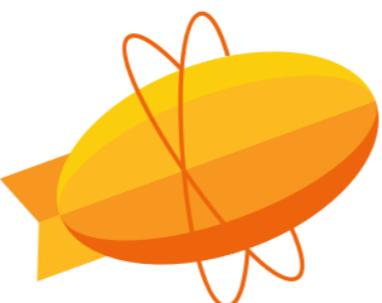


Web Performansı Ölçümü ve Optimizasyonu

Sertac Olgunsoylu
sertac@zeplin.io



ZEPLIN



Rebuilding Pinterest pages for performance resulted in a 40% decrease in wait time, a 15% increase in SEO traffic and a 15% increase in conversion rate to signup.



BBC has seen that they lose an additional 10% of users for every additional second it takes for their site to load

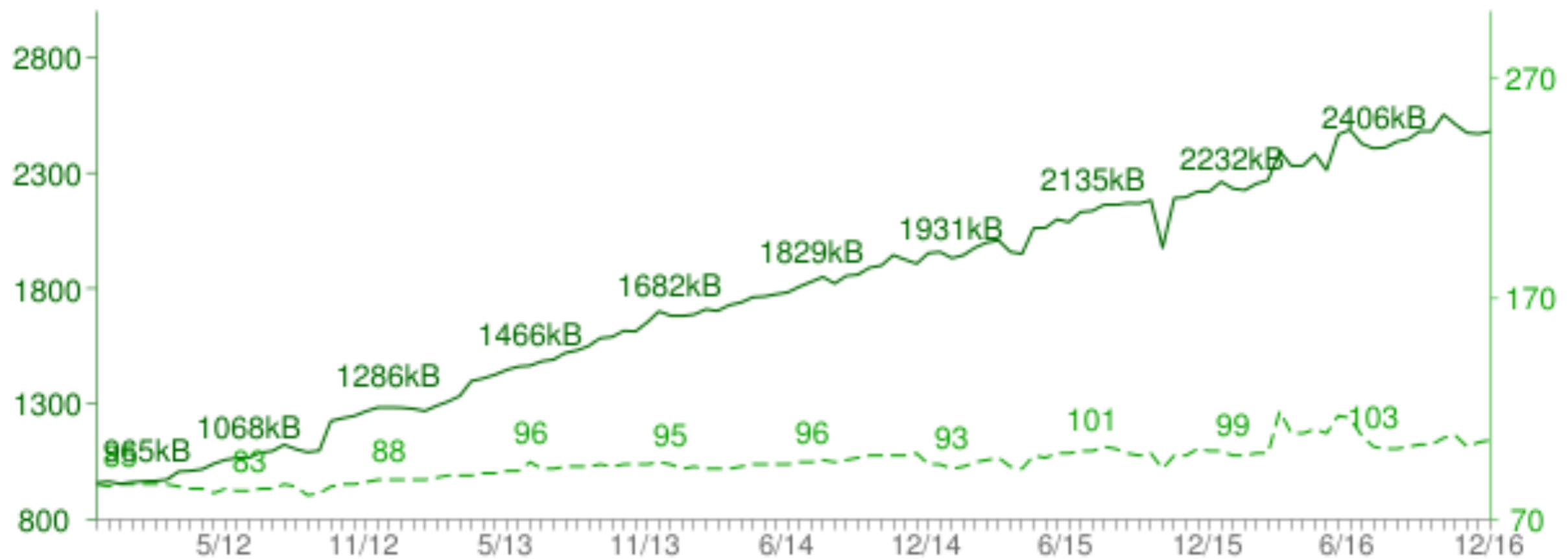


AliExpress reduced load time by 36% and saw a 10.5% increase in orders and a 27% increase in conversion for new customers.



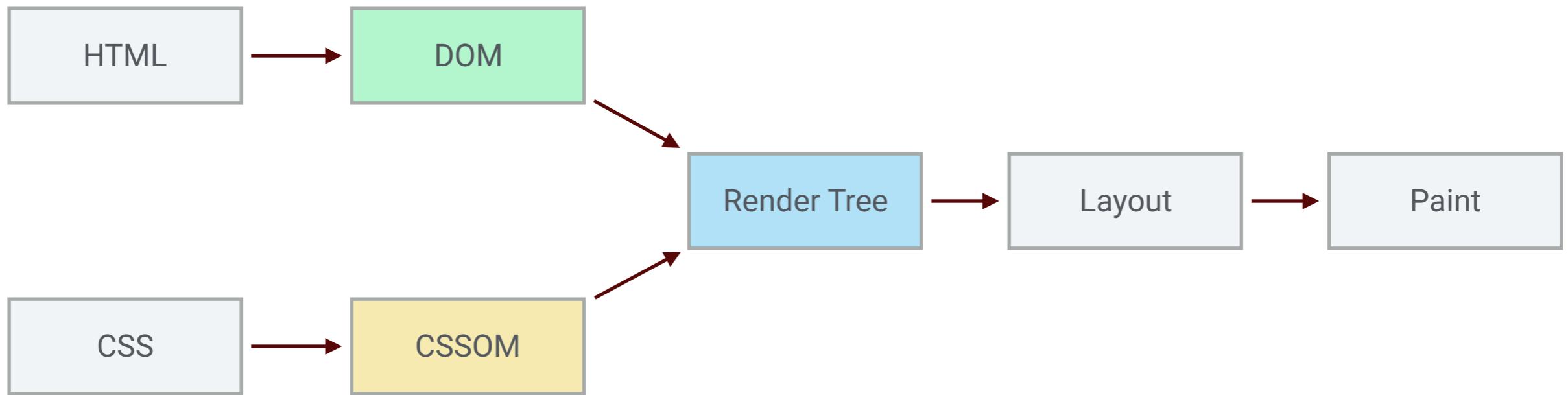
Optimizely added artificial latency to the Telegraph and saw page views plummet: by 11% for a 4 second delay and 44% for a 20 second delay.

Total Transfer Size & Total Requests



- Yüklenen verinin toplam boyutu
 - (HTML, CSS, JS, görseller, vs.)
- Sunucuya yapılan istek sayısı
 - Eşzamanlı maksimum 6 istek (aynı domain için)
- Ağ bağlantı hızı
- “Critical Rendering Path”

