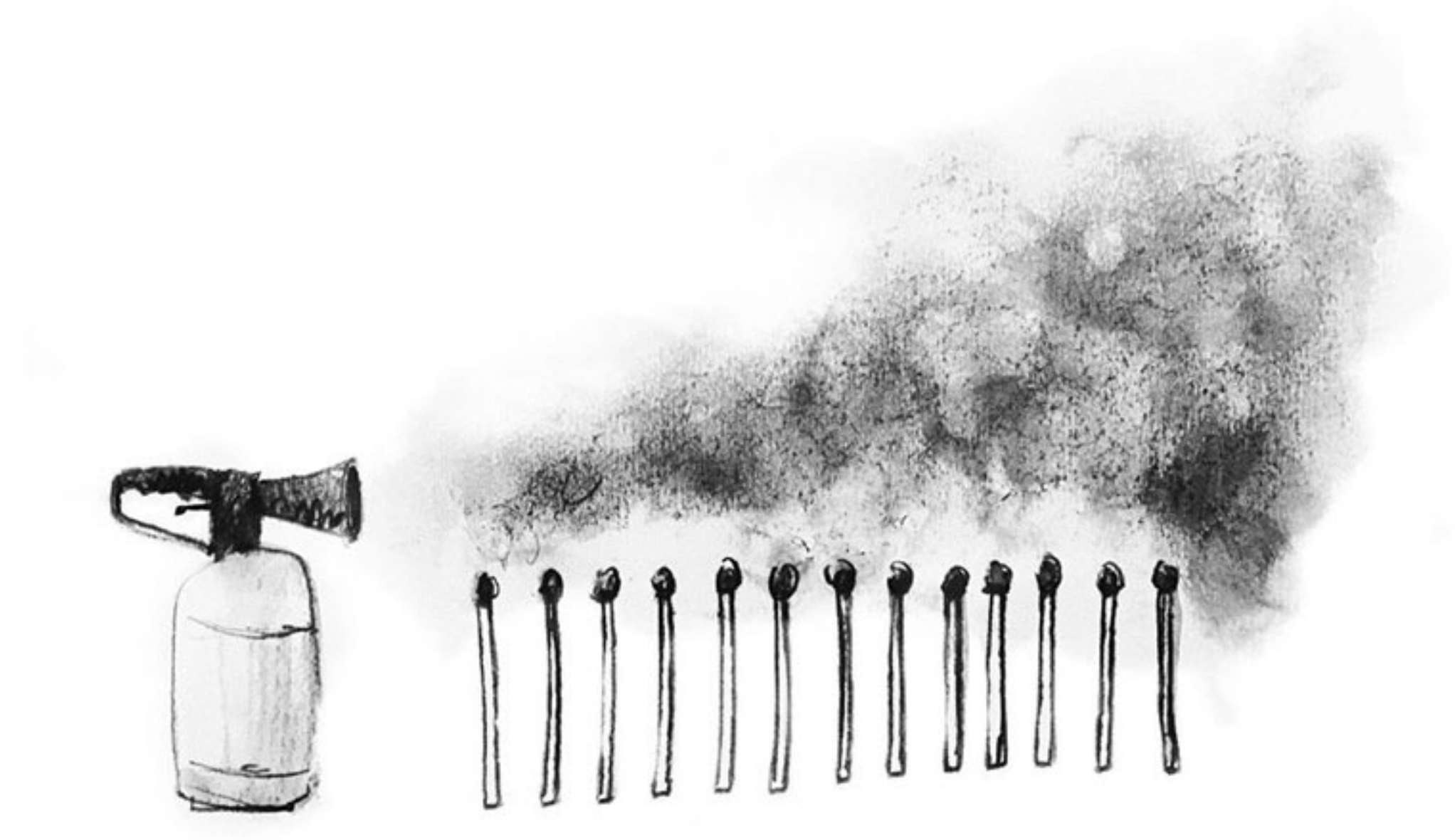


RESPONSIBLE JAVASCRIPT

JEREMY WAGNER — @MALCHATA — JEREMY.CODES

WE LOVE SPEED — LILLE, FRANCE — SEPTEMBER 2019

A LIST APART



Responsible JavaScript: Part I

by [Jeremy Wagner](#) · March 28, 2019

Published in [Application Development](#), [JavaScript](#)

By the numbers, [JavaScript is a performance liability](#). If the trend persists, the median page will be shipping at least 400 KB of it before too long, and that's merely what's *transferred*. Like other text-based resources, JavaScript is almost always served compressed—but that might be the only thing we're getting consistently right in its delivery.

Unfortunately, while reducing resource transfer time is a big part of that whole performance thing, compression has no effect on how long browsers take to process a script once it arrives in its entirety. If a server sends 400 KB of compressed JavaScript, the actual amount browsers have to process after decompression is north of a megabyte. How well devices cope with these heavy workloads depends, well, on the *device*. [Much has been written](#) about how adept various devices are at processing lots of JavaScript, but the truth is, the amount of time it takes to process even a trivial amount of it varies greatly between devices.

Share this:



