Houdini

What lies ahead

Arun Michael Dsouza

Software Engineer, AdPushup Inc

@amdsouza92

JSConf Iceland 2018

Arun Michael Dsouza

New Delhi, India

adpushup

CSS Houdini

Source(s) : shareicon.net iconarchive.com

1994

W3C

1995-2007

DOM, Ajax, jQuery

2010

Polyfills

remysharp.com/2010/10/08/what-is-a-polyfill

2013

The Extensible Web Manifesto

Brendan Eich, Yehuda Katz, Alex Russell, Brian Kardell, Chris Eppstein, Paul Irish, Tab Atkins and more…

extensiblewebmanifesto.org

The underlying magic…

Polyfill new ES feature ?

Polyfill new CSS layout ?

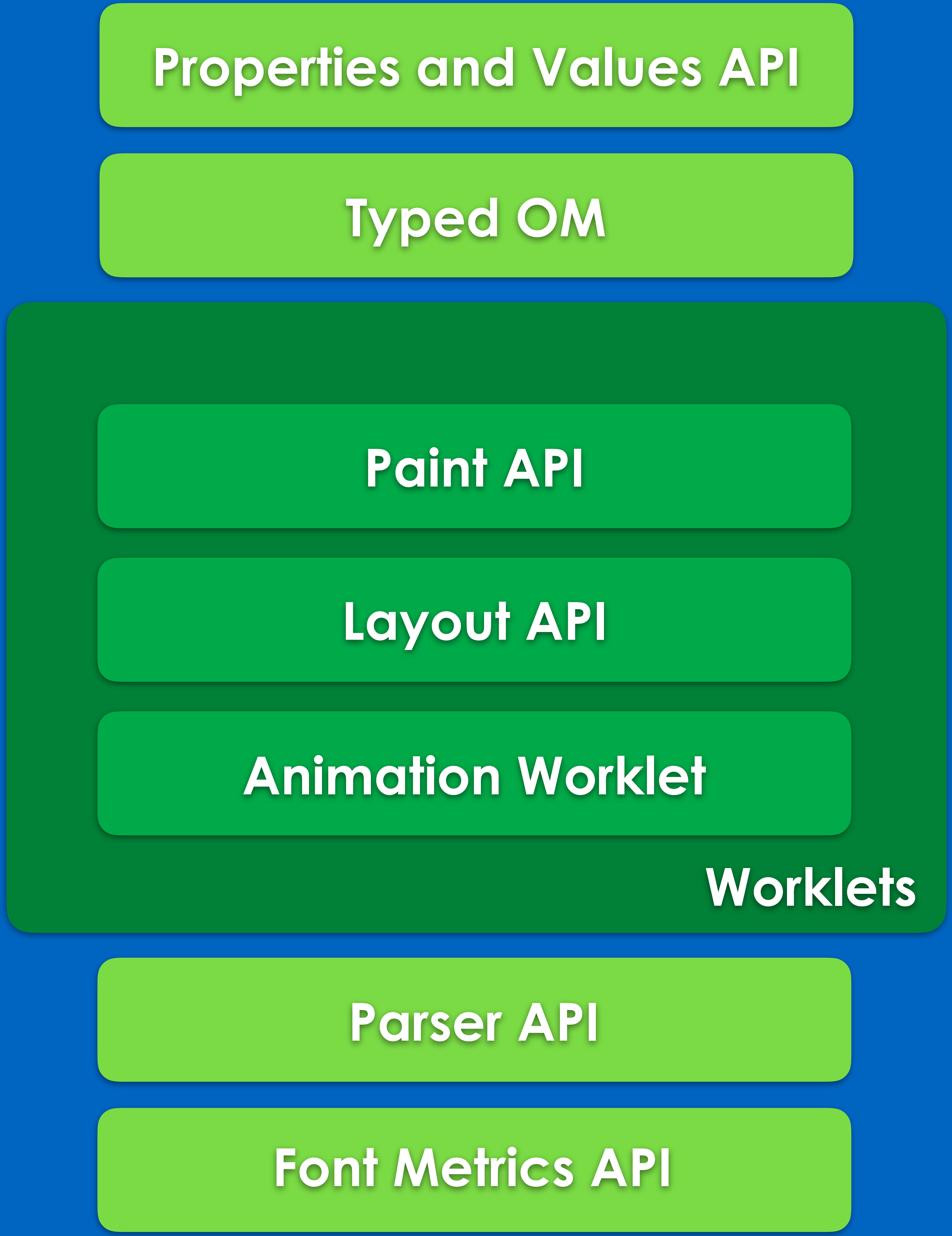
Where’s the underlying CSS magic ?

It’s all hidden!

“Houdini”

Source(s) : wikimedia.org

“CSS Houdini is a W3C effort to define lower-level CSS APIs for authors to understand, recreate, and extend highlevel CSS authoring features.