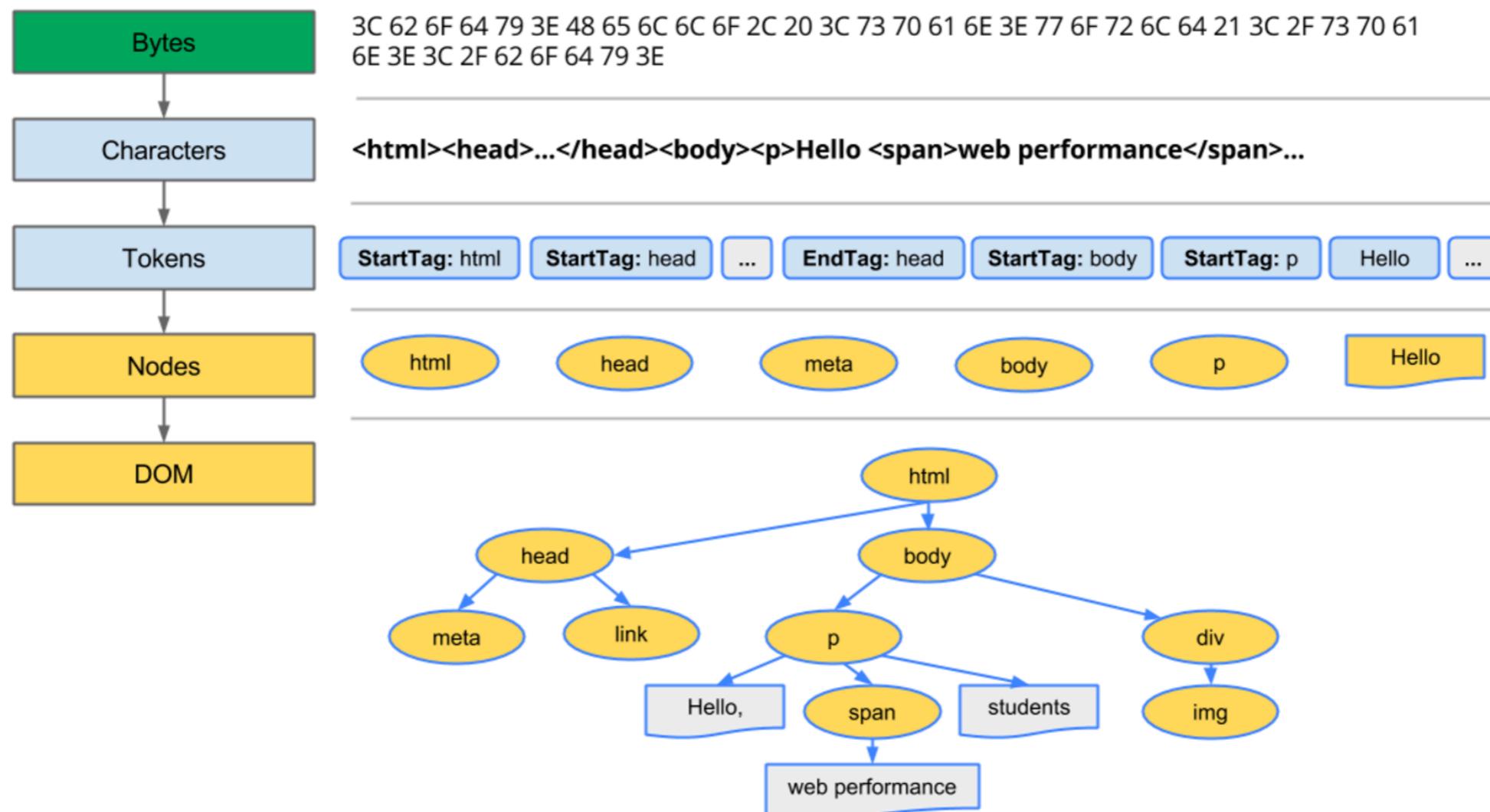
Critical Rendering Path



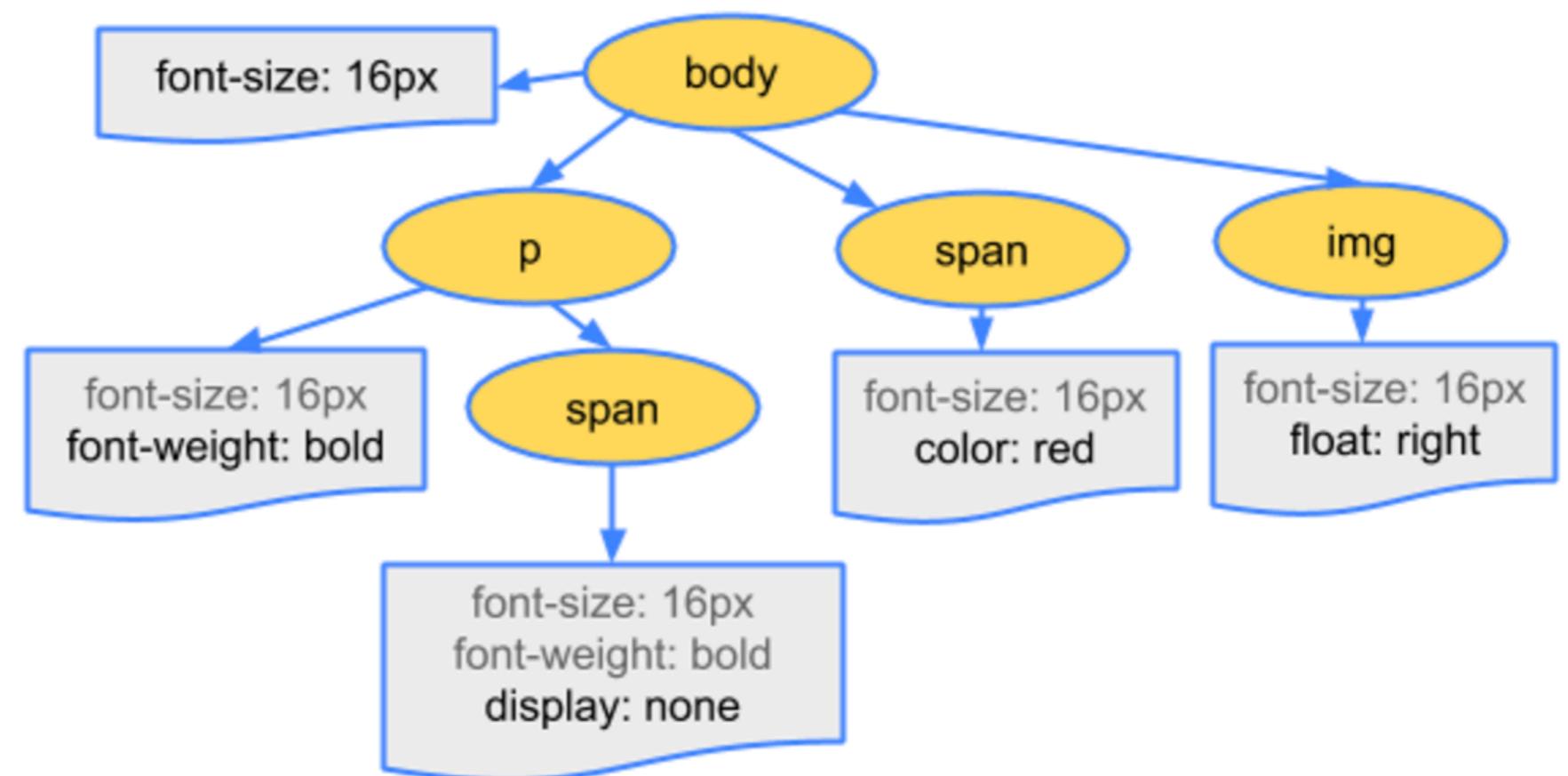
```

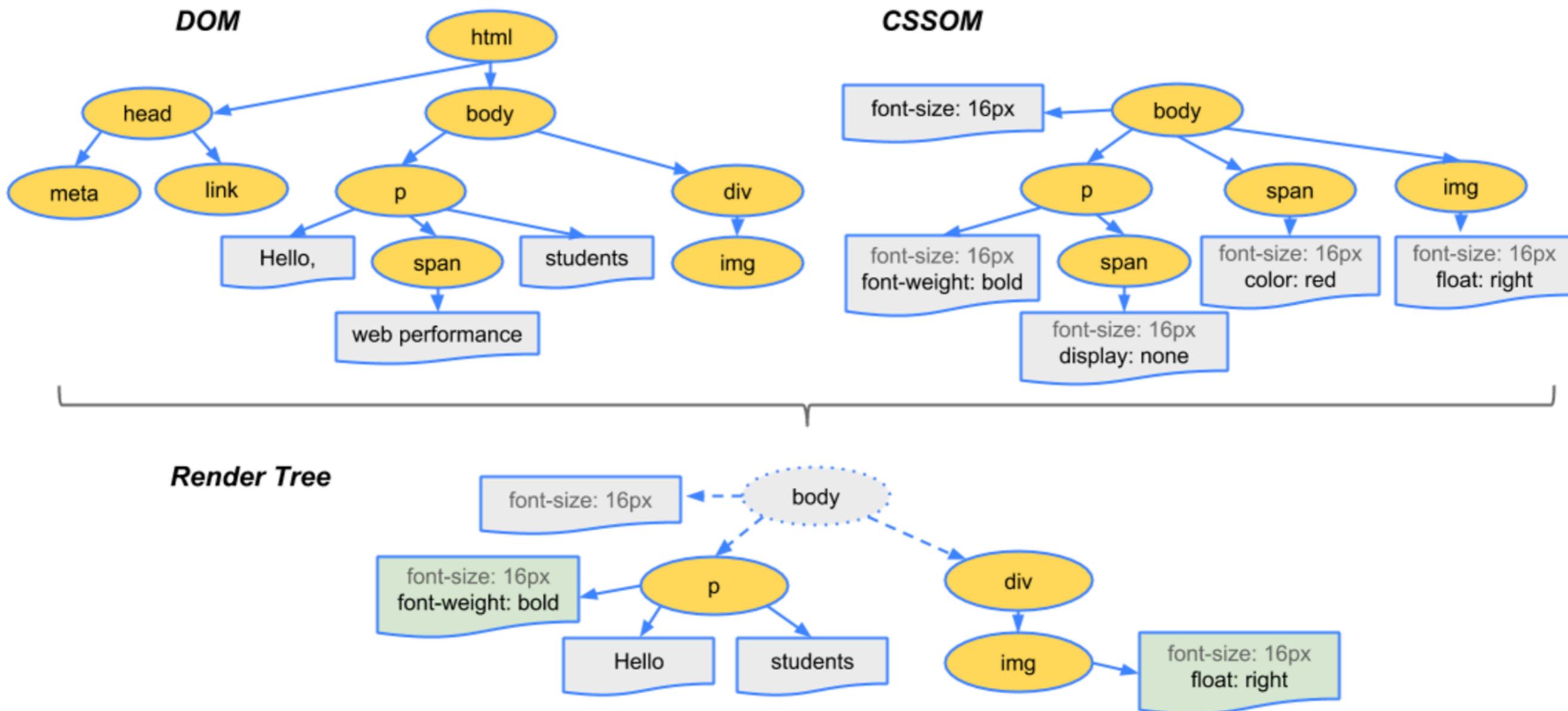
<html>
  <head>
    <meta name="viewport" content="width=device-width,initial-scale=1">
    <link href="style.css" rel="stylesheet">
    <title>Critical Path</title>
  </head>
  <body>
    <p>Hello <span>web performance</span> students!</p>
    <div></div>
  </body>
</html>