Become a patron

Translations

SPHEXISHNESS

SPHEXISH

(of animal behavior) deterministic; preprogrammed

SPHEX

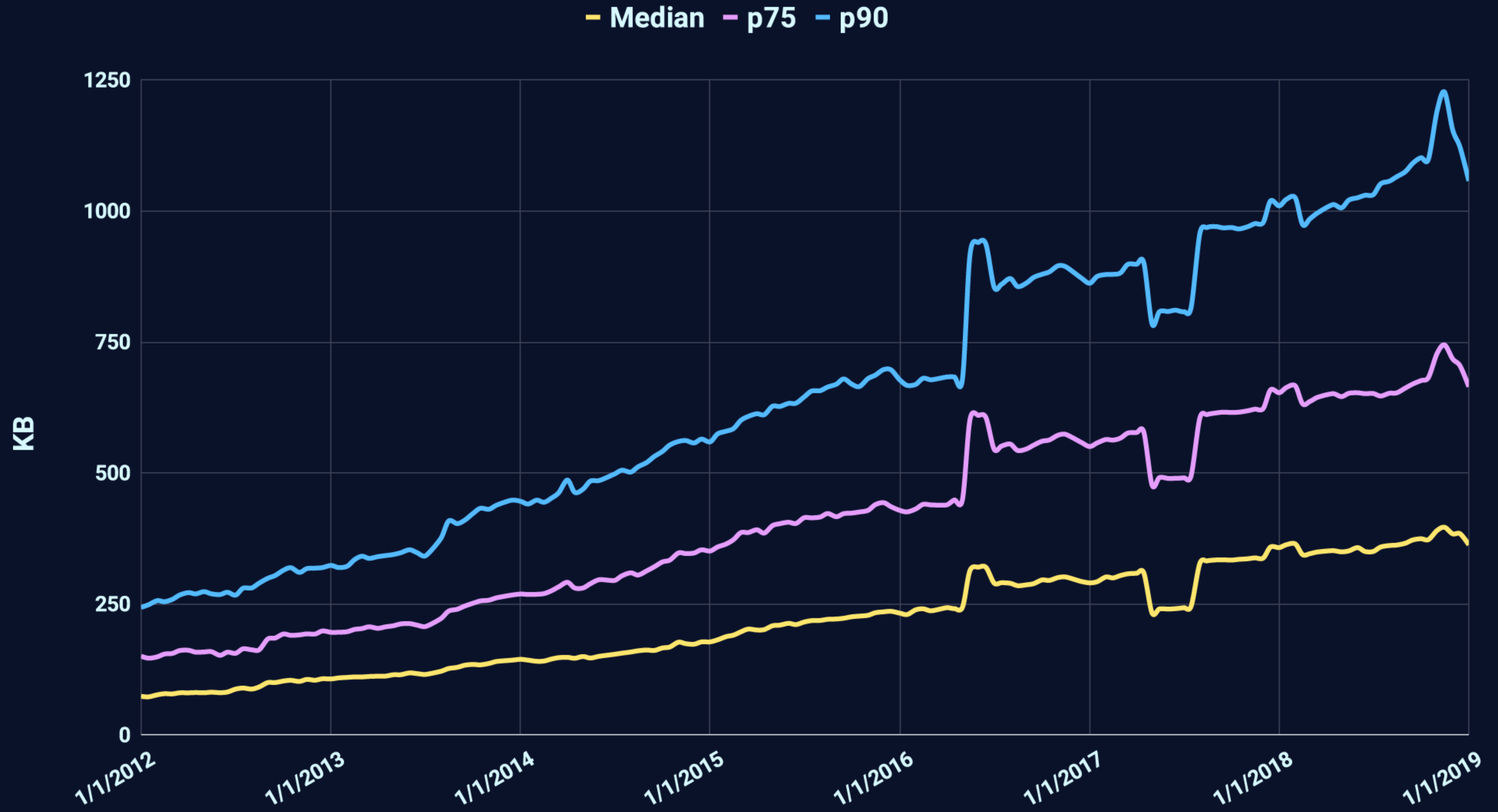
Sphex pensylvanicus





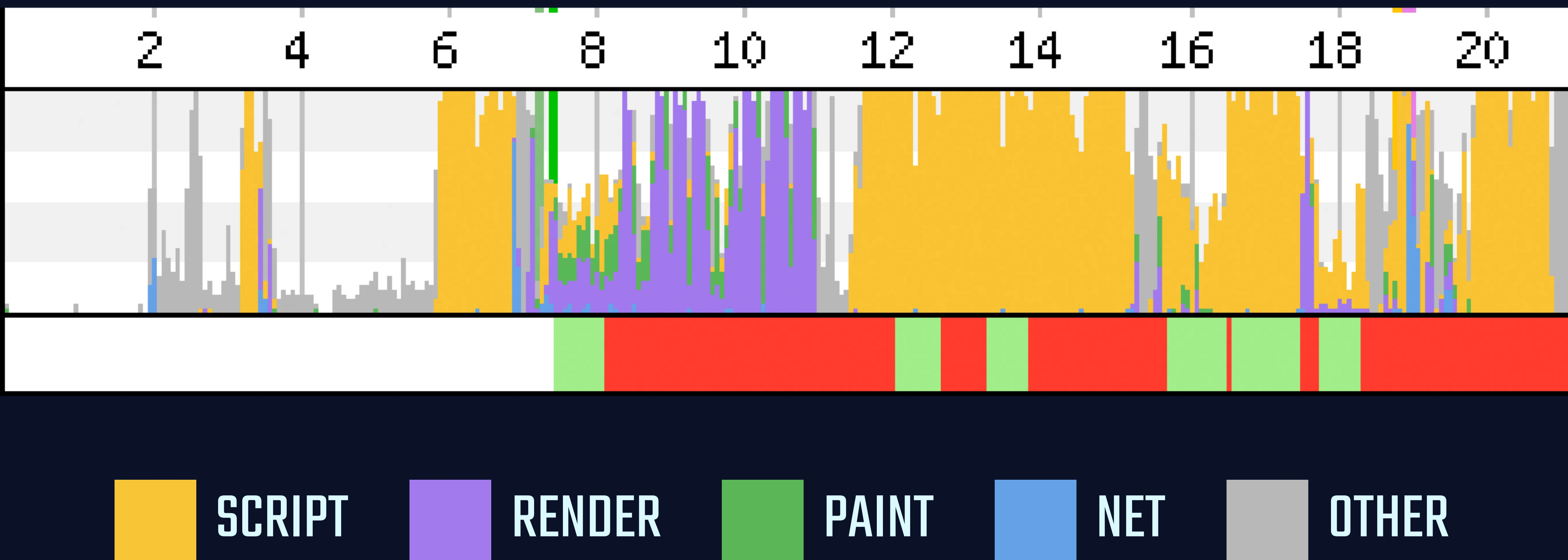
<code>npm install react</code>	<i>6.5 KB</i>
<code>npm install react-dom</code>	<i>103.7 KB</i>
<code>npm install react-router</code>	<i>21.6 KB</i>
<code>npm install react-redux</code>	<i>14.4 KB</i>

JavaScript Delivered on Mobile Devices











ANTI-SPHEXISHNESS

PAINT THE PICTURE

NOT THE FRAME

A LIST APART

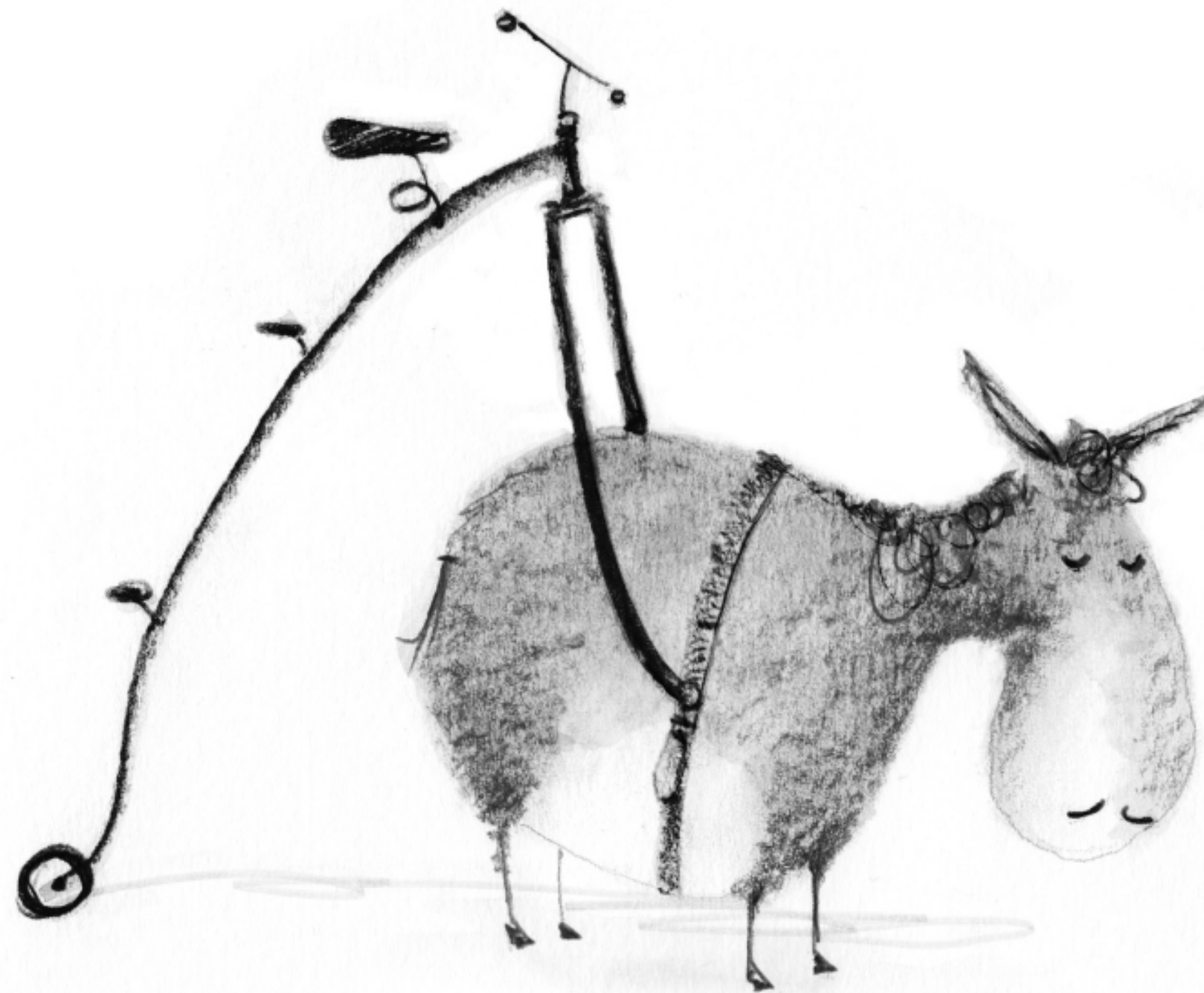


Illustration by [Dougal MacPherson](#)

Paint the Picture, Not the Frame: How Browsers Provide Everything Users Need

by [Eric Bailey](#) · February 07, 2019

Published in [Accessibility](#), [User Experience](#)

Kip Williams, professor of psychology sciences at Purdue University, conducted [a fascinating experiment](#) called “cyberball.” In his experiment, a test subject and two other participants played a computer game of catch. At a predetermined time, the test subject was excluded from the game, forcing them to only observe as the clock ran down.