Properties and Values API
Typed OM
Paint API
Layout API
Animation Worklet
Worklets
Parser API
Font Metrics API

Properties and Values API

bit.ly/css-properties-and-values-api

- Extends the **CSS Variables** spec
- Property values can have a **type**
- Support to set an **initial value**
- Support to define **inheritance behaviour**

```
window.CSS.registerProperty({
  name: "--bgColor",
  syntax: "<color>",
  initialValue: "black",
  inherits: true
});
```

```
window.CSS.registerProperty({
  name: "--bgColor",
  syntax: "<color>",
  initialValue: "black"
});
```

```
window.CSS.registerProperty({
  name: "--bgColor",
  syntax: "<color>",
  initialValue: "black"
});
```

```
window.CSS.registerProperty({
  name: "--bgColor",
  syntax: "<color>",
  initialValue: "black"
});
```

```
.thing {
  background-color: var(--bgColor);
}
```

```
window.CSS.registerProperty({
  name: "--bgColor",
  syntax: "<color>",
  initialValue: "black"
});
```

```
.thing {
  --bgColor: green;
  background-color: var(--bgColor);
}
```

```
window.CSS.registerProperty({
  name: "--bgColor",
  syntax: "<color>",
  initialValue: "black"
});
```

```
.thing {
  --bgColor: “not-a-color”;
  background-color: var(--bgColor);
}
```

<color>, <number>, <percentage>, <url> …

bit.ly/css-properties-and-values-api

Typed OM

bit.ly/css-typed-om-api

- **Typed** value support via JS
- **Performant** manipulation of property values

CSSStyleValue

CSSLengthValue

CSSTransformValue

CSSPositionValue

CSSMathValue

bit.ly/cssstylevalue-subclasses

Style Map

```
const styleMap = document.getElementById("myElement").styleMap;
```

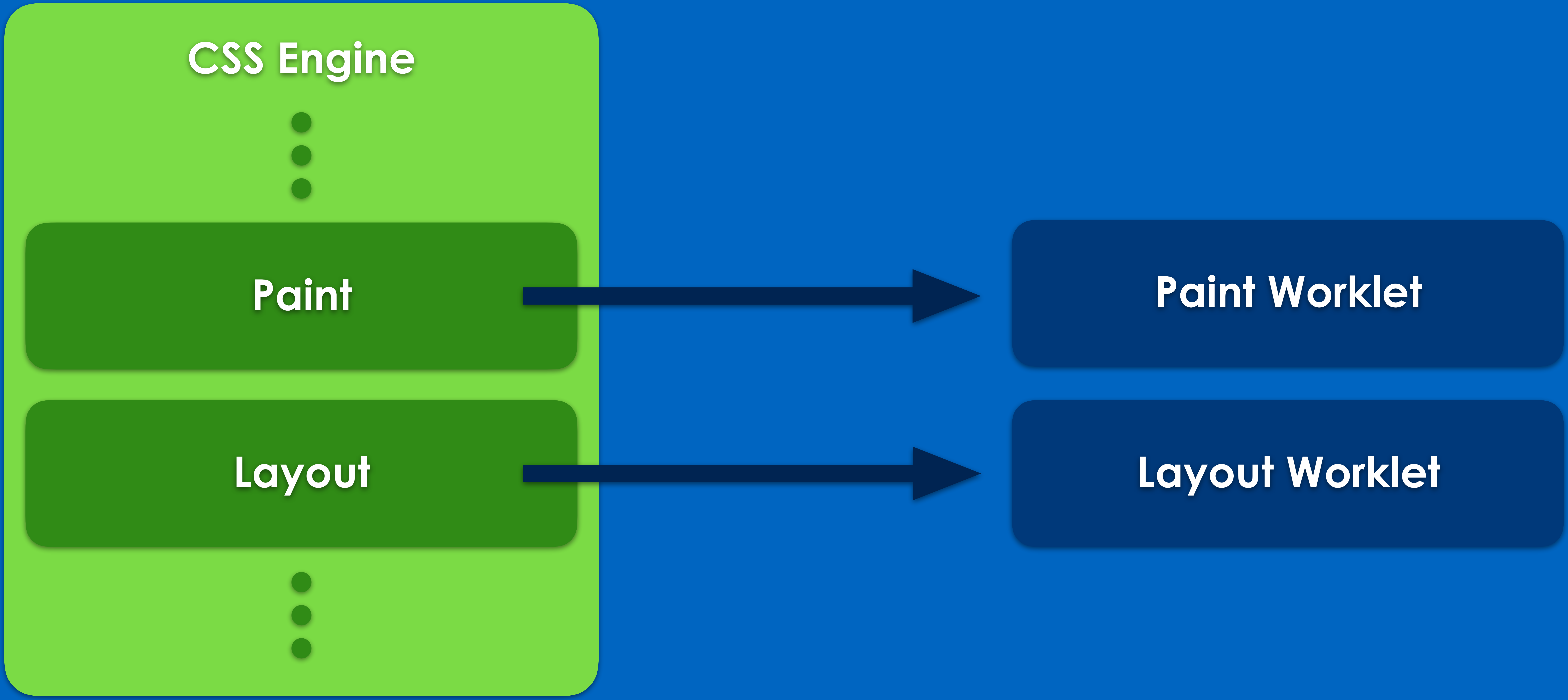
```
// Set new property value
styleMap.set("height", new CSSSimpleLength(100, "px"));

// Get property value
styleMap.get("height");

// -> Returns height as a subclass of CSSStyleValue
```