```

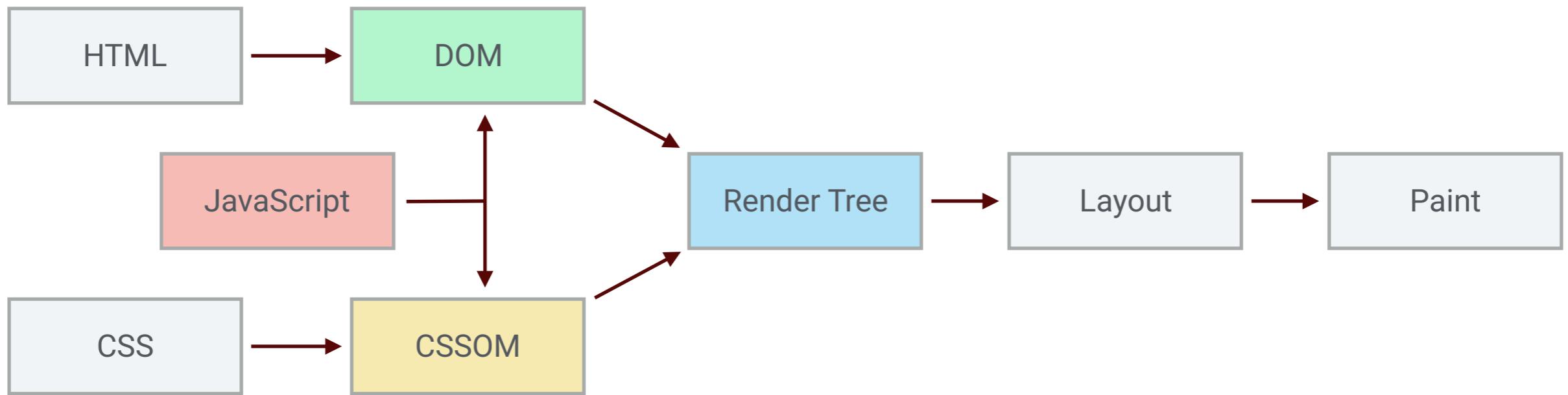


```
body {  
    font-size: 16px  
}  
  
p {  
    font-weight: bold  
}  
  
span {  
    color: red  
}  
  
p span {  
    display: none  
}  
  
img {  
    float: right  
}
```





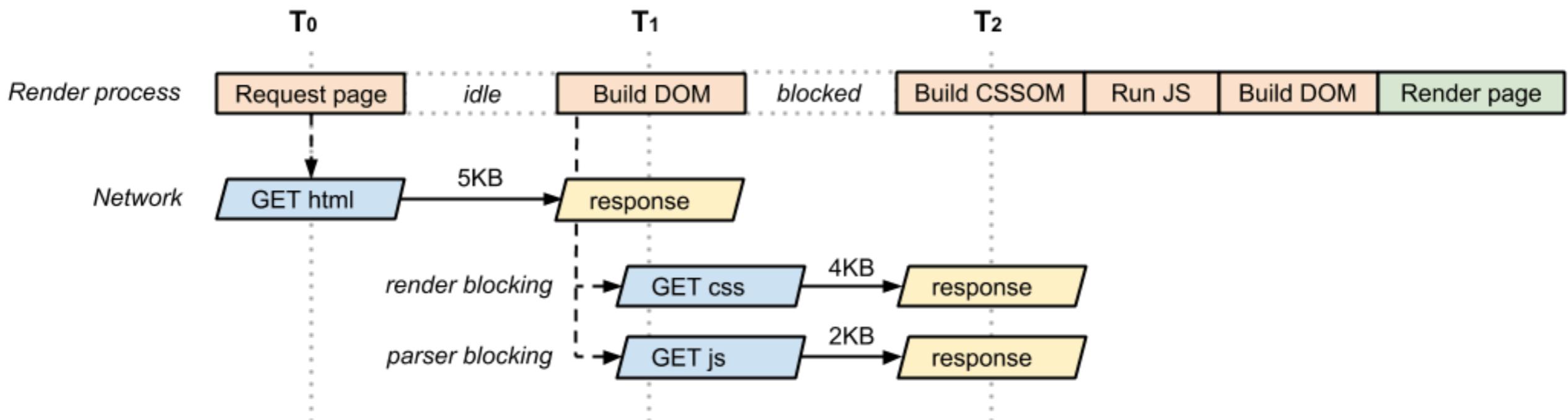
Critical Rendering Path



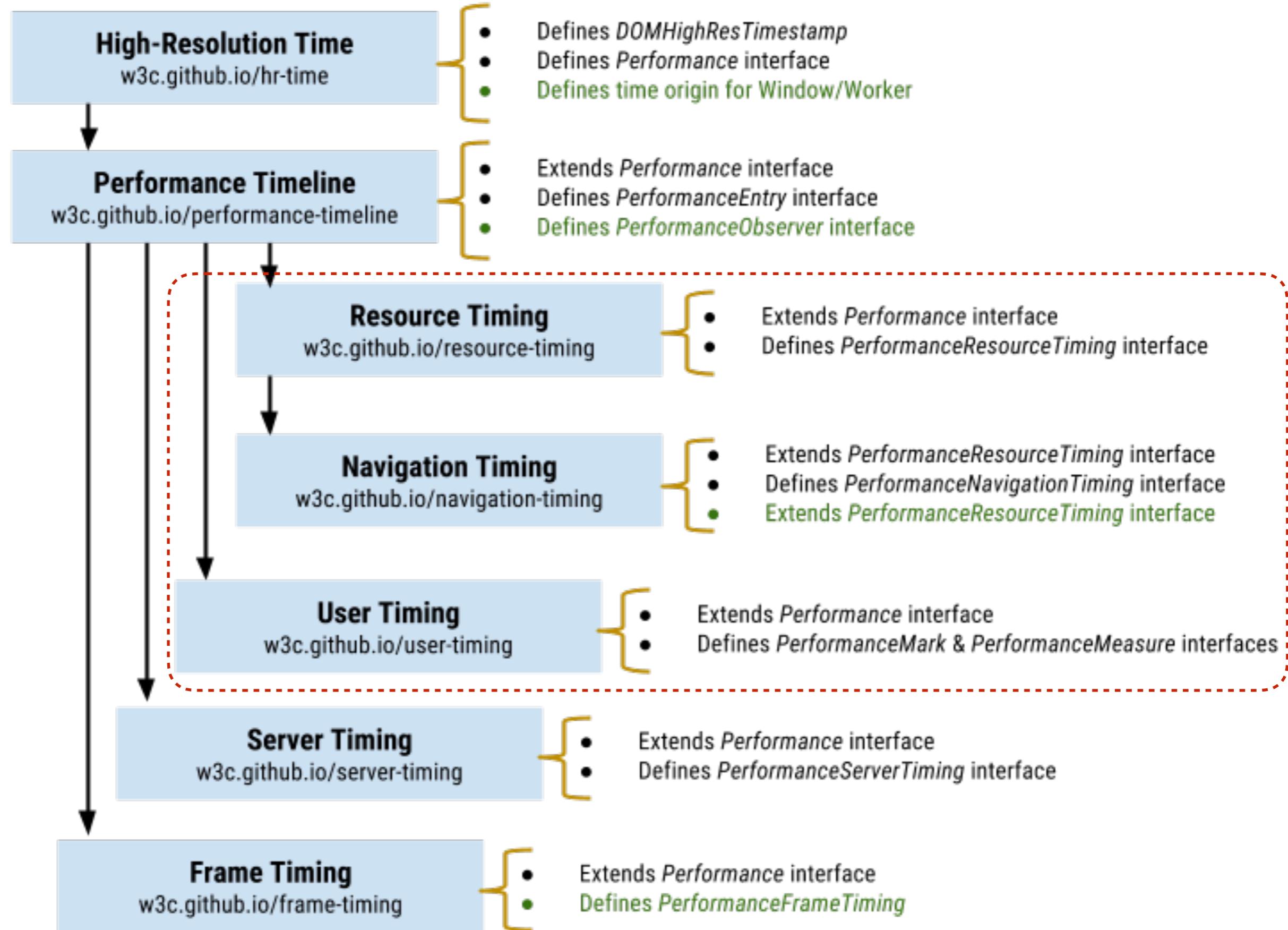
```

<html>
  <head>
    <meta name="viewport" content="width=device-width,initial-scale=1">
    <link href="style.css" rel="stylesheet">
  </head>
  <body>
    <p>Hello <span>web performance</span> students!</p>
    <div></div>
    <script src="app.js"></script>
    <span>"app.js" cannot see me!</p>
  </body>
</html>