However, the stakes for a quality design were pretty high: scrollbars are part of an application's interface, not a website's. In inclusive design, it's part of what we call *external consistency*.

External consistency is the idea that an object's functionality is informed and reinforced by similar implementations elsewhere. It's why you can flip a wall switch in most houses and be guaranteed the lights come on instead of flushing the toilet.


```
import React, { Component } from "react";
import { validateEmail } from "helpers/validation";

class SignupForm extends Component {
  constructor (props) {
    super(props);

    this.handleSubmit = this.handleSubmit.bind(this);
    this.updateEmail = this.updateEmail.bind(this);
    this.state.email = "";
  }

  updateEmail (event) {
    this.setState({
      email: event.target.value
    });
  }

  handleSubmit () {
    // If the email checks out, submit
    if (validateEmail(this.state.email)) {
      // ...
    }
  }

  render () {
    return (
      <div>
        <span class="email-label">Enter your email:</span>
        <input type="text" id="email" onChange={this.updateEmail} />
        <div class="submit-button" onClick={this.handleSubmit}>Sign Up</div>
      </div>
    );
  }
}
```

```
render () {  
  return (  
    <div>  
      <span class="email-label">Enter your email:</span>  
      <input type="text" id="email" onChange={this.updateEmail} />  
      <div class="submit-button" onClick={this.handleSubmit}>Sign Up</div>  
    </div>  
  );  
}
```

```
render () {  
  return (  
    <form method="POST" action="/signup" onSubmit={handleSubmit}>  
      <label for="email" class="email-label">Enter your email:</label>  
      <input type="email" id="email" required />  
      <button>Sign Up</button>  
    </form>  
  );  
}
```

```
import React from "react";

const SignupForm = function (props) {
  const handleSubmit = function (event) {
    // Needed in case we're sending data to the server XHR-style
    // (but will still work if server-rendered with JS disabled).
    event.preventDefault();

    // Carry on...
  };

  return (
    <form method="POST" action="/signup" onSubmit={handleSubmit}>
      <label for="email" class="email-label">Enter your email:</label>
      <input type="email" id="email" required />
      <button>Sign Up</button>
    </form>
  );
};
```




Der 19. Psalm.
Gerechtigkeit Gottes in der Natur und in seinem
Wort. Bitte um Sündenvergebung und um
Bewahrung vor Anfecht.
1. Ein Psalm Davids, vorausfingen.
2. Die Himmel erzählen die Ehre

Gebet des Psalms.
1. Ein Psalm.
2. Der Herr ist
dich!
3. Er sende

und Rache dich.
1. Ein Psalm.
2. Der Herr ist
dich!
3. Er sende
und Rache dich.
1. Ein Psalm.
2. Der Herr ist
dich!
3. Er sende
und Rache dich.

und Rache dich.
1. Ein Psalm.
2. Der Herr ist
dich!
3. Er sende
und Rache dich.
1. Ein Psalm.
2. Der Herr ist
dich!
3. Er sende
und Rache dich.

und Rache dich.
1. Ein Psalm.
2. Der Herr ist
dich!
3. Er sende
und Rache dich.
1. Ein Psalm.
2. Der Herr ist
dich!
3. Er sende
und Rache dich.