Typed OM Polyfill

bit.ly/typed-om-polyfill

Worklets

bit.ly/worklets


CSS Engine
Paint
Layout
Paint Worklet
Layout Worklet

- **Worker scripts** for Houdini APIs
- **Independent** of the main thread

```
// index.html
window.CSS.paintWorklet.addModule("paint-worklet.js");
```

```
// paint-worklet.js
registerPaint("checkerboard", class CheckerboardPainter {
  ...
}
```

Paint API

bit.ly/css-paint-api

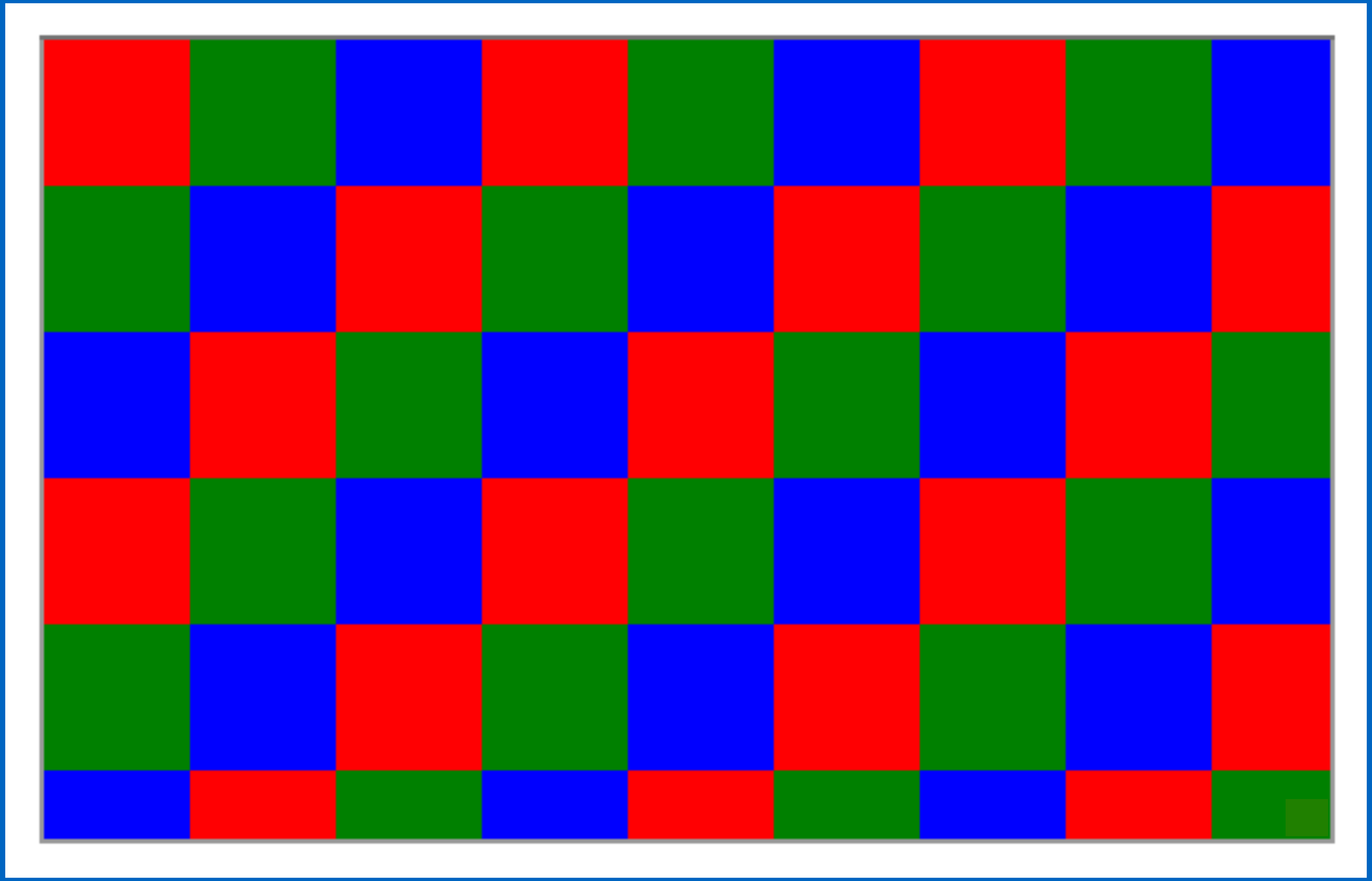
“The paint stage is responsible for painting the background, content and highlight of a box based on that box’s size (as generated by the layout stage) and computed style.

```css
background-image: paint(mypaint);
```

```
// index.html
<style>
    div {
        width: 600px;
        height: 400px;
        background-image: paint(checkerboard);
    }
</style>
<div></div>
<script>
    window.CSS.paintWorklet.addModule("checkerboard.js");
</script>
```