```



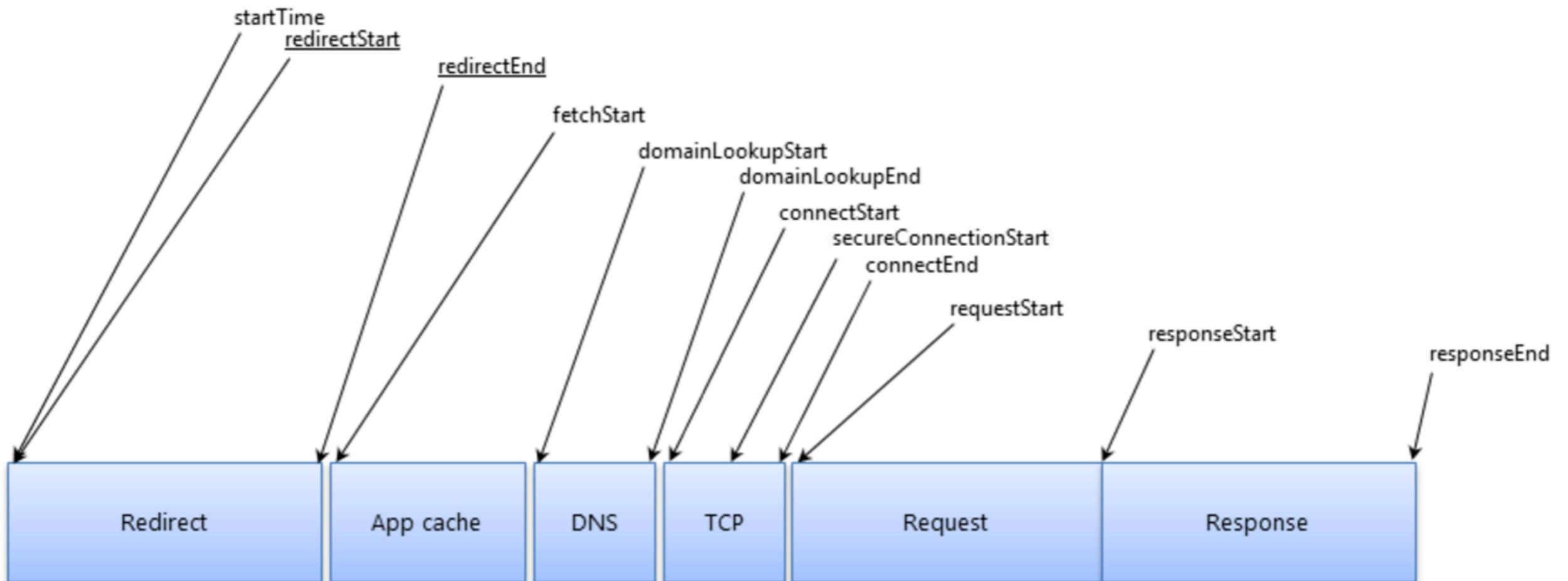
Nasıl Ölçebiliriz?

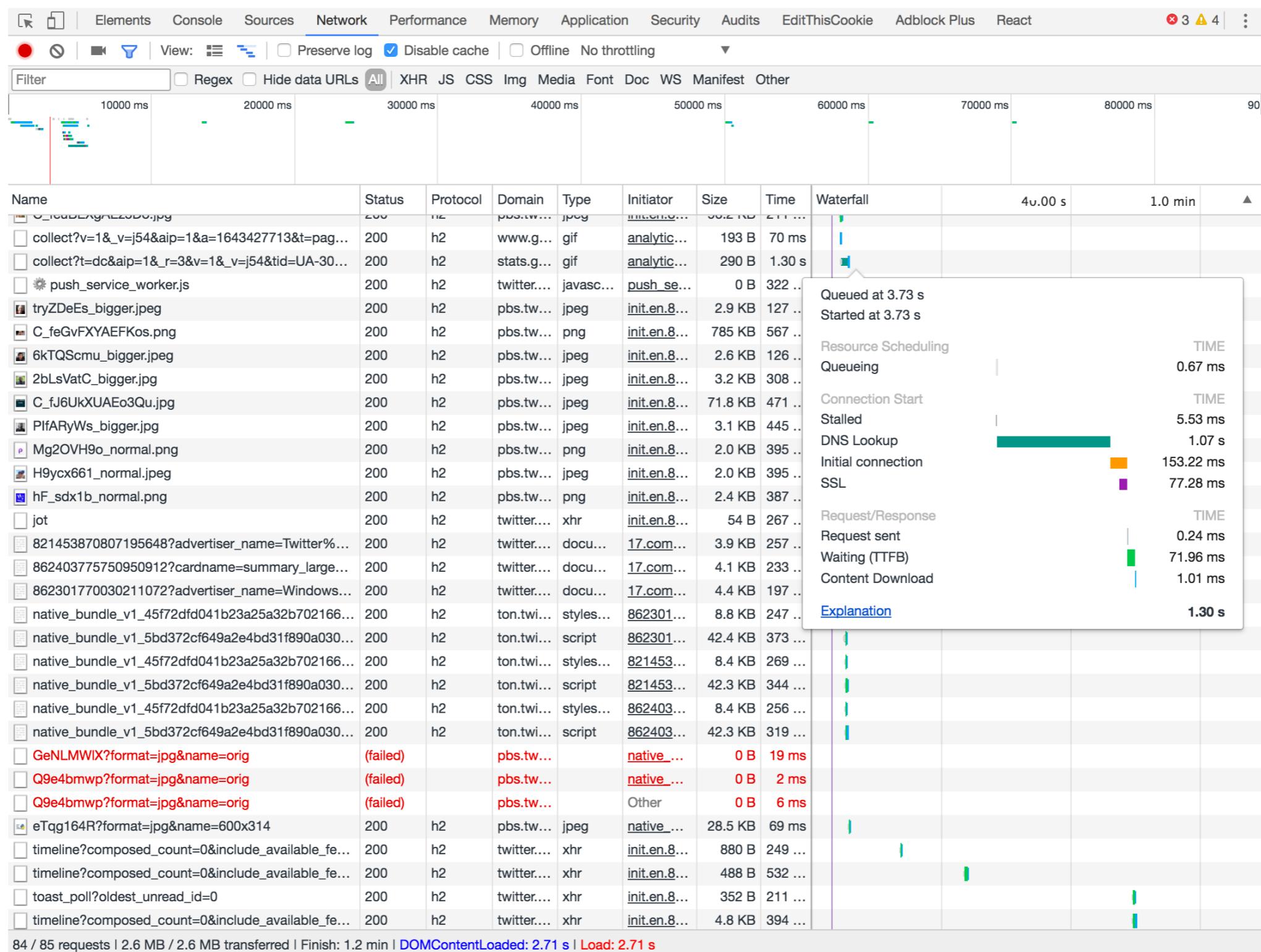


Resource Timing API

API:

