

MAY 16, 2018

DEPLOYING ES6 IN PRODUCTION



米粽

CODER / @MYST729 / ELEME INC.

JUNE, 2015

ECMAScript 2015 (ES6)

DECEMBER, 1999

ECMAScript 3 (ES3)

DECEMBER, 2009

ECMAScript 5 (ES5)

JUNE, 2015

ECMAScript 2015 (ES6)

APRIL, 2016

TAOBAO AND TMALL TO STOP SUPPORTING INTERNET EXPLORER 6

IE6

AUGUST 27, 2001 🤯

Sort by Engine types

Show obsolete platforms

Show unstable platforms

V8 SpiderMonkey JavaScriptCore
Minor difference (1 point) Small feature (2 points)

Feature name		Current browser	Compilers/polyfills								
			Traceur	Babel 6 + core-js 2	Babel 7 + core-js 2	Babel 7 + core-js 3	Closure 2019.03	TypeScript + core-js 3	es6-shim	Konq 4.14 ^[1]	IE 11
Optimisation		98%	56%	71%	71%	72%	50%	69%	17%	5%	11%
proper tail calls (tail call optimisation)		0/2	0/2	0/2	0/2	0/2	0/2	0/2	0/2	0/2	0/2
Syntax											
default function parameters		7/7	4/7	4/7	4/7	4/7	5/7	5/7	0/7	0/7	0/7
rest parameters		5/5	4/5	3/5	3/5	3/5	2/5	4/5	0/5	0/5	0/5
spread syntax for iterable objects		15/15	15/15	13/15	13/15	14/15	11/15	14/15	0/15	0/15	0/15
object literal extensions		6/6	6/6	6/6	6/6	6/6	5/6	6/6	0/6	0/6	0/6
for..of loops		9/9	9/9	9/9	9/9	9/9	6/9	9/9	0/9	0/9	0/9
octal and binary literals		4/4	2/4	4/4	4/4	4/4	2/4	4/4	2/4	0/4	0/4
template literals		7/7	6/7	6/7	6/7	6/7	5/7	5/7	0/7	0/7	0/7
RegExp "y" and "u" flags		6/6	4/6	4/6	4/6	4/6	0/6	0/6	0/6	0/6	0/6
destructuring, declarations		22/22	20/22	21/22	21/22	21/22	20/22	21/22	0/22	0/22	0/22
destructuring, assignment		24/24	23/24	24/24	24/24	24/24	22/24	24/24	0/24	0/24	0/24
destructuring, parameters		24/24	19/24	21/24	21/24	21/24	20/24	21/24	0/24	0/24	0/24
Unicode code point escapes		2/2	1/2	1/2	1/2	1/2	1/2	1/2	0/2	0/2	0/2
new.target		2/2	0/2	0/2	0/2	0/2	0/2	0/2	0/2	0/2	0/2
Bindings											
const		18/18	16/18	16/18	16/18	16/18	16/18	16/18	0/18	2/18	14/18
let		14/14	12/14	12/14	12/14	12/14	12/14	12/14	0/14	0/14	12/14

ES5 vs. ES2015+

Version	Size (minified)	Size (minified + gzipped)	Parse/eval time
ES5	175K	43K	367ms
ES2015+	80K 🍑	21K 🍑	172ms 🍑



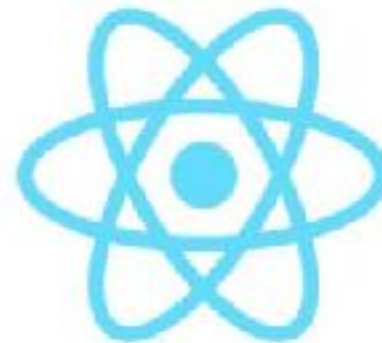
我有个大胆的想法



TypeScript



BABEL



BABEL

.BABELRC

```
{
  "presets": [
    [ "env", {
      "modules": false,
      "useBuiltIns": true,
      "targets": {
        "browsers": [
          "> 1%",
          "last 2 versions",
          "not ie <= 8"
        ]
      }
    } ]
  ]
}
```



```
{
  "presets": [
    [ "env", {
      "modules": false,
      "useBuiltIns": true,
      "targets": {
        "browsers": [
          "Chrome >= 60",
          "Safari >= 10.1",
          "iOS >= 10.3",
          "Firefox >= 54",
          "Edge >= 15"
        ]
      }
    } ]
  ]
}
```