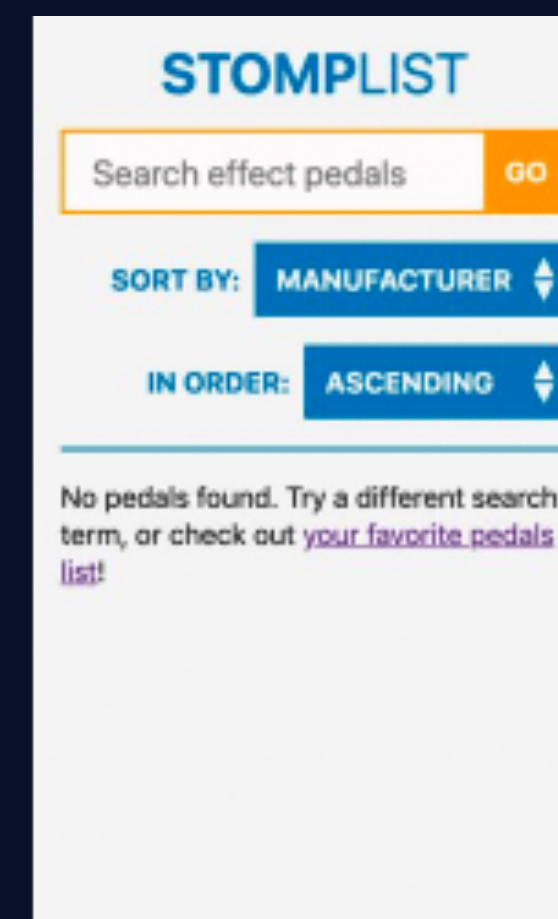
1 MS



2.07 S

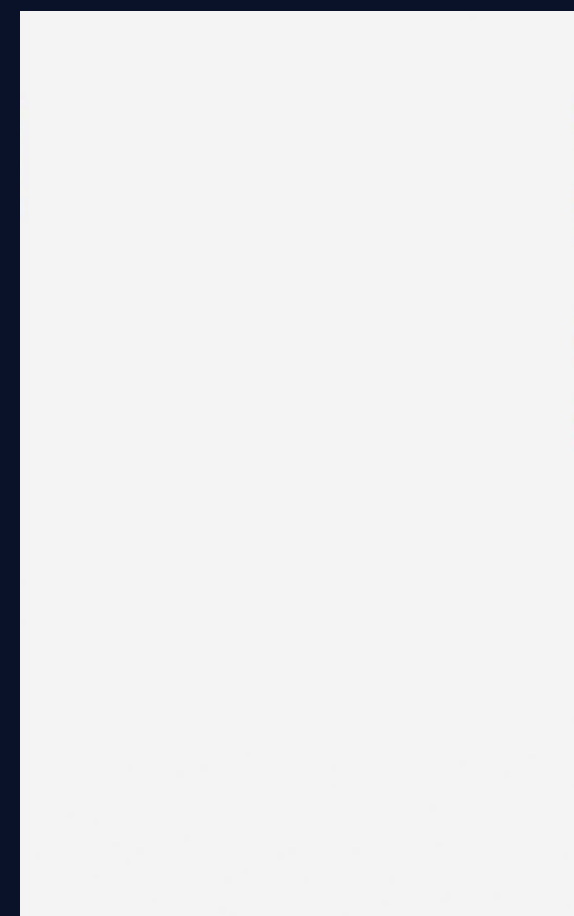


5.24 S

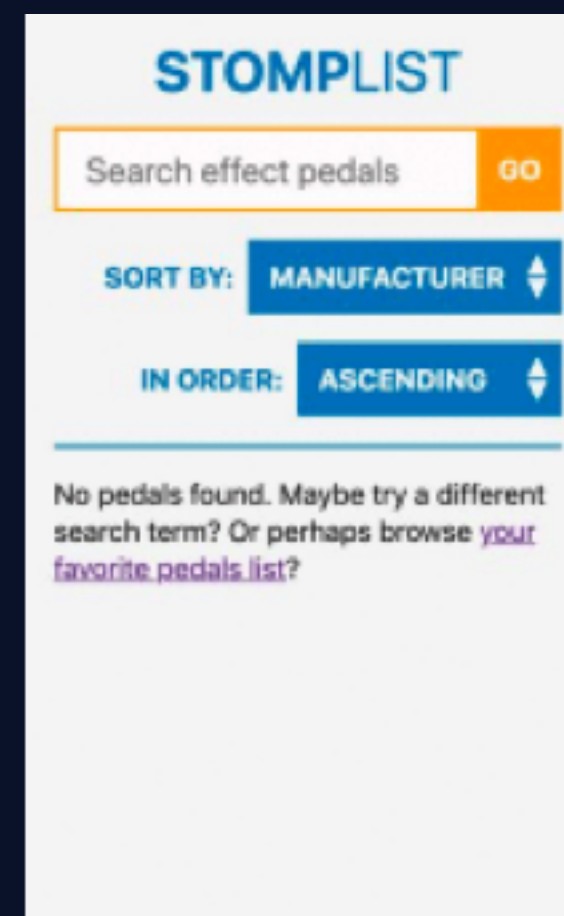


CLIENT-SIDE RENDERING

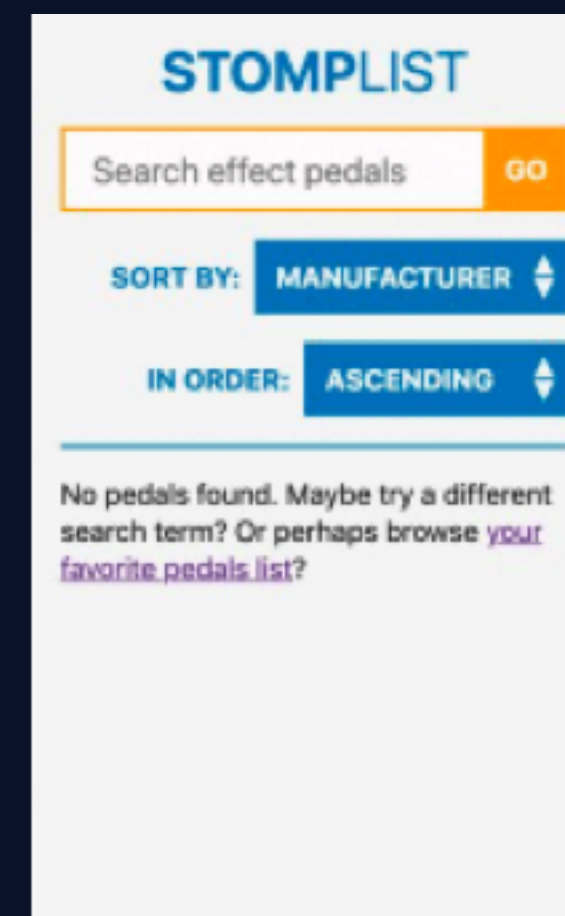
1 MS



2.07 S



5.24 S



SERVER-SIDE RENDERING
(WITH CLIENT-SIDE HYDRATION)

```
<link rel="prefetch" href="/products/snes-console">
```



npm v1.0.0 gzip size 782 B downloads 37k build passing

quicklink

Faster subsequent page-loads by prefetching in-viewport links during idle time

How it works

Quicklink attempts to make navigations to subsequent pages load faster. It:

- **Detects links within the viewport** (using [Intersection Observer](#))
- **Waits until the browser is idle** (using [requestIdleCallback](#))
- **Checks if the user isn't on a slow connection** (using `navigator.connection.effectiveType`) or has data-saver enabled (using `navigator.connection.saveData`)
- **Prefetches URLs to the links** (using `<link rel=prefetch>` or XHR). Provides some control over the request priority (can switch to `fetch()` if supported).

BROWSERS GIVE US
A LOT OF FREE STUFF

THE TOOLS ARE
NOT INFALLIBLE



PHOTO CREDIT: JOHN HOEY


```
// Untransformed code:  
function logger(message, level = "log") {  
    console[level](message);  
}
```

```
// Babel-transformed code:  
function logger(message) {  
  var level = arguments.length > 1 && arguments[1] !== undefined ? arguments[1] : "log";  
  console[level](message);  
}
```

```
// Code that Babel won't touch:
function logger(message, level) {
  console[level || "log"](message);
}           // ^^^^^^^^^^^^^^^^^
```

```
export class User {  
  constructor (id, name, email) {  
    this.id = id;  
    this.name = name;  
    this.email = email;  
  }  
  
  getId () {  
    return this.id;  
  }  
  
  getName () {  
    return this.name;  
  }  
  
  getEmail () {  
    return this.email;  
  }  
}
```



```
"use strict";

Object.defineProperty(exports, "__esModule", {
  value: true
});
exports.User = void 0;

function _classCallCheck(instance, Constructor) { if (!(instance instanceof Constructor)) { throw new TypeError("Cannot call a class as a function"); } }

function _defineProperties(target, props) { for (var i = 0; i < props.length; i++) { var descriptor = props[i]; descriptor.enumerable = descriptor.enumerable || false; descriptor.configurable = true; if ("value" in descriptor) descriptor.writable = true; Object.defineProperty(target, descriptor.key, descriptor); } }

function _createClass(Constructor, protoProps, staticProps) { if (protoProps) _defineProperties(Constructor.prototype, protoProps); if (staticProps) _defineProperties(Constructor, staticProps); return Constructor; }

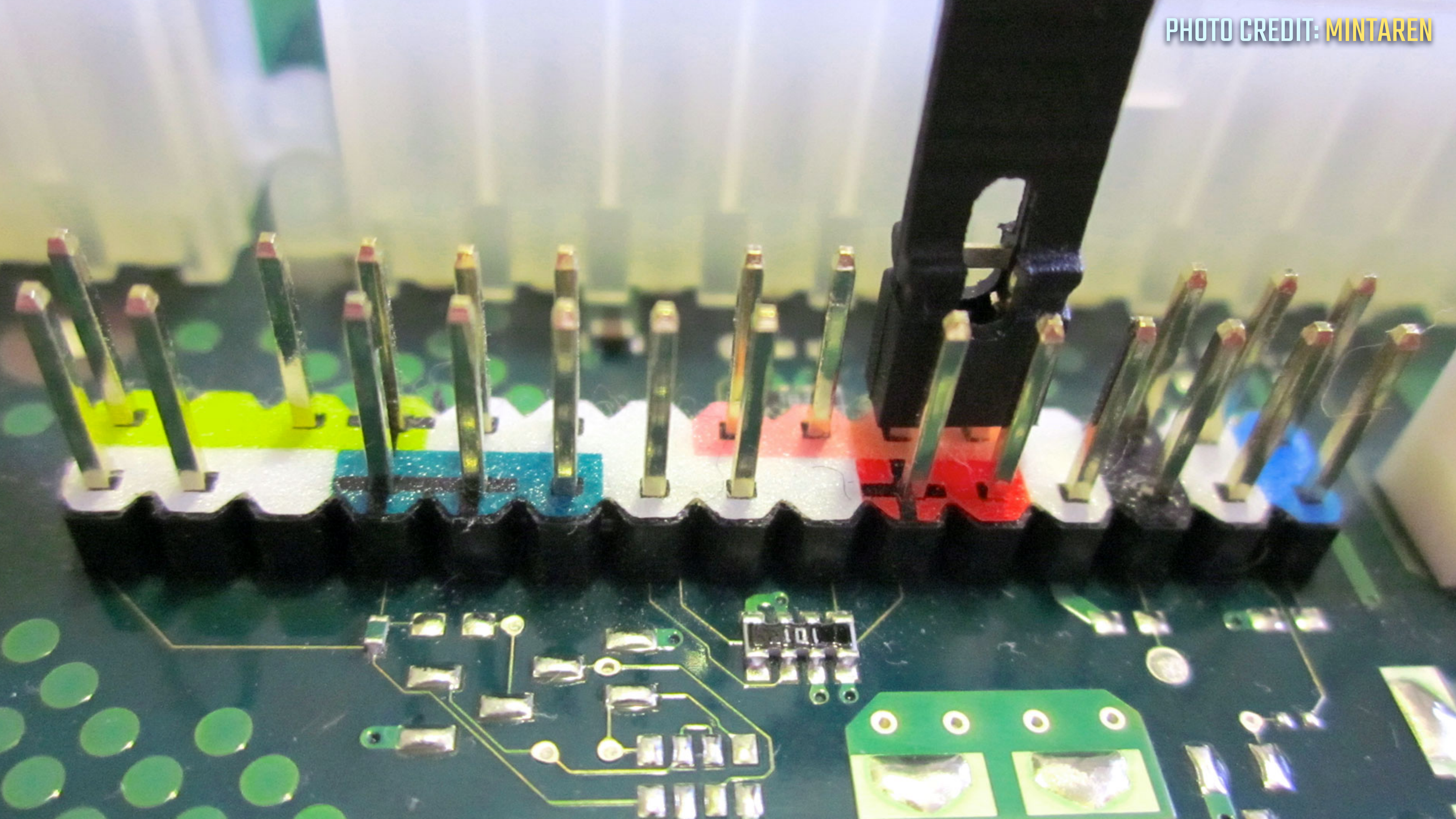
var User =
/*#__PURE__*/
function () {
  function User(id, name, email) {
    _classCallCheck(this, User);

    this.id = id;
    this.name = name;
    this.email = email;
  }

  _createClass(User, [{
    key: "getId",
    value: function getId() {
      return this.id;
    }
  }, {
    key: "getName",
    value: function getName() {
      return this.name;
    }
  }, {
    key: "getEmail",
    value: function getEmail() {
      return this.email;
    }
  }]);

  return User;
}();

exports.User = User;
```