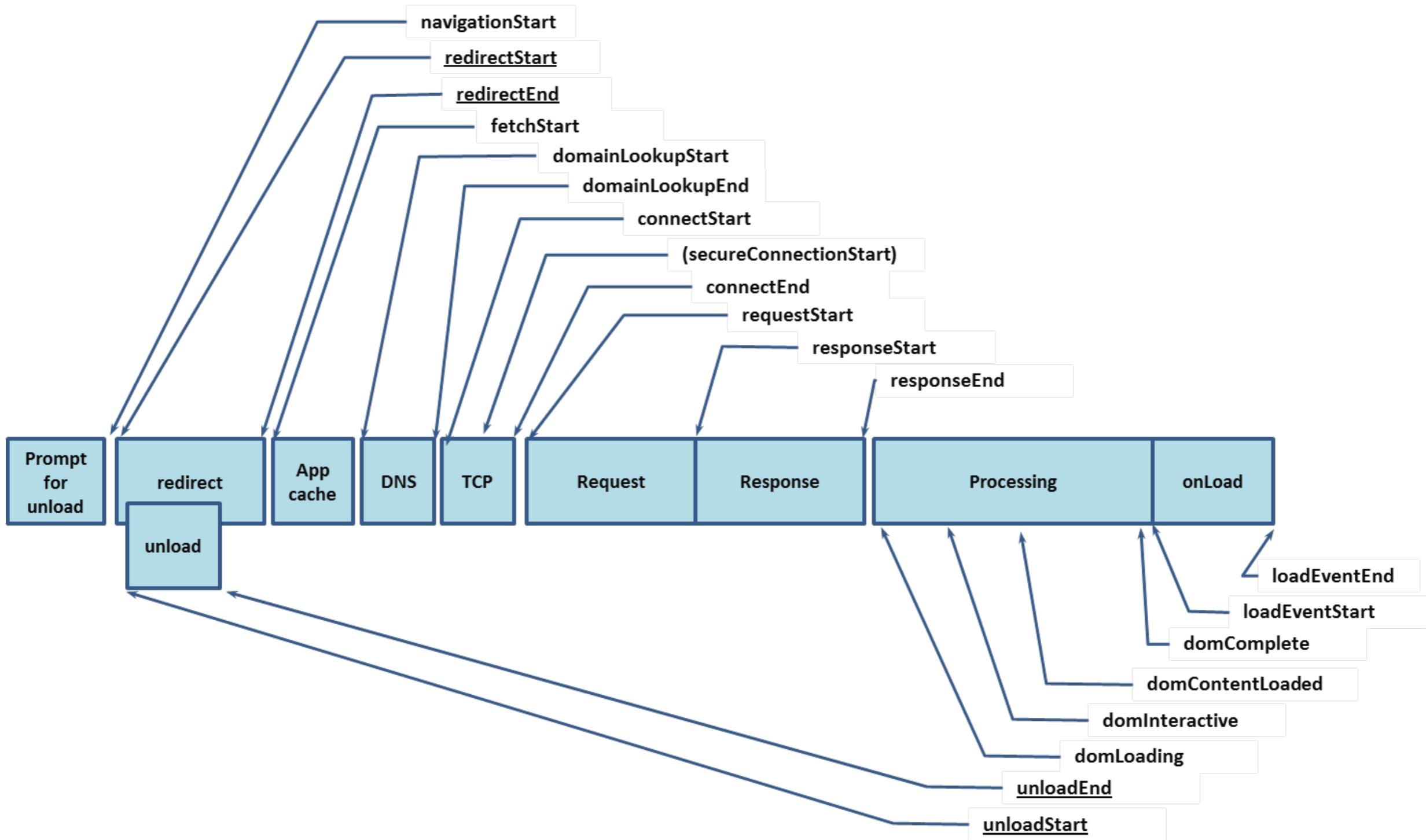
```
window.performance.getEntriesByType('resource')
```





API:

window.performance.timing



1

Search T Q

Elements Console VM1030:1

Filter Info

VM1030:1

The future we're building -- and boring

Elon Musk discusses his new project digging tunnels under LA, the latest from Tesla and SpaceX and his motivation for building a future on Mars in...

ted.com

3 56 144

New Relic Retweeted

New Relic @newrelic · Apr 26

7 Things You Need to #Monitor to Optimize your Media Performance
bddy.me/2pjcbuh @adwetzel #apm

Start Failure Funnel

Video Play Errors by Type

Buffer Events by Video

Video Buffering time

Video Plays by City

Bitrate by City

Bitrate by video

Average Bitrate by Browser

Bitrate by Browser

Video plays

User Timing API

API:

```
window.performance.mark('mark-1')  
window.performance.mark('mark-2')
```

```
window.performance.measure('measure-1', 'mark-1', 'mark-2')
```

```
window.performance.getEntriesByType('mark')  
window.performance.getEntriesByType('measure')
```

```
window.performance.clearMeasures()  
window.performance.clearMarks()
```

The screenshot shows the Chrome DevTools interface with the 'Console' tab selected. The console output displays JavaScript code used for measuring the duration of an XMLHttpRequest. The code includes marking start and end points using the `window.performance` API, sending the request, and then retrieving the measured entry.

```
> var xhr = new XMLHttpRequest();
  xhr.open('GET', 'https://www.html5rocks.com', true);
  xhr.onload = function(e) {
    window.performance.mark('mark_end_xhr');
    window.performance.measure('measure_xhr', 'mark_start_xhr', 'mark_end_xhr');
  }
  window.performance.mark('mark_start_xhr');
  xhr.send();
< undefined
> window.performance.getEntriesByType('measure');
< ▼ [PerformanceMeasure] ⓘ
  ▼ 0: PerformanceMeasure
    duration: 1074.42
    entryType: "measure"
    name: "measure_xhr"
    startTime: 6431.67
    ► __proto__: PerformanceMeasure
    length: 1
    ► __proto__: Array(0)
> |
```

Optimizasyon?

- Yüklenen verinin toplam boyutu
 - (HTML, CSS, JS, görseller, vs.)
- Sunucuya yapılan istek sayısı
 - Eşzamanlı maksimum 6 istek (aynı kaynak için)
- Ağ bağlantı hızı
- “Critical Rendering Path”

- Font ve görsellerin optimizasyonu
 - SVG kullanımı, metadata temizleme, sıkıştırma
- Daha az kod
 - Gereksiz framework/kütüphane kullanımı (ör: jQuery)
 - Transpiling? (Babel)
 - Statik kod optimizasyonu (minify, tree shaking, prepack)
- Bundling ve code-splitting
 - Webpack, rollup, browserify, vs.

Evaluate

Presets: es2015, react, stage-0 ▾

 Line Wrap Minify (Babili)

Babel 6.24.0

```
1  async function foo() {
2    await bar();
3  }
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
```

```
2
3  var foo = function () {
4    var _ref = _asyncToGenerator(regeneratorRuntime.mark(function _callee() {
5      return regeneratorRuntime.wrap(function _callee$(_context) {
6        while (1) {
7          switch (_context.prev = _context.next) {
8            case 0:
9              _context.next = 2;
10             return bar();
11
12            case 2:
13            case "end":
14              return _context.stop();
15            }
16          }
17        }, _callee, this);
18      }));
19
20      return function foo() {
21        return _ref.apply(this, arguments);
22      };
23    }();
24
25  function _asyncToGenerator(fn) { return function () { var gen = fn.apply(this, arguments); return new Promise(function (resolve, reject) { function step(key, arg) { try { var info = gen[key](arg); var value = info.value; } catch (error) { reject(error); return; } if (info.done) { resolve(value); } else { return Promise.resolve(value).then(function (value) { step("next", value); }, function (err) { step("throw", err); }); } } return step("next"); }; };
```

```

class Mangler {
  constructor(program) {
    this.program = program;
  }
}
new Mangler();

```

Minify
→

```
class a{constructor(b){this.program=b}}new a;
```

React.js

Input (raw): 124.7 KB
 Input (gzip): 29.8 KB

minifier	output	raw win	gzip output	gzip win
babili	36.13kB	%71	12.6kB	%58
uglify	35.73kB	%71	11.96kB	%60
closureCompiler	34.77kB	%72	11.96kB	%60
closureCompilerJs	65.41kB	%48	15.83kB	%47

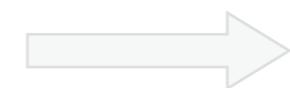
main.js (entry module)

```
1 import { cube } from './maths.js';
2 console.log( cube( 5 ) ); // 125
```

