



zenika

<animés par la passion>

CODING
THE WORLD

Let's talk about browser rendering

Singapore
Technozaure
2018





SOMMAIRE

- 1 Why this subject?
- 2 Basics
- 3 A bit of CSS
- 4 Tools?



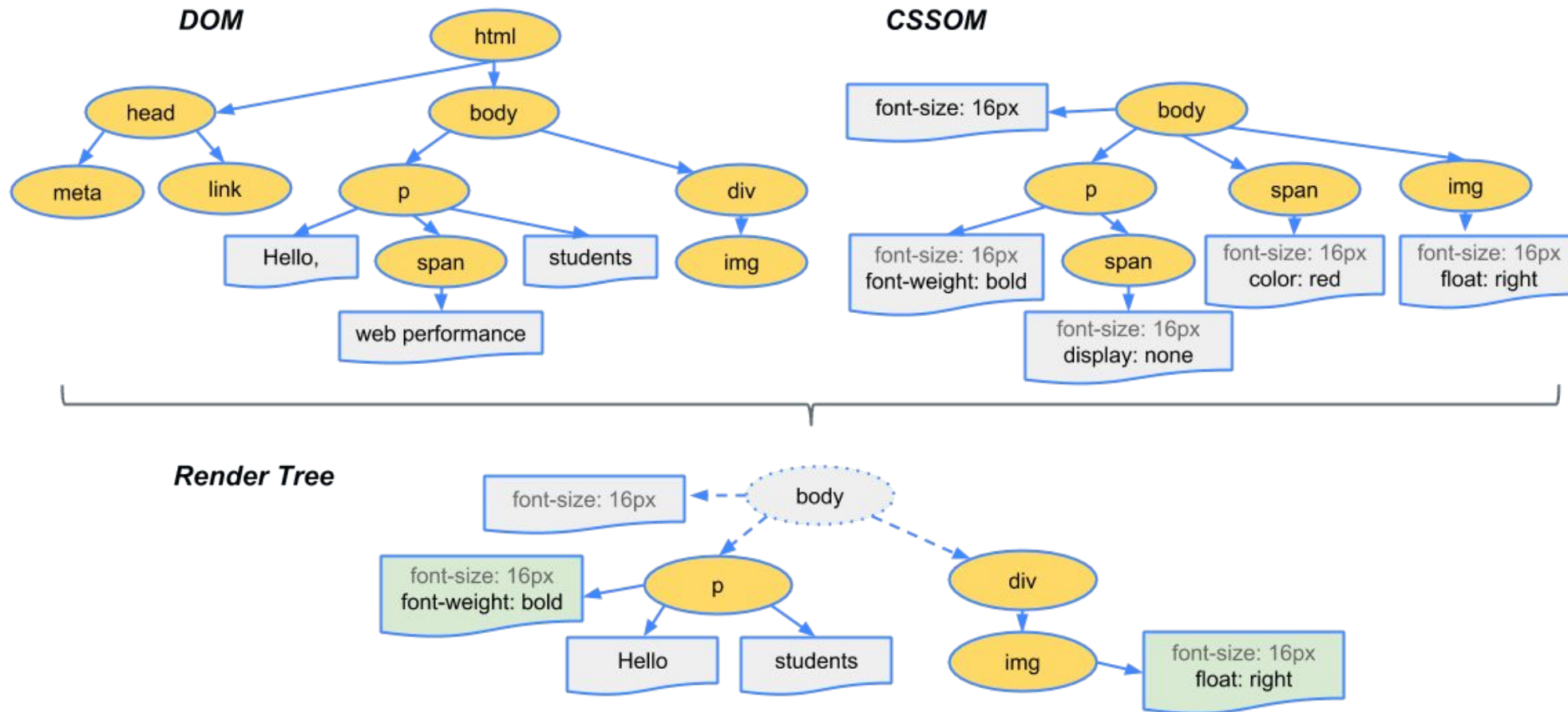


Why this subject?