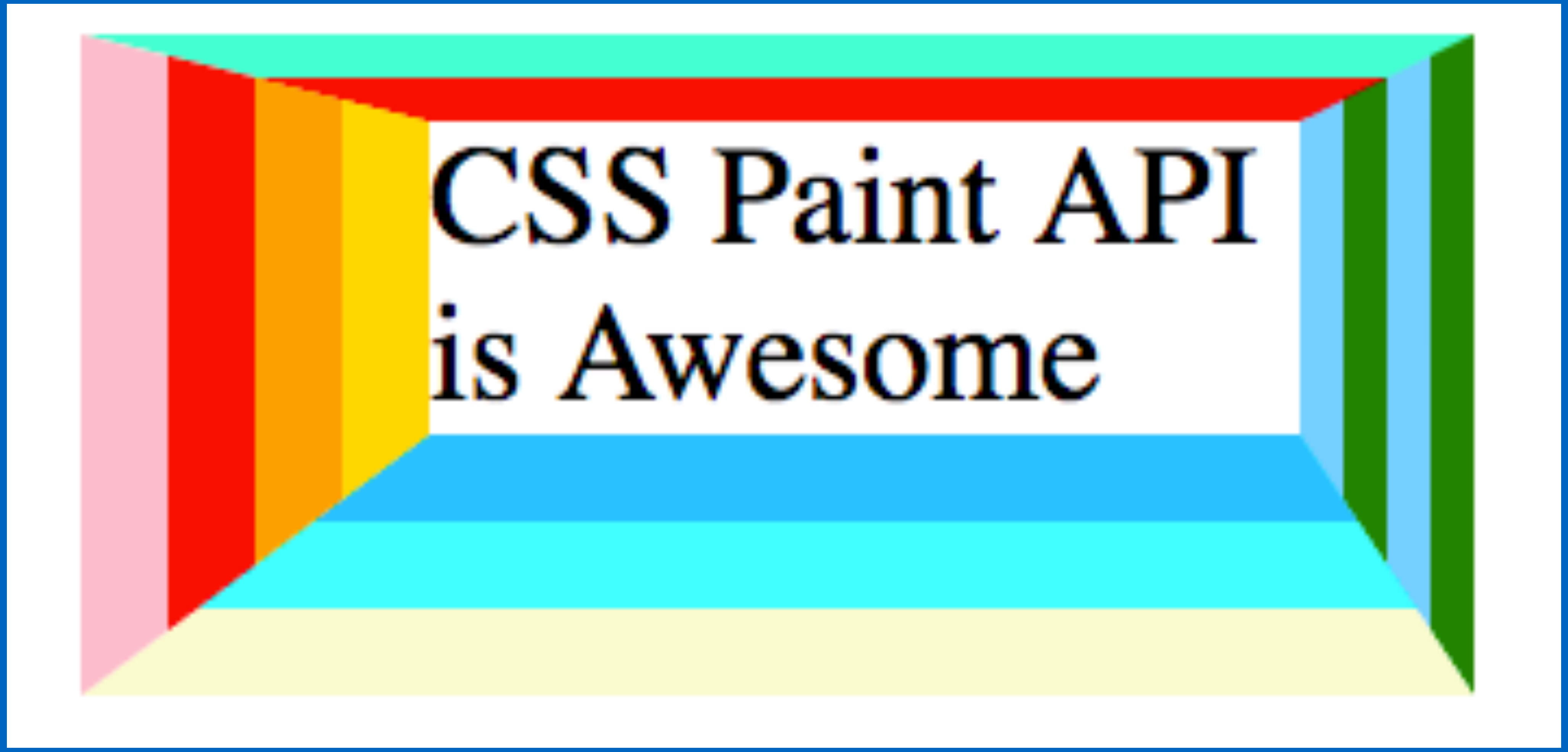
bit.ly/paint-api-demo

```
// checkerboard.js
class CheckerboardPainter {
  paint(ctx, geom, properties) {
    const colors = ["red", "green", "blue"];
    const size = 32;
    for(let y = 0; y < geom.height/size; y++) {
      for(let x = 0; x < geom.width/size; x++) {
        const color = colors[(x + y) % colors.length];
        ctx.beginPath();
        ctx.fillStyle = color;
        ctx.rect(x * size, y * size, size, size);
        ctx.fill();
      }
    }
  }
}

registerPaint("checkerboard", CheckerboardPainter)
```



```
<style>
    div {
        —checkerboard-spacing: 20;
        —checkerboard-size: 12;
    }
</style>
```

```
class CheckerboardPainter {
  static get inputProperties() { return ["--checkerboard-spacing", "--checkerboard-size"]; }

  paint(ctx, geom, properties) {
    const size = parseInt(properties.get("--checkerboard-size").toString());
    const spacing = parseInt(properties.get("--checkerboard-spacing").toString());
    ...
  }
}
```