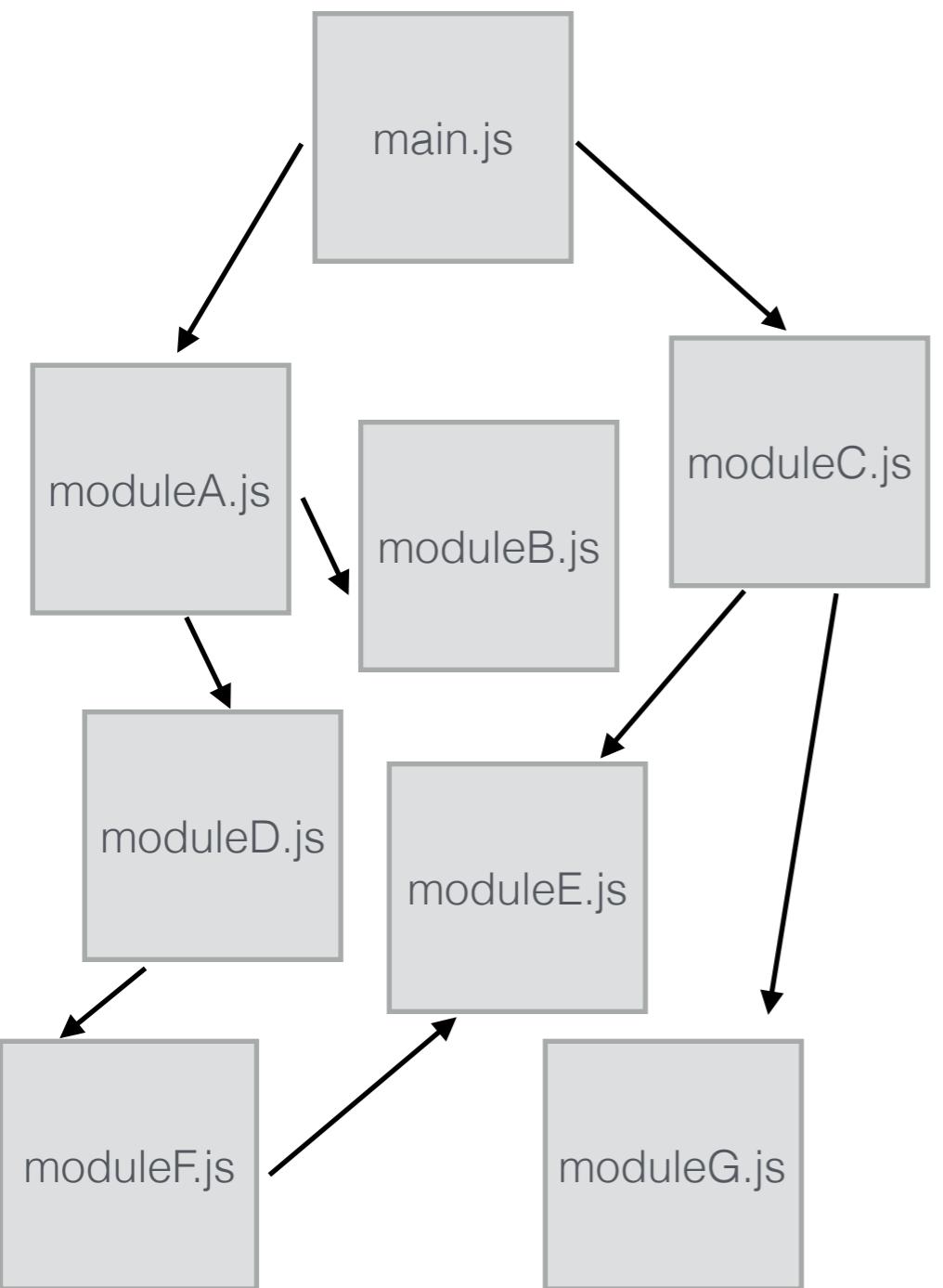
maths.js

```
1 // This function isn't used anywhere, so
2 // Rollup excludes it from the bundle...
3 export function square ( x ) {
4     return x * x;
5 }
6
7 // This function gets included
8 export function cube ( x ) {
9     // rewrite this as `square( x ) * x`
10    // and see what happens!
11    return x * x * x;
12 }
```

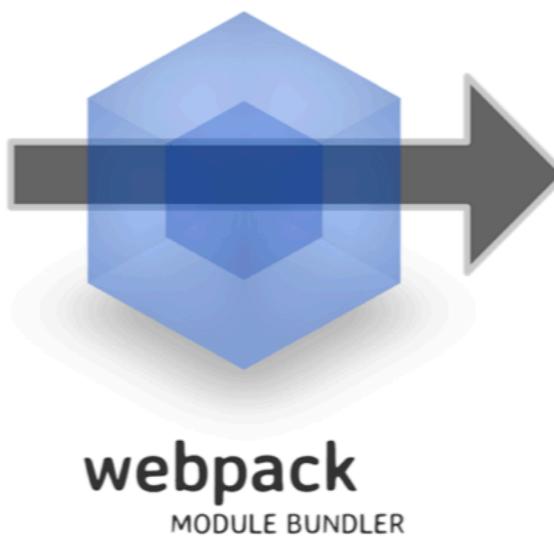
Tree-shaking



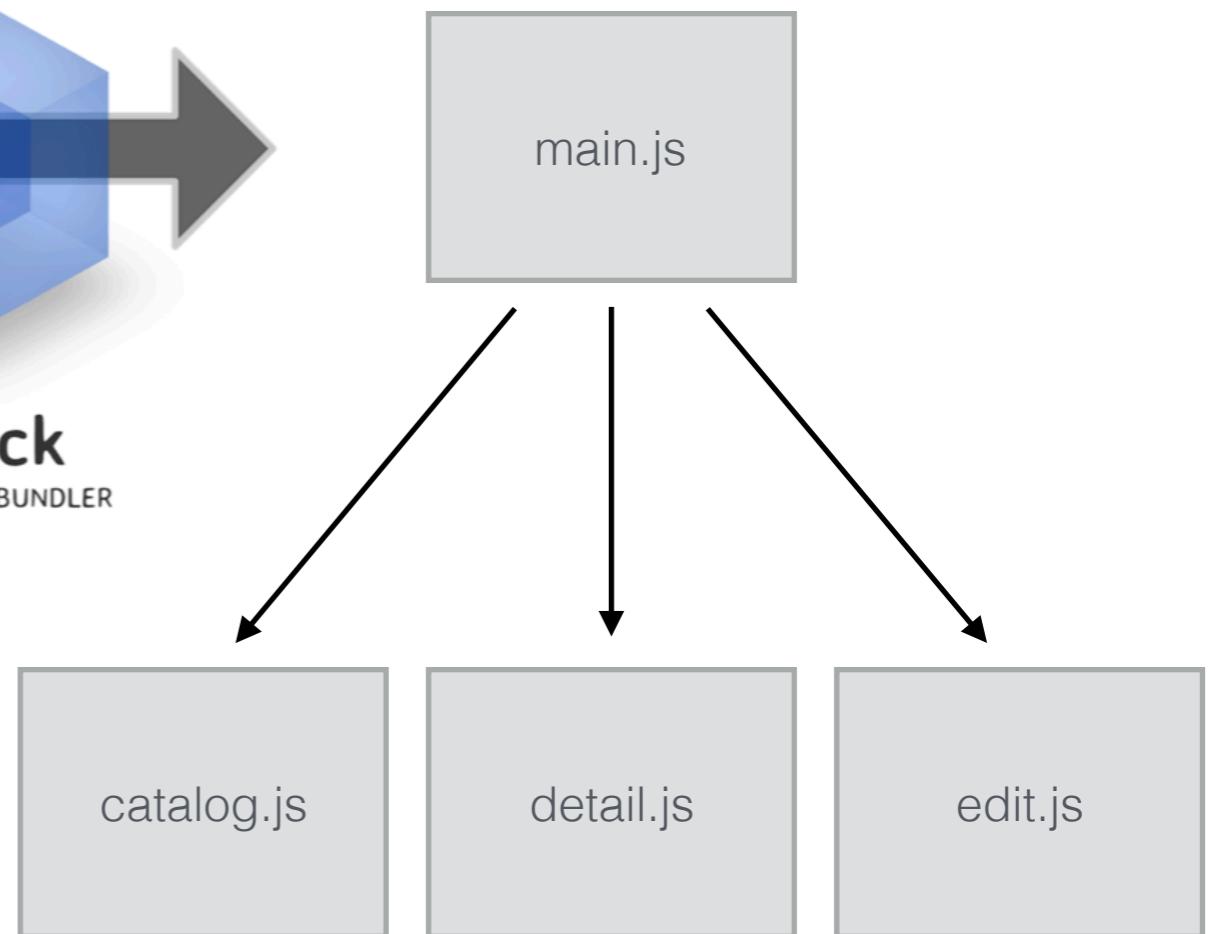
```
1 'use strict';
2
3 // This function isn't used anywhere, so
4 // Rollup excludes it from the bundle...
5
6
7 // This function gets included
8 function cube ( x ) {
9     // rewrite this as `square( x ) * x`
10    // and see what happens!
11    return x * x * x;
12 }
13
14 console.log( cube( 5 ) ); // 125
15
```