Treemap sizes:

Stat

Parsed

Gzipped

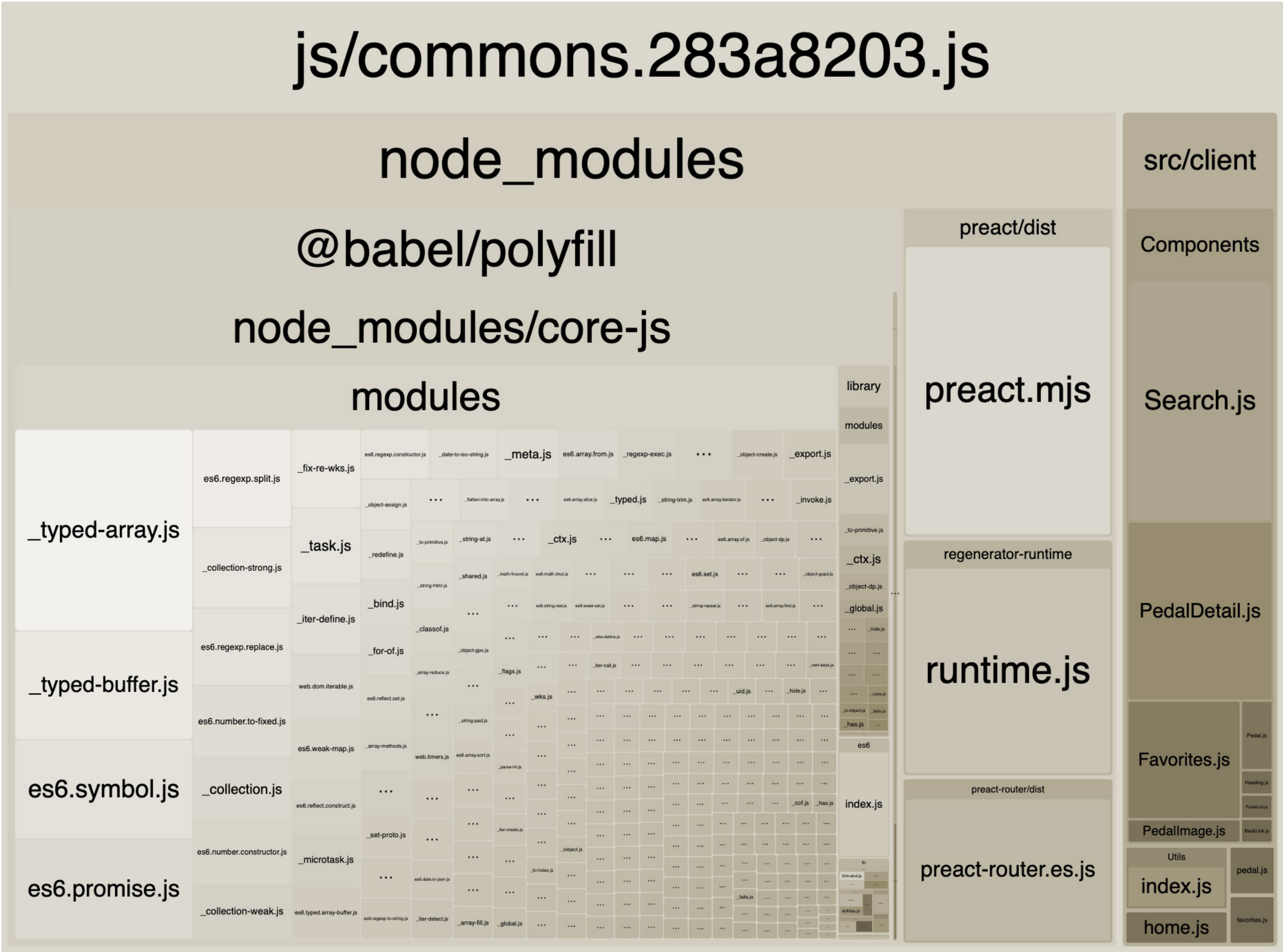
Search modules:

Enter regexp

X

Show chunks:

- ☒ All (116.59 KB)
- ☒ js/commons.283a8203.js (112.17 KB)
- ☒ js/favorites.a8b9ec0f.js (1.48 KB)
- ☒ js/pedal.ac9b429d.js (1.47 KB)
- ☒ js/home.12464d05.js (1.47 KB)



TOTAL BUNDLE SIZE: ~117 KB


```
presets: [  
  [  
    "@babel/preset-env", {  
      modules: false,  
      useBuiltIns: "entry",  
      corejs: 3,  
      targets: "> 0.25%, IE > 10, Firefox ESR, not dead"  
    }  
  ]  
]
```

```
presets: [  
  [  
    "@babel/preset-env", {  
      modules: false,  
      useBuiltIns: "usage",  
      corejs: 3,  
      targets: "> 0.25%, IE > 10, Firefox ESR, not dead"  
    }  
  ]  
]
```



```
presets: [  
  [  
    "@babel/preset-env", {  
      modules: false,  
      useBuiltIns: "usage",  
      loose: true,  
      corejs: 3,  
      targets: "> 0.25%, IE > 10, Firefox ESR, not dead"  
    }  
  ]  
]
```

Treemap sizes:

