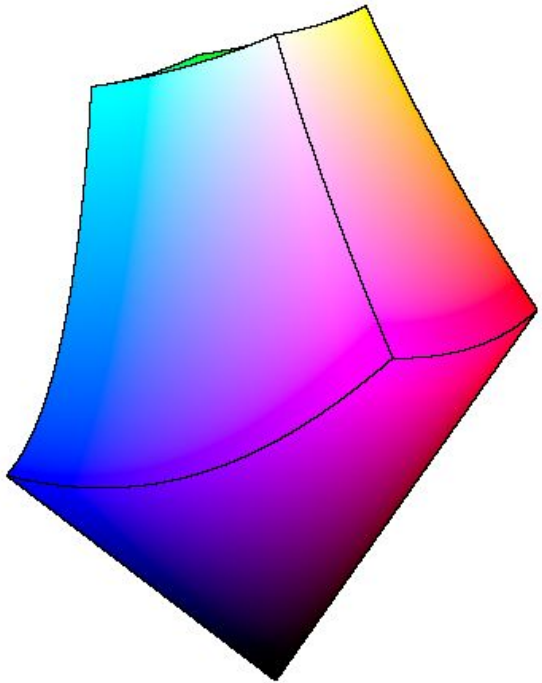
When does gamut mapping improve quality?

Clipping Rec2020 to sRGB is *unknown* quality.

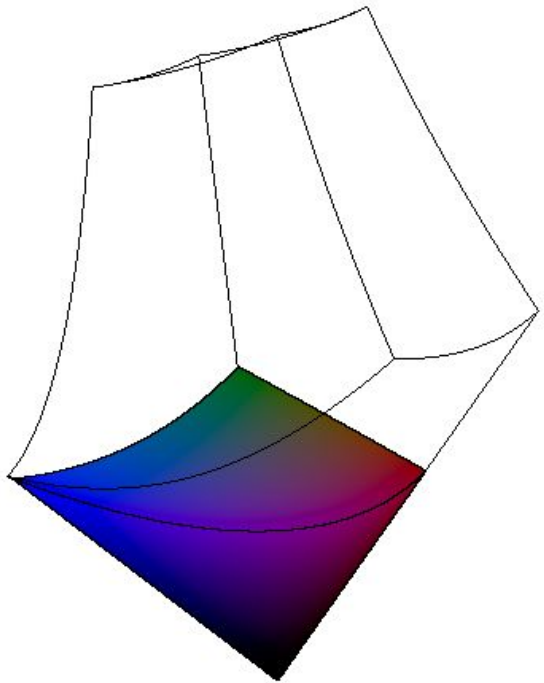
Evidence suggests that clipping produces good quality here.

Evidence suggests that CSS gamut mapping produces poor quality.