Developer Experience



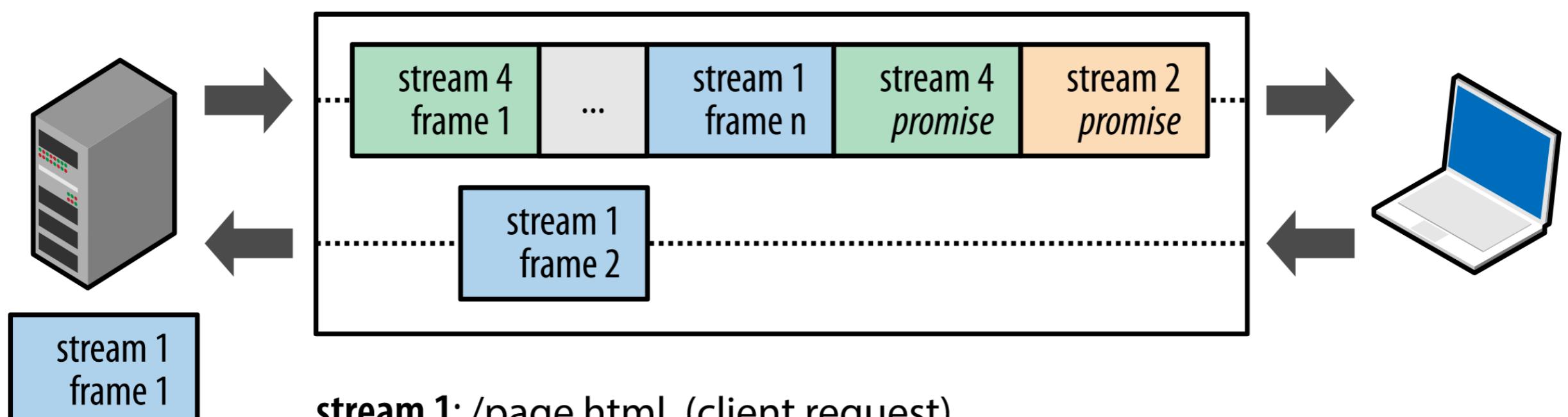
webpack
MODULE BUNDLER

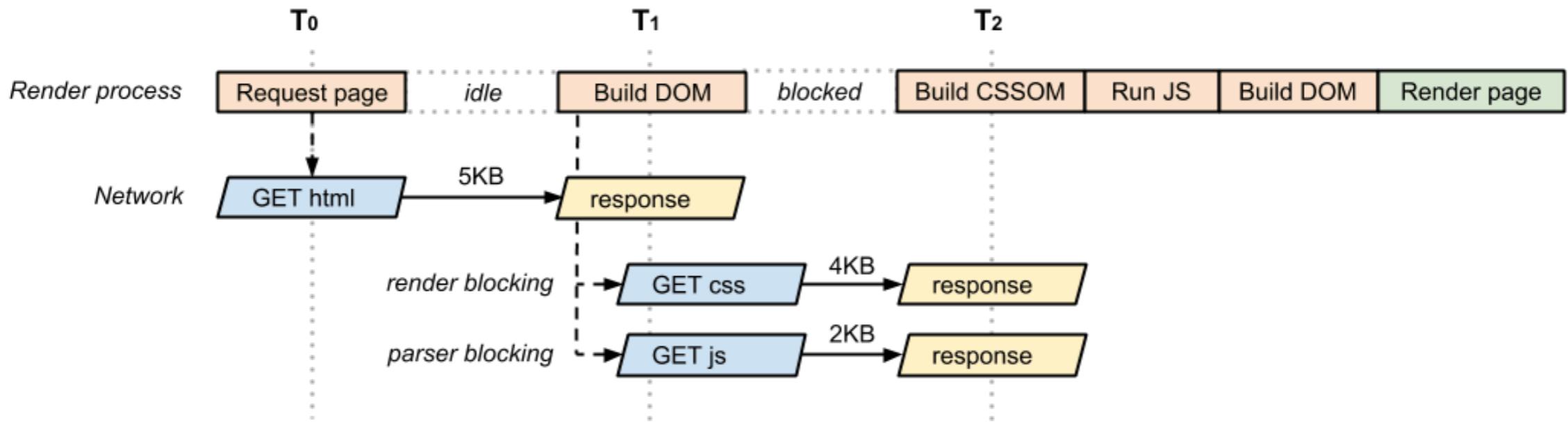


User Experience

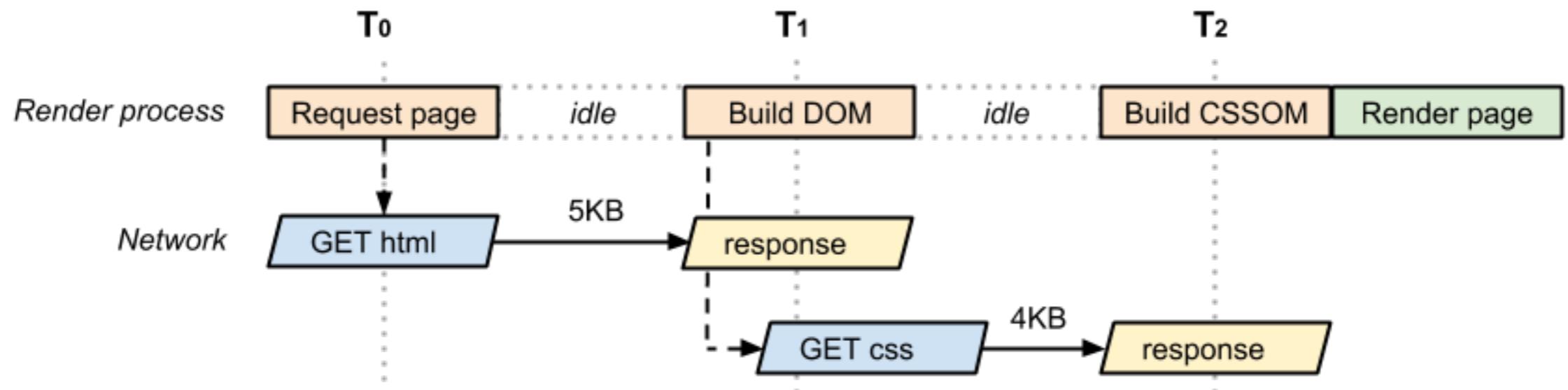
- GZIP ile sıkıştırma
- HTTP caching
 - script.js -> script-a34fed.js
- CDN kullanımı
- HTTP/2 protokolü
- “Server-side rendering”

HTTP 2.0 connection





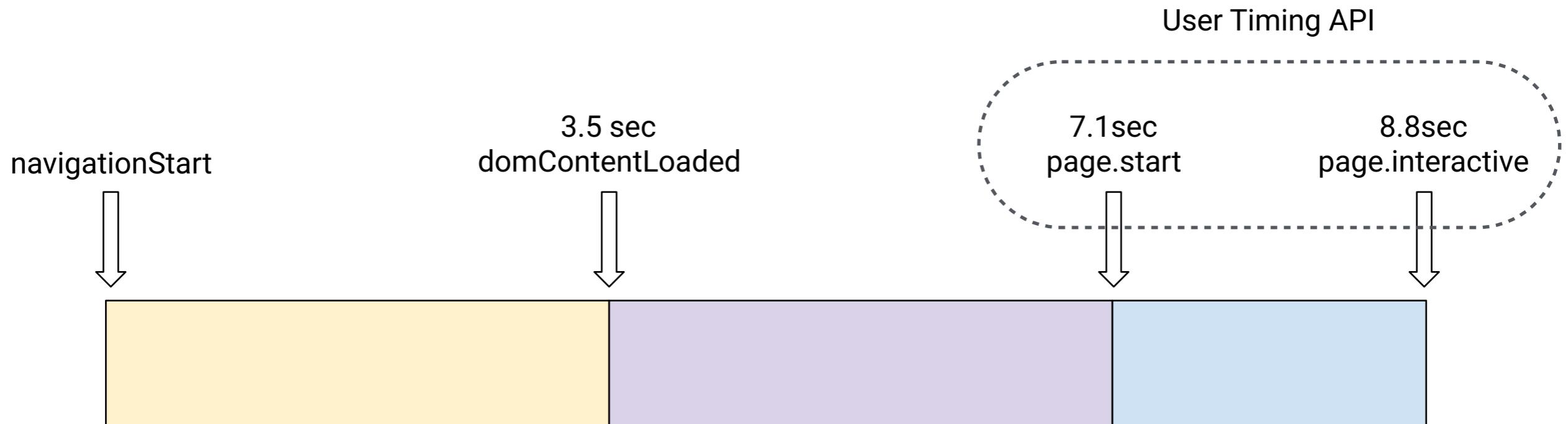
Server-side rendering



Tarayıcı optimizasyonu

```
<link rel="dns-prefetch" href="//hostname_to_resolve.com">
<link rel="subresource" href="/javascript/script.js">
<link rel="prefetch" href="/images/big.jpeg">
<link rel="prerender" href="//example.org/next_page.html">
```