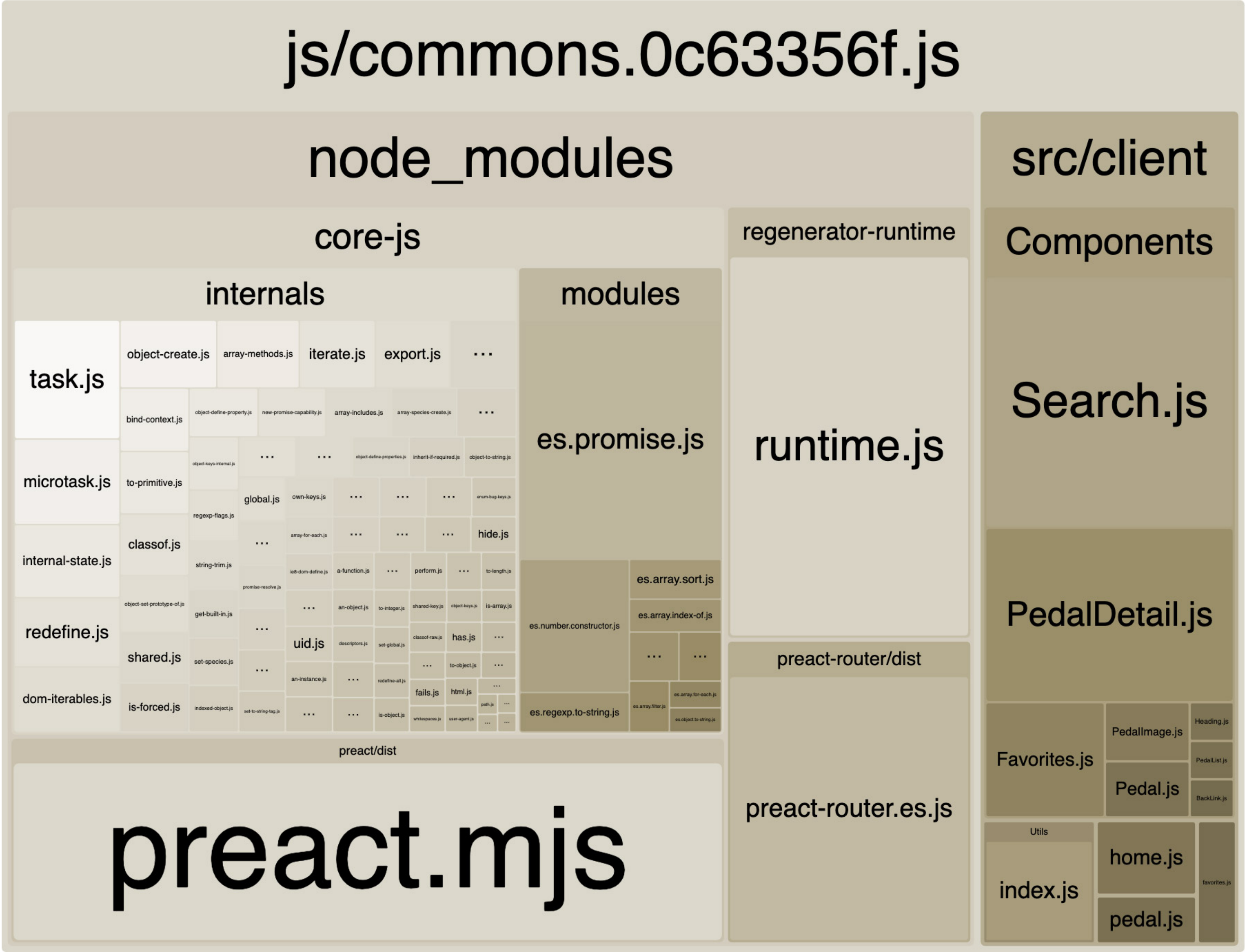
Stat **Parsed** Gzipped

Search modules:

Enter regexp

Show chunks:

- ☒ All (56.09 KB)
- ☒ js/commons.0c63356f.js (51.67 KB)
- ☒ js/favorites.6eb32550.js (1.48 KB)
- ☒ js/pedal.7c773d3d.js (1.47 KB)
- ☒ js/home.5331f7ef.js (1.47 KB)



TOTAL BUNDLE SIZE: ~56.09 KB

DIFFERENTIAL SERVING


```
<!-- The way we've always done it: -->  
<script defer src="/js/app.js"></script>
```

```
<!-- Modern browsers get this: -->  
<script type="module" src="/js/app.mjs"></script>  
<!-- Legacy browsers get this: -->  
<script nomodule defer src="/js/app.js"></script>
```



```
// Config for legacy browsers
presets: [
  [
    "@babel/preset-env", {
      modules: false,
      useBuiltIns: "usage",
      targets: "> 0.25%, IE > 10, Firefox ESR"
    }
  ]
]
```

```
// Config for modern browsers
presets: [
  [
    "@babel/preset-env", {
      modules: false,
      targets: {
        esmodules: true
      }
    }
  ]
]
```



```
// babel.config.js
module.exports = {
  env: {
    clientLegacy: {
      presets: [
        [
          "@babel/preset-env", {
            modules: false,
            targets: "> 0.25%, IE > 10, Firefox ESR"
          }
        ]
      ]
    },
    clientModern: {
      presets: [
        [
          "@babel/preset-env", {
            modules: false,
            targets: {
              esmodules: true
            }
          }
        ]
      ]
    }
  }
};
```

```
// Legacy config ...
module: {
  rules: [
    {
      test: /\.m?js$/i,
      exclude: /node_modules/i,
      use: [
        {
          loader: "babel-loader",
          options: {
            envName: "clientLegacy"
          }
        }
      ]
    }
  ]
},
// ...
```



```
// Modern config ...
module: {
  rules: [
    {
      test: /\.m?js$/i,
      exclude: /node_modules/i,
      use: [
        {
          loader: "babel-loader",
          options: {
            envName: "clientModern"
          }
        }
      ]
    }
  ]
},
// ...
```

Treemap sizes:

StatParsedGzipped

Search modules:

Enter regexpX

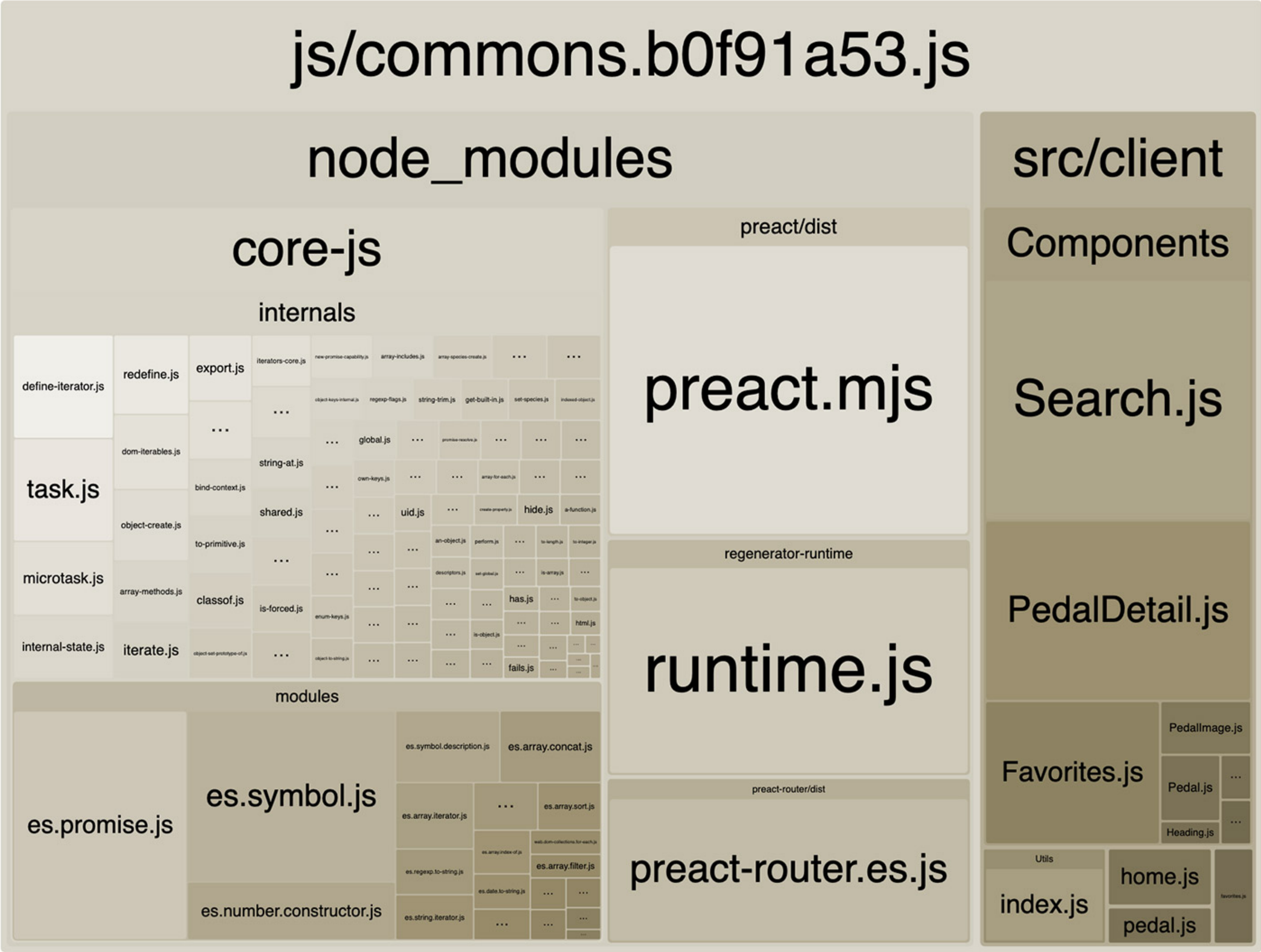
- Show chunks:
- ☒ All (68.48 KB)