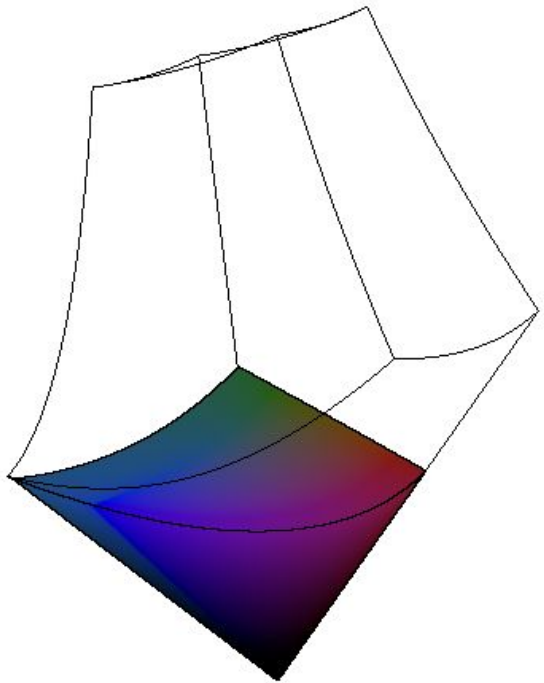
Rec2020: Clipping to sRGB



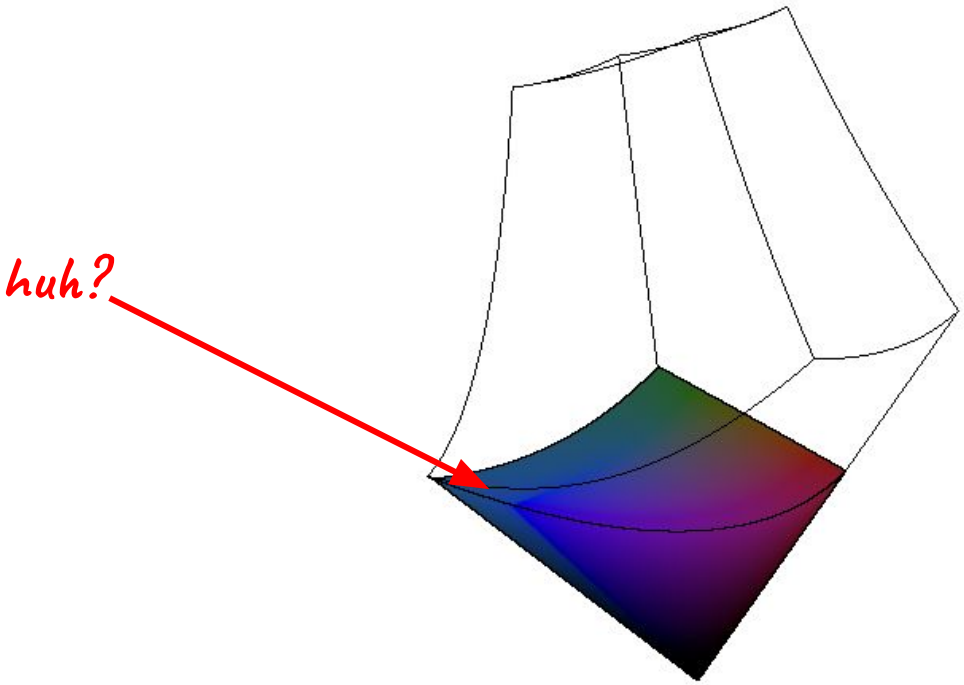
Rec2020: Clipping to sRGB



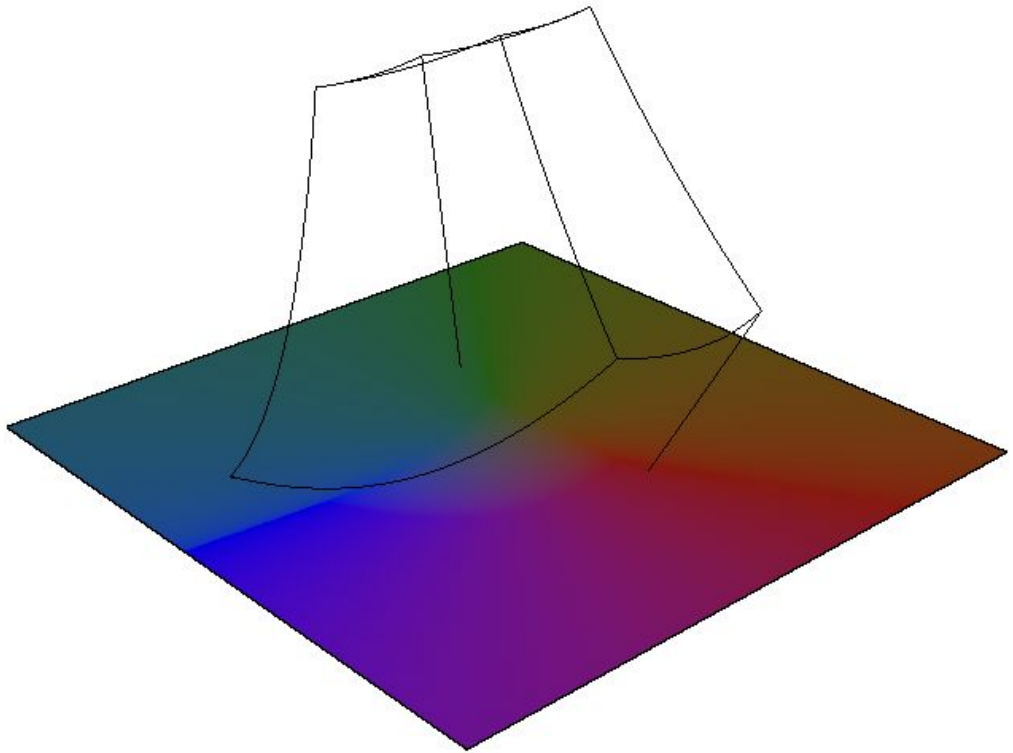
Rec2020: CSS gamut mapping to sRGB



Rec2020: CSS gamut mapping to sRGB



Rec2020: CSS gamut mapping to sRGB



Proposal

Current spec

Attempts to deliver safe (hue, chroma, luma) space via oklch space

- This is not what oklch is designed for
- This results in imaginary and extremely-out-of-gamut colors
- This is “fixed up” by imposing a meaning on oklch via CSS gamut mapping

Current spec

Attempts to deliver safe (hue, chroma, luma) space via oklch space

- This is not what oklch is designed for
- This results in imaginary and extremely-out-of-gamut colors
- This is “fixed up” by imposing a meaning on oklch via CSS gamut mapping

deliver a space (or spaces) that are designed for this

Proposal

Provide safe (hue, chroma, luma) parameter spaces for specifying and manipulating colors.

Discourage the use of extremely-out-of-gamut and imaginary colors in CSS.