Zeplin Webapp (optimizasyon öncesi)



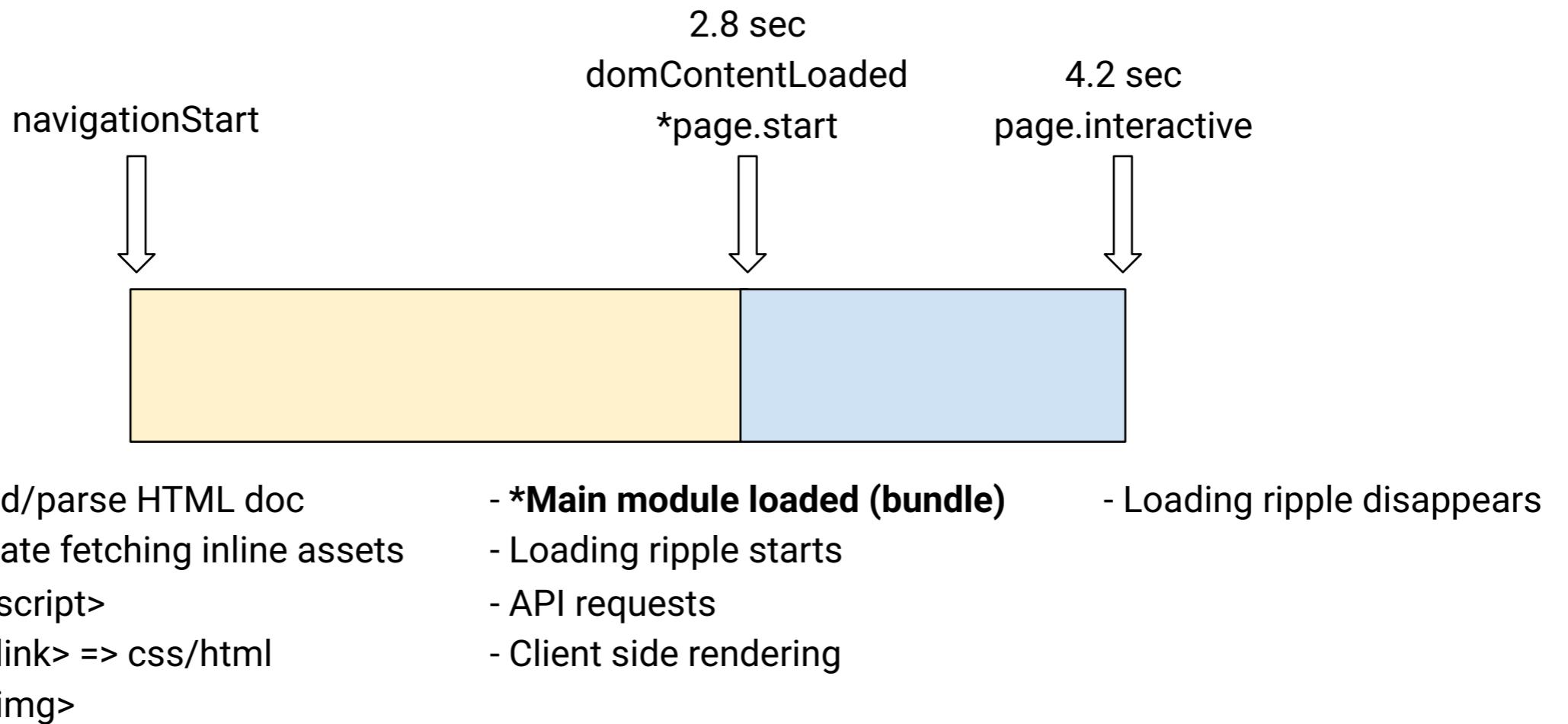
- Load/parse HTML
- Fetch inline assets
 - <script>
 - <link> => css/html
 -
- ***requirejs**

- Requirejs loads main module and its dependencies
- (30+ javascript modules)**

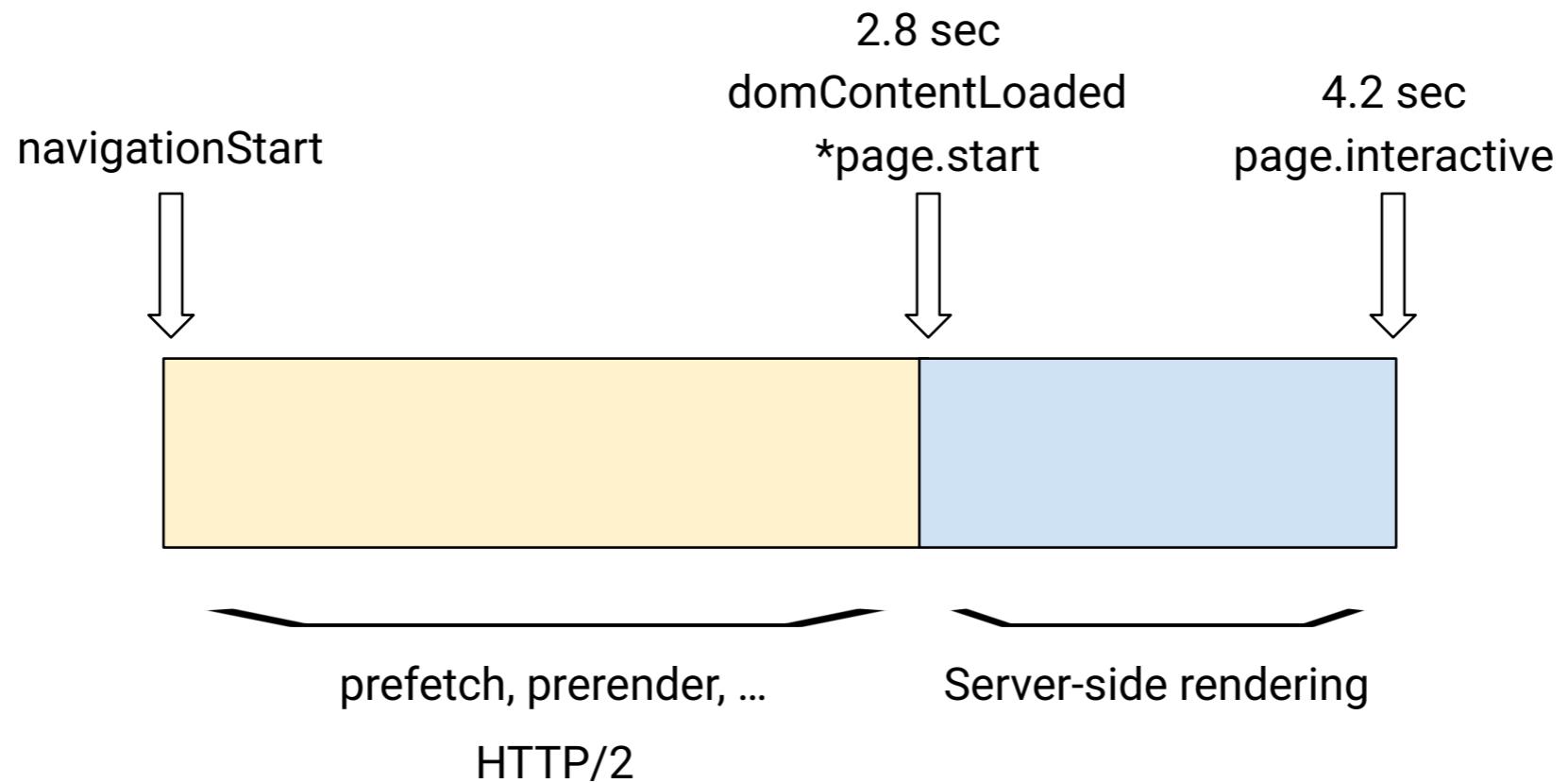
- Loading spinner starts
- API requests
- Client side rendering

- Loading spinner disappears

Zeplin Webapp (optimizasyon sonrası)



Zeplin Webapp (optimizasyon sonrası)



Teşekkürler!

sertac@zeplin.io