☒ js/commons.b0f91a53.js (64.05 KB)

☒ js/favorites.760ee393.js (1.48 KB)

☒ js/pedal.2f0b52ea.js (1.47 KB)

☒ js/home.9c5b675b.js (1.47 KB)



LEGACY BROWSERS: 68.48 KB

Treemap sizes:

Stat **Parsed** Gzipped

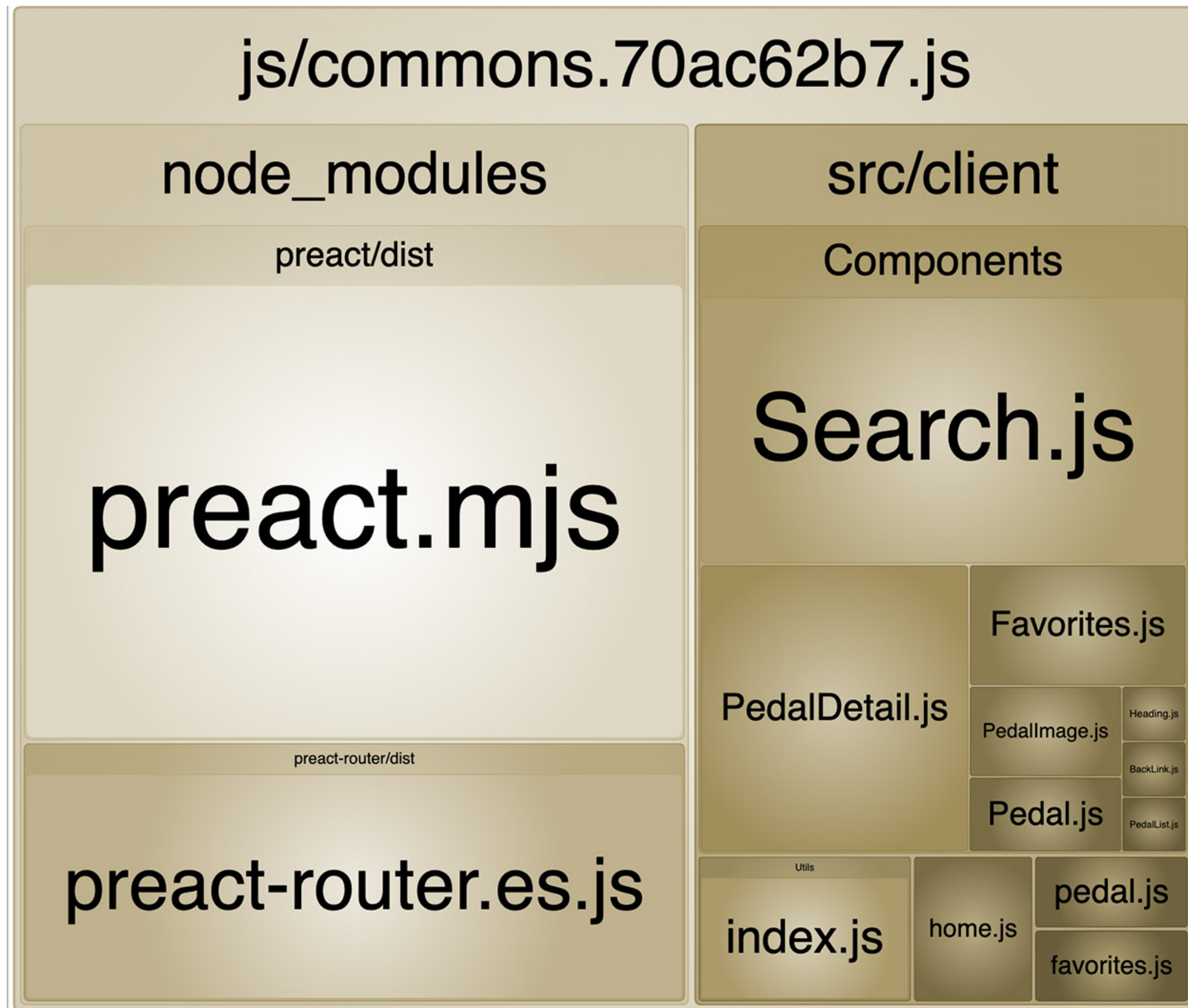
Search modules:

Enter regexp

x

Show chunks:

- ☒ All (**26.75 KB**)
- ☒ js/commons.70ac62b7.js (**22.32 KB**)
- ☒ js/favorites.8c4199f2.js (**1.48 KB**)
- ☒ js/pedal.b0a6b91b.js (**1.47 KB**)
- ☒ js/home.1350bbcf.js (**1.47 KB**)



MODERN BROWSERS: 26.75 KB

A Less Risky Differential Serving Pattern

18 July, 2019

2019 has been the year of differential serving for me. I've written [an article](#) about it. I've given [a talk](#) about it. I've even been helping one of my clients to roll it out broadly across their entire site. It's a great way to cut down on the amount of JavaScript you serve to your site's visitors without sacrificing features.

The pattern we use to ensure both modern and legacy browsers get the scripts that are appropriately transpiled according to their capabilities looks like this:

```
<script type="module" src="/js/modern.mjs"></script>  
<script nomodule defer src="/js/legacy.js"></script>
```


BE ACCOMMODATING

The Unacceptable Persistence of the Digital Divide

Millions of Americans lack broadband access and computer skills. Can President Trump bring them into the digital economy?