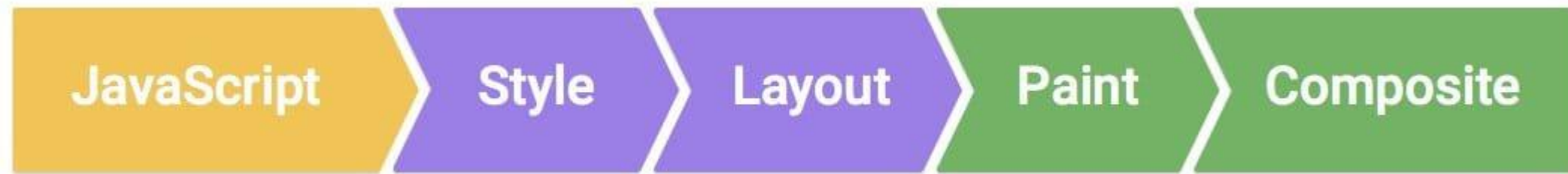
- USER EXPERIENCE
 - Most display now run on 60fps devices
 - For a smoother experience, display code (animation, processing) should be executed within 10ms
- OVERALL PERFORMANCE
 - Most users quit websites/apps whose displays take longer than 3s to update
 - Browsers are so fast nowadays that developers forget to optimize their UI code
- LET'S GET GREEN
 - High RAM / Network usage comes with a cost
 - Efficiency is our responsibility



Basics - initial rendering



Basics - Pixels-to-screen pipeline





Basics - Threads

```
for (var i = 0; i < paragraphs.length; i++) {  
    paragraphs[i].style.width = box.offsetWidth + 'px';  
}
```



Problem?



Basics - Threads

```
for (var i = 0; i < paragraphs.length; i++) {  
    paragraphs[i].style.width = box.offsetWidth + 'px';  
}
```



Problem?

```
function logBoxHeight() {  
    box.classList.add('super-big');  
    console.log(box.offsetHeight);  
}
```



Problem?





A bit of CSS - optimize complex layout?





A bit of CSS - optimize complex layout?

Learn and use *display:flex*!

(play froggy)

<https://flexboxfroggy.com/>





A bit of CSS - optimize selectors? (low priority)

style attribute	id	class	tag
1000	100	10	1



A bit of CSS - optimize selectors? (low priority)

style attribute	id	class	tag
1000	100	10	1

*div * {...}*

#id {...}

.class1 {...}

.class1.class2 {...}

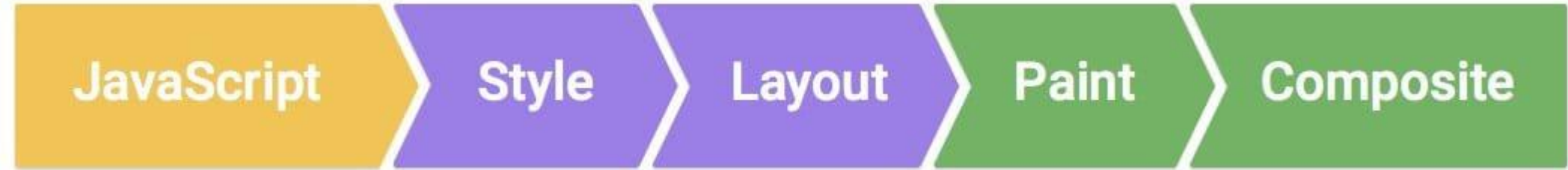
.list .list-item:last-child {...}

BEM

.list__list-item--last-child {...}



A bit of CSS - bypass layout and paint?



A bit of CSS - bypass layout and paint?

Position	<code>transform: translate(npx, npx);</code>
Scale	<code>transform: scale(n);</code>
Rotation	<code>transform: rotate(ndeg);</code>
Skew	<code>transform: skew(X Y)(ndeg);</code>
Matrix	<code>transform: matrix(3d)(...);</code>
Opacity	<code>opacity: 0...1;</code>

(The element will need to be on its own compositor layer.)



A bit of CSS - bypass layout and paint?