bit.ly/css-paint-worklet-samples

Layout API

bit.ly/css-layout-api

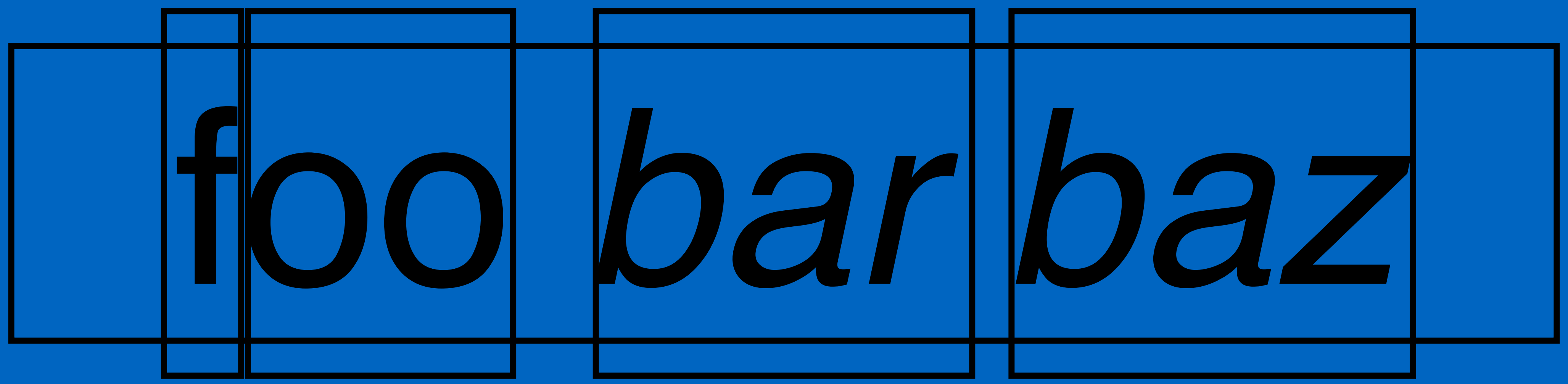
“The layout stage is responsible for generating and positioning fragments from the box tree.