by David Talbot December 16, 2016

M

ost homes in the United States have Internet service, but they

don't in the poor parts of Cleveland and nearby suburbs. **A**

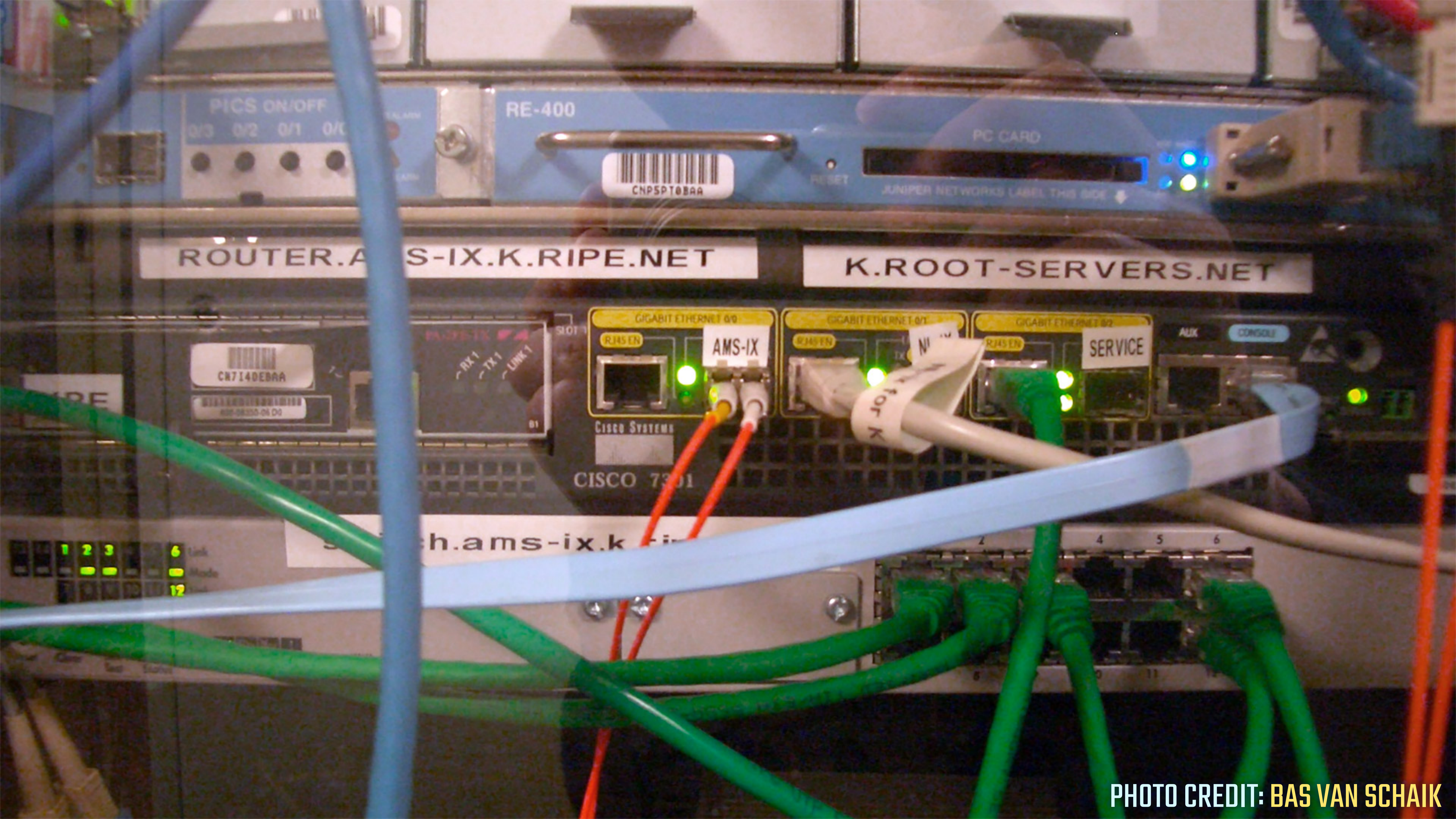
survey in 2012 showed that 58 percent of the area's

households with incomes under \$20,000 had neither home broadband

nor mobile Internet access, often because of the cost. Another 10

percent had a mobile phone but no home broadband. Until recently, one

Marcella and Ma’Niyah are among the millions of people on the wrong side of America’s persistent digital divide. A survey by Pew Research shows that fully one-third of American adults do not subscribe to any Internet access faster than dial-up at their home at a time when many basic tasks—finding job listings, doing homework, obtaining social services, and even performing many jobs—require being online. Even many people who are willing to pay for service can’t get it. Thirty-four million Americans have no access at all to broadband as the U.S. Federal Communications Commission defines it: a download speed of at least 25 megabits per second and an upload speed of three megabits per second. These speeds are what FCC chairman Tom Wheeler calls “table stakes for 21st-century communications.”



RTT

Approximate round trip time (*ms*)

Downlink

Approximate download speed (*kbps*)

ECT

Effective connection type ("4g", "3g", "2g", "*slow-2g*")

Accept-CH: RTT, Downlink, ECT

Accept-CH-Lifetime: 86400


```
<?php
$sect = "4g" ;

if (isset($_SERVER["HTTP_ECT"])) {
    $sect = $_SERVER["HTTP_ECT"] ;
}
?>
```

```
<?php
if ($sect === "4g" || $sect === "3g") {
    ?>
    <div class="carousel">
        <!-- Carousel content... -->
    </div>
    <script defer src="/js/carousel.js"></script>
    <?php
}
?>
```


ADAPTIVE PERFORMANCE

555-555-5555



INFO@SCONNIETIMBER.COM

SCONNIE
TIMBER INC

OUR PROCESS

LOCATIONS

CONTACT US

Our management team believes we can benefit central Wisconsin's landowners best by providing additional services that compliment commercial logging operations. As industry leaders, we strive to continue to provide excellent service by employing two accredited foresters that can assist landowners with a variety of management objectives. See our Q&A section below to better understand the process and some of the details associated with working with us.



Q&A

► Q: How many years have you been in business?

► Q: How many timber sales do you do in a year?

► Q: How long does it take to complete a timber sale?


► Q: How long does it take from the beginning of the process to the

WISCONSIN'S LEADER IN FOREST
MANAGEMENT AND LOGGING
SINCE 1952

22 REQUESTS, 740 KB

91.26 SECONDS OVER 2G

555-555-5555



INFO@SCONNIETIMBER.COM

SCONNIE
TIMBER INC

OUR PROCESS

LOCATIONS

CONTACT US

Our management team believes we can benefit central Wisconsin's landowners best by providing additional services that compliment commercial logging operations. As industry leaders, we strive to continue to provide excellent service by employing two accredited foresters that can assist landowners with a variety of management objectives. See our Q&A section below to better understand the process and some of the details associated with working with us.

Q&A

Q: How many years have you been in business?

A: We have been in business since 1952. We can't believe we have had the honor of working with central Wisconsin landowners for over 40 years. We are proud of our past but we are always looking forward to the future and to make the next 40 years even better.

Q: How many timber sales do you do in a year?

A: In a typical year we do around 175 individual timber sales. These sales are a mixture of state land, county land, MFL properties and private land.

Q: How long does it take to complete a timber sale?

A: The time frame can vary based on many factors. Acreage, type of trees to be harvested, type of harvest being conducted and ground/soil conditions can all affect the time frame of a harvest.

Q: How long does it take from the beginning of the process to the end?

A: The first step is to contact one of our foresters to schedule an appointment to look at your property. From there we will send a bid on the designated timber to be harvested, followed by a timber sale contract if everything looks good to you in the bid. Once we have a signed timber sale contract in house we will schedule any work that needs to be done and schedule the

WISCONSIN'S LEADER IN FOREST
MANAGEMENT AND LOGGING
SINCE 1952

5 REQUESTS, 12 KB

5.17 SECONDS OVER 2G



Web Fundamentals

GUIDES

CODE LABS

SAMPLES

[Replace Animated GIFs with Video](#)[JavaScript Start-up Optimization](#)[Loading Third-Party JavaScript](#)[Web Font Optimization](#)[HTTP Caching](#)[Adapting to Users with Client Hints](#)[Delivering Fast and Light Applications with Save-Data](#)[▶ Optimizing JavaScript](#)[▶ Never Load the Same Resource Twice](#)[▶ Lazy Loading Resources](#)[▶ Order Loading Thoughtfully](#)[PRPL Pattern](#)[Resource Prioritization](#)[▶ Web Performance Optimization with webpack](#)[▶ Rendering Performance](#)[▶ Audit your site](#)

Adapting to Users with Client Hints

**By** [Jeremy Wagner](#)

Jeremy is a contributor to WebFundamentals

Developing sites that are fast everywhere can be a tricky prospect. The plethora of device capabilities—and the quality of the networks they connect to—can make it seem like an insurmountable task. While we can take advantage of browser features to improve loading performance, how do we know what the user's device is capable of, or the quality of their network connection? The solution is [client hints](#)!

Client hints are a set of opt-in HTTP request headers that give us insight into these aspects of the user's device and the network they're connected to. By tapping into this information server side, we can change *how* we deliver content based on device and/or network conditions. This can help us to create more inclusive user experiences.

It's All About Content Negotiation

Contents

[It's All About Content Negotiation](#)[Opting in](#)[All the client hints!](#)[Device hints](#)[Network hints](#)[Tying it all together](#)[Responsive Images](#)[Helping users on slow networks](#)[Mind those caches!](#)[Client hints in service workers](#)[Wrapping up](#)[Resources](#)

FIGURE OUT WHAT PEOPLE WANT
AND WORK BACKWARD FROM THERE





THANK YOU

JEREMY WAGNER — @MALCHATA — JEREMY.CODES