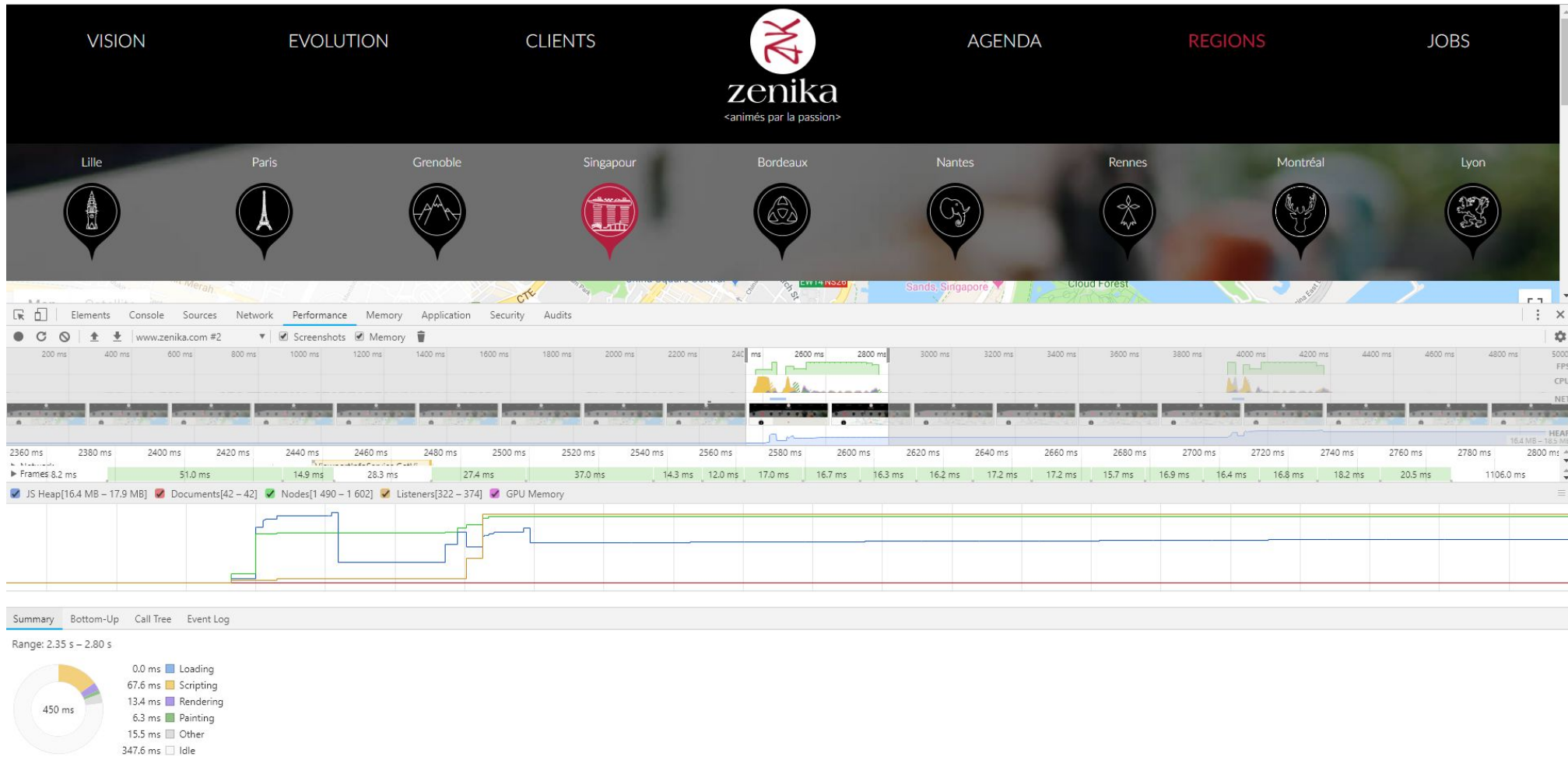
Position	<code>transform: translate(npx, npx);</code>
Scale	<code>transform: scale(n);</code>
Rotation	<code>transform: rotate(ndeg);</code>
Skew	<code>transform: skew(X Y) (ndeg);</code>
Matrix	<code>transform: matrix(3d)(...);</code>
Opacity	<code>opacity: 0...1;</code>

(The element will need to be on its own compositor layer.)

```
.moving-element {  
  will-change: transform;  
}
```



Tools - Chrome is fine



Tools - Lighthouse for quick overview

