2D Canvas: Wide Color Gamut & High Dynamic Range

Chris Cameron (Google)

Wide Color Gamut

Existing Wide Color Gamut Web Content!

Existing Wide Color Gamut Web Content!

Media

- <img> and <video>

Existing Wide Color Gamut Web Content!

Media

- <img> and <video>

CSS Colors

- #840628
- rgb(132, 6, 40)

Existing Wide Color Gamut Web Content!

Media

- <img> and <video>

CSS Colors (CSS Color Level 4)!

- #840628
- rgb(132, 6, 40)
- color(sRGB 0.52 0.02 0.16)
- color(display-p3 0.47 0.09 0.16)

2D Canvas

Color managed.

De facto sRGB.

2D Canvas

Color managed.

De facto sRGB.

```
ctx = document.getElementById('myCanvas').getContext(
    '2d');
ctx.drawImage(myDisplayP3Image, 0, 0);
ctx.fillStyle = 'color(display-p3 1 0 0);
ctx.fillRect(0, 0, 256, 256);
```

2D Canvas

Color managed.

De facto sRGB.

```
ctx = document.getElementById('myCanvas').getContext(
    '2d', {colorSpace: 'display-p3'});
ctx.drawImage(myDisplayP3Image, 0, 0);
ctx.fillStyle = 'color(display-p3 1 0 0);
ctx.fillRect(0, 0, 256, 256);
```

ImageData

Bitmap representation for CPU pixel manipulation.

De facto sRGB.

ImageData

Bitmap representation for CPU pixel manipulation.

De facto sRGB.

```
imgData = new ImageData(width, height);

imgData.data[0] = 132;
imgData.data[1] = 6;
imgData.data[2] = 40;
imgData.data[3] = 255;
// ...
context.putImageData(imgData, 0, 0);
```

ImageData

Bitmap representation for CPU pixel manipulation.

De facto sRGB.

```
imgData = new ImageData(width, height,
                        {colorSpace:'display-p3'});
imgData.data[0] = 132;
imgData.data[1] = 6;
imgData.data[2] = 40;
imgData.data[3] = 255;
// ...
context.putImageData(imgData, 0, 0);
```

ImageBitmap

Efficient and asynchronous bitmap representation.

De facto sRGB.

ImageBitmap

Efficient and asynchronous bitmap representation.

De facto sRGB.

```
blob = await fetch(myImageURL);
bitmap = await createImageBitmap(blob);

gl.texImage2D(gl.TEXTURE_2D, 0, gl.RGBA, gl.RGBA,
              gl.UNSIGNED_BYTE, bitmap);
```

ImageBitmap

Efficient and asynchronous bitmap representation.

De facto sRGB.

```
blob = await fetch(myImageURL);
bitmap = await createImageBitmap(blob,
                                 {colorSpace: 'display-p3'});
gl.texImage2D(gl.TEXTURE_2D, 0, gl.RGBA, gl.RGBA,
              gl.UNSIGNED_BYTE, bitmap);
```

High Dynamic Range

Existing High Dynamic Range (Web) Content!

Media

- Hybrid-log gamma (HLG) and Perceptual quantizer (PQ), 10 bit fixed point

Games

- Extended linear sRGB, 16 bit floating point

High Dynamic Range: More color spaces, more bits!

Proposal

```
context = element.getContext('2d',
    {colorSpace: 'rec2100-hlg', storageFormat:'unorm10'}

                                or

    {colorSpace: 'rec2100-pq', storageFormat:'unorm10'}

                                or

    {colorSpace: 'srgb-linear', storageFormat:'float16'});
```

High Dynamic Range Difficulties

More power!

- Don’t use it unless you need it!

Fingerprinting

- Can’t expose display properties

Converting colors between HDR spaces

- ITU 2408 is an option, can we make it simpler?

Thank You!