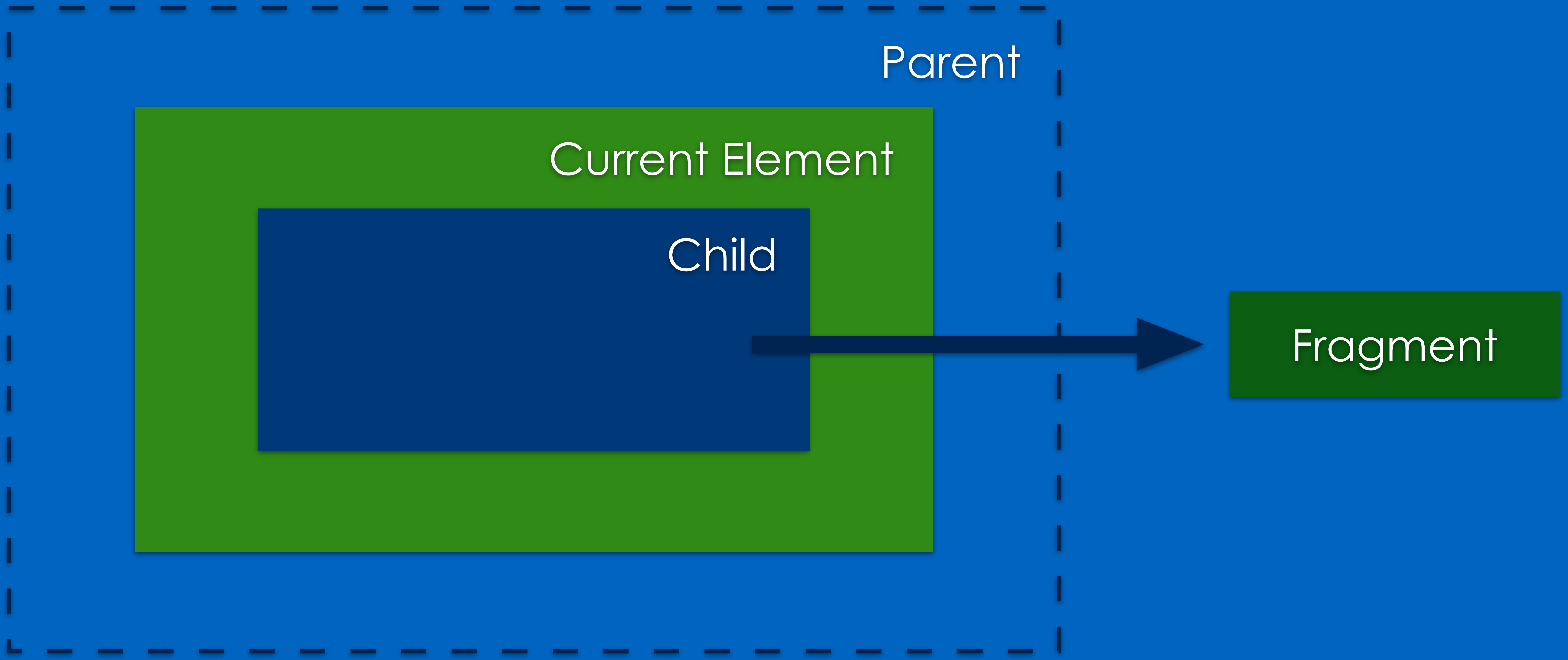
Box Tree

- Represents the formatting structure of the rendered document
- Each box in the box tree represents its corresponding element or pseudo element

Fragments

```
<style>
  p::first-line { color: green; }

  p::first-letter { color: red; }
</style>
<p>foo <i>bar baz</i></p>
```




Parent
Current Element
Child
Fragment

```html
// index.html
<style>
    div {
        width: 50px;
        height: 50px;
        display: layout(block-like);
    }
</style>
<div>
  . . .
</div>
<script>
    CSS.layoutWorklet.addModule("block-like.js");
</script>
```

```js
// block-like.js
class BlockLike {
    static get inputProperties() { return ["--foo"]; }
    static get childrenInputProperties() { return ["--bar"]; }
    static get childDisplay() { return "normal"; }

    *intrinsicSizes(children, styleMap) {
        // Intrinsic sizes code goes here.
    }

    *layout(space, children, styleMap, edges, breakToken) {
        // Layout code goes here.
    }
}

registerLayout("block-like", BlockLike);
```

```
. . .

*intrinsicSizes(styleMap, children) {
    const childrenSizes = yield children.map((child) => {
        return child.intrinsicSizes();
    });

    const maxContentSize = childrenSizes.reduce((sum, childSizes) => {
        return sum + childSizes.maxContentContribution;
    }, 0);

    const minContentSize = childrenSizes.reduce((max, childSizes) => {
        return sum + childSizes.minContentContribution;
    }, 0);

    return { maxContentSize, minContentSize };
}

. . .
```

```
*layout(space, children, styleMap, edges, breakToken) {
    const inlineSize = resolveInlineSize(space, styleMap);

    const availableInlineSize = inlineSize - edges.all.inline;
    const availableBlockSize =
        resolveBlockSize(space, styleMap) - edges.all.block;

    const childConstraintSpace = new ConstraintSpace({
        inlineSize: availableInlineSize,
        blockSize: availableBlockSize,
    });

    const unconstrainedChildFragments = yield children.map((child) => {
        return child.layoutNextFragment(childConstraintSpace);
    });

    // Position the fragments.
     . . .

    // Resolve our block size.
     . . .

    return {
        inlineSize: inlineSize,
        blockSize: blockSize,
        childFragments: childFragments,
    };
}
```

bit.ly/css-layout-api

Animation Worklet

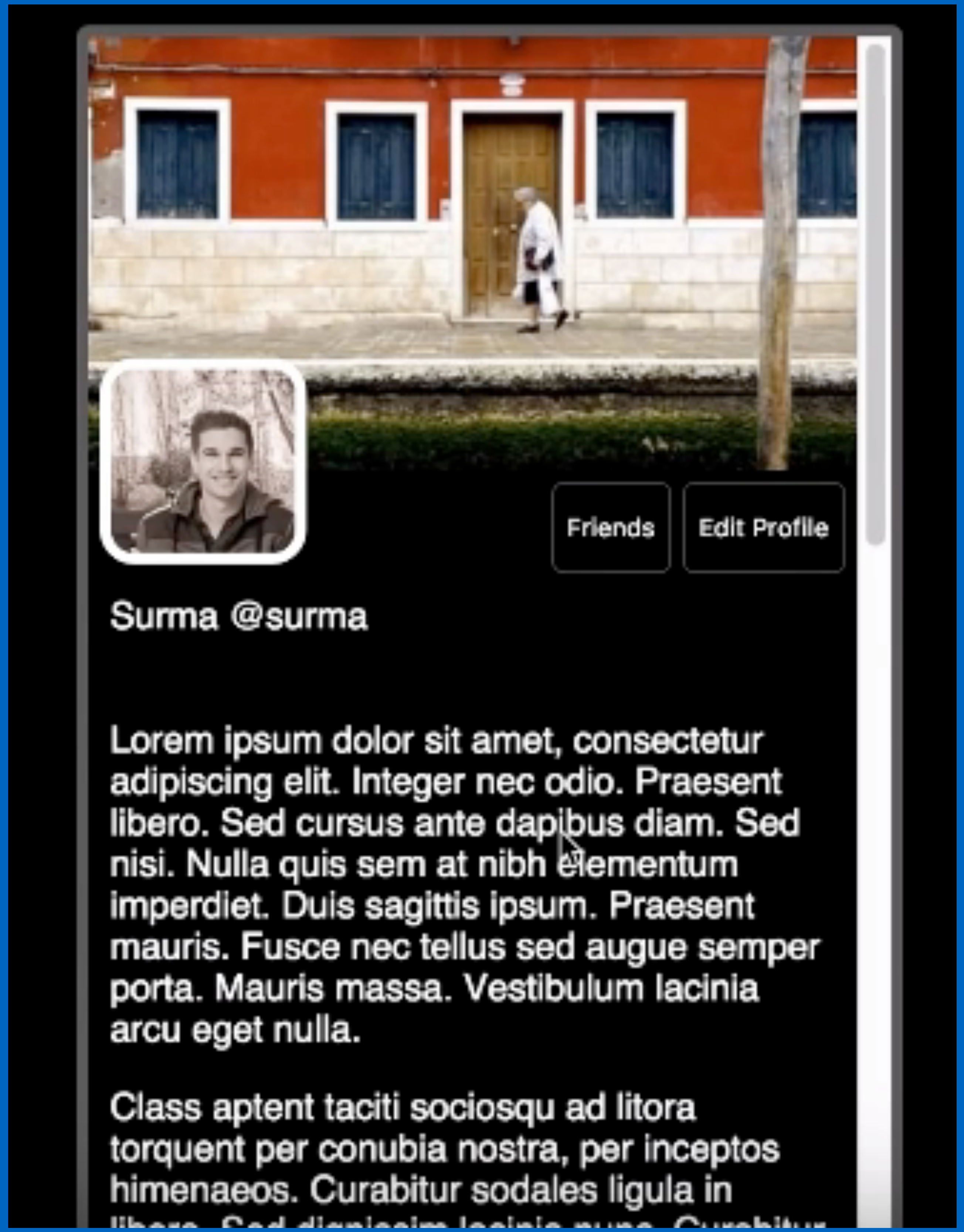
bit.ly/css-animation-worklet-api

- **High Performant** animations
- Exposes an **Animation** interface on the main thread

```html
// index.html
<div id="scrollingContainer">
    <section id="header"></section>
    <section>
        <picture id="avatar">
            <img src="avatar.jpg">
        </picture>
        <section class="profilecontrols">
            <button>Friends</button>
            <button>Edit Profile</button>
        </section>
    </section>
    <section class="profile">
        Surma @DasSurma
    </section>
    <section class="tweets">
        . . .
    </section>
</div>
```

```
// index.html
. . .

<script>
  window.animationWorklet.addModule("twitter-header-animator.js").then(_ => {
      const workletAnim = new WorkletAnimation("twitter-header",
       [new KeyFrameEffect($avatar, /* scales down as we scroll */ [{ transform: "scale(1)" }, { transform:
       "scale(0.5)" }], { duration: 1, iterations: 1 }),
       new KeyFrameEffect($header, /* loses transparency as we scroll */ [{ opacity: 0 }, { opacity: 0.8 }],
       { duration: 1, iterations: 1 })
       ],
       new ScrollTimeline($scrollingContainer, { timeRange: 1, startScrollOffset: 0, endScrollOffset:
       $header.clientHeight }),
      );
 });
</script>
```

```js
// twitter-header-animator.js.
registerAnimator("twitter-header", class {
  constructor(options) {
    this.timing_ = new CubicBezier("ease-out");
  }

  clamp(value, min, max) {
    return Math.min(Math.max(value, min), max);
  }

  animate(currentTime, effect) {
    const scroll = currentTime;

    effect.children[0].localTime = scroll;
    effect.children[1].localTime = this.timing_(clamp(scroll, 0, 0.5));
  }
});
```



Surma @surma

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Integer nec odio. Praesent libero. Sed cursus ante dapibus diam. Sed nisi. Nulla quis sem at nibh elementum imperdiet. Duis sagittis ipsum. Praesent mauris. Fusce nec tellus sed augue semper porta. Mauris massa. Vestibulum lacinia arcu eget nulla.

Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos himenaeos. Curabitur sodales ligula in

bit.ly/css-animation-worklet-samples

Parser API

bit.ly/css-parser-api

- **Parse CSS** rules or rulesets into Typed OM representations

Font Metrics API

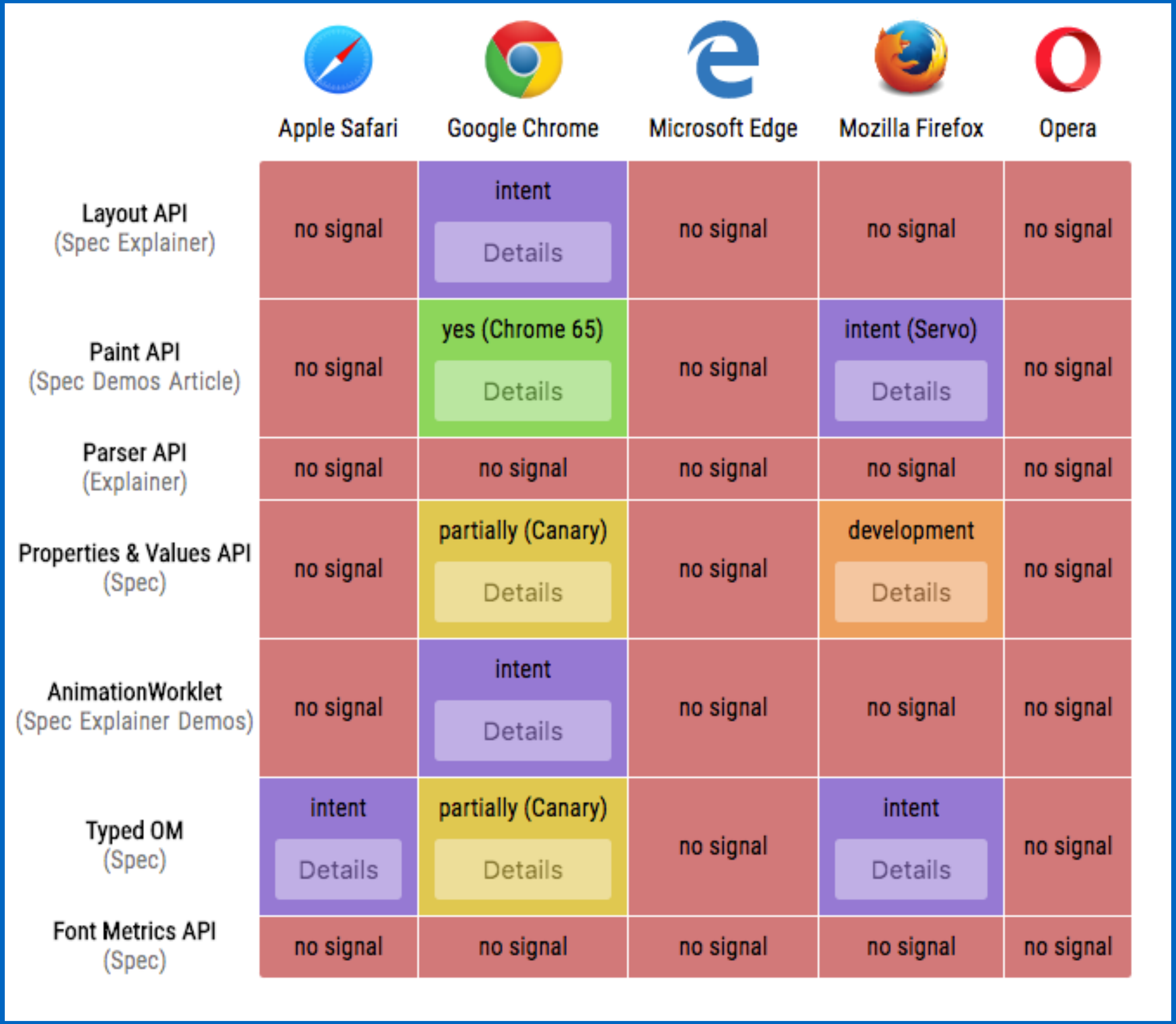
bit.ly/font-metrics-api

- Provides basic **Font Metrics** for our document content *measureElement( )*

measureText( )

Can we use Houdini today ?

Source(s) : giphy.com



	Apple Safari	Google Chrome	Microsoft Edge	Mozilla Firefox	Opera
Layout API (Spec Explainer)	no signal	intent Details	no signal	no signal	no signal
Paint API (Spec Demos Article)	no signal	yes (Chrome 65) Details	no signal	intent (Servo) Details	no signal
Parser API (Explainer)	no signal	no signal	no signal	no signal	no signal
Properties & Values API (Spec)	no signal	partially (Canary) Details	no signal	development Details	no signal
AnimationWorklet (Spec Explainer Demos)	no signal	intent Details	no signal	no signal	no signal
Typed OM (Spec)	intent Details	partially (Canary) Details	no signal	intent Details	no signal
Font Metrics API (Spec)	no signal	no signal	no signal	no signal	no signal

ishoudinireadyyet.com

Thank You!

Arun Michael Dsouza

Software Engineer, AdPushup Inc

@amdsouza92

JSConf Iceland 2018