.BABELRC 🙌

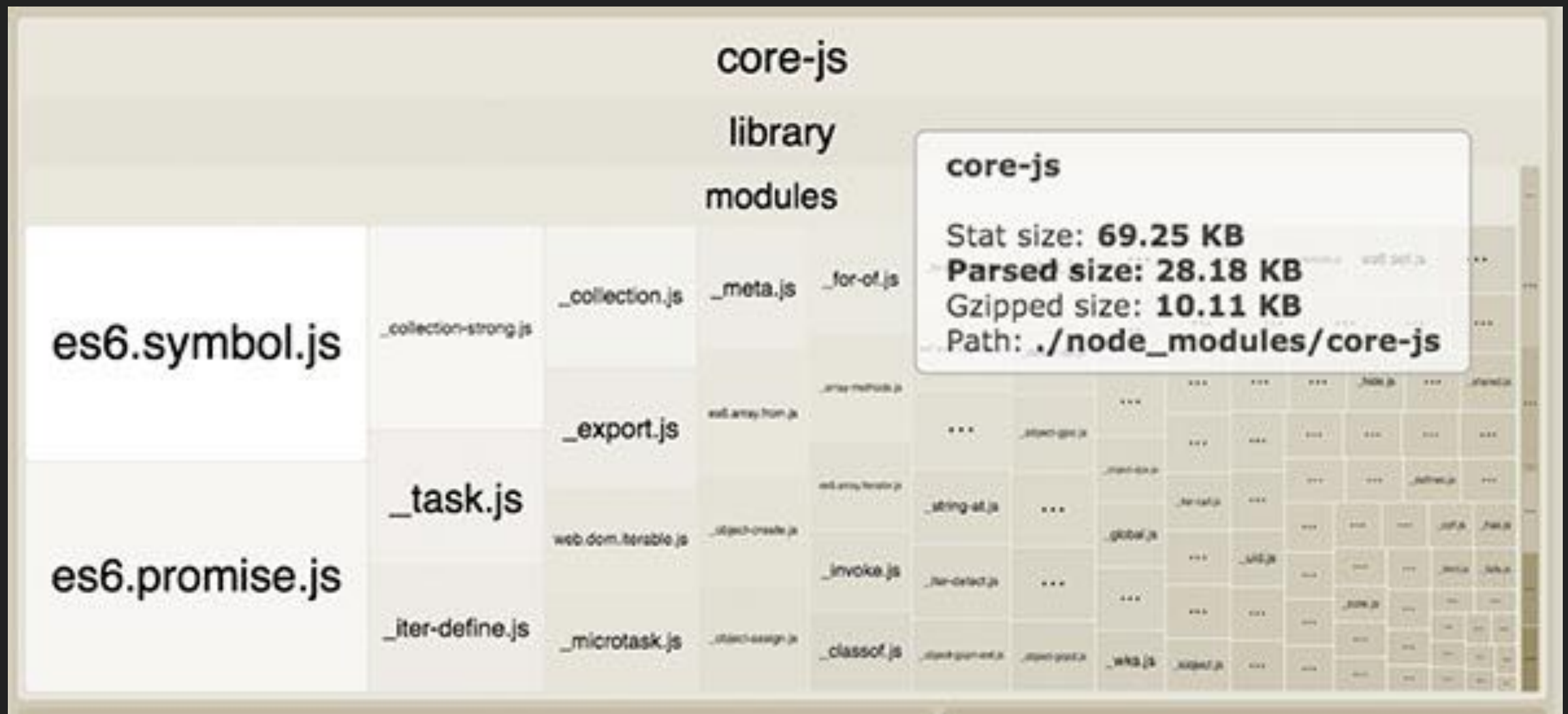
```
{
  "presets": [
    [ "env", {
      "modules": false,
      "useBuiltIns": true,
      "targets": {
        "browsers": [
          "> 1%",
          "last 2 versions",
          "not ie <= 8"
        ]
      }
    } ]
  ]
}
```



```
{
  "presets": [
    [ "env", {
      "modules": false,
      "useBuiltIns": true,
      "targets": {
        "browsers": [
          "Chrome >= 60",
          "Safari >= 10.1",
          "iOS >= 10.3",
          "Firefox >= 54",
          "Edge >= 15"
        ]
      }
    } ]
  ]
}
```

CORE-JS

10.11 KB GZIPPED



REGENERATOR-RUNTIME

2.37 KB GZIPPED

regenerator-runtime

regenerator-runtime

Stat size: **24.61 KB**

Parsed size: 6.43 KB

Gzipped size: **2.37 KB**

Path: **./node_modules/regenerator-runtime**



BACKWARD COMPATIBLE?

MODULES VIA <SCRIPT> TAG

<!-- ✓ : download and run -->

<!-- ✗ : download -->

```
<script type="module" src="js/es2015.js"></script>
```

<!-- ✓ : does not download -->

<!-- ✗ : download and run -->

```
<script nomodule src="js/es5.js"></script>
```



完美!



KĒNG

坑

U+5751

PITFALLS

- ▶ **"MODULE" VS. "TEXT/JAVASCRIPT"**
- ▶ **UGLIFYJS: ES2015 WHO?**
- ▶ **TYPE="MODULE", TYPE "MORE DUE"?**
- ▶ **SAFARI 10 IS THE NEW IE6**
- ▶ **FEATURE DETECTION FAILS...**

IN MODULES...

- ▶ CODE ALWAYS RUN IN STRICT MODE.
- ▶ TOP-LEVEL VAR AND FUNCTION DECLARATIONS CAN'T BE ACCESSED FROM WINDOW.



UGLIFYJS DOESN'T RECOGNIZE ES2015

```
module.exports = {  
  plugins: [  
    new webpack.optimize.UglifyJsPlugin()  
  ]  
}
```



```
const UglifyJsPlugin = require('uglifyjs-webpack-plugin')  
  
module.exports = {  
  plugins: [  
    new UglifyJsPlugin()  
  ]  
}
```

UGLIFYJS DOESN'T RECOGNIZE ES2015

```
module.exports = {  
  plugins: [  
    new webpack.optimize.UglifyJsPlugin()  
  ]  
}
```



```
const UglifyJsPlugin = require('uglifyjs-webpack-plugin')
```

```
module.exports = {  
  plugins: [  
    new UglifyJsPlugin()  
  ]  
}
```



TYPE="MODULE" BEHAVES LIKE DEFER

```
<link rel="preload" as="script" href="js/es2015.js">
```



SAFARI 10: I DON'T KNOW NOMODULE.

```
(function () {  
  var check = document.createElement('script')  
  if (!('noModule' in check) && 'onbeforeload' in check) {  
    var support = false  
    document.addEventListener('beforeload', function (e) {  
      if (e.target === check) {  
        support = true  
      } else if (!e.target.hasAttribute('nomodule') || !support) {  
        return  
      }  
      e.preventDefault()  
    }, true)  
  
    check.type = 'module'  
    check.src = '.'  
    document.head.appendChild(check)  
    check.remove()  
  }  
})()
```



SAFARI 10: HOW OLD ARE YOU?

(怎么老是你)

```
let e = e => {  
  console.log(e)  
  for (let e of [1, 2, 3])  
    console.log(e)  
}
```



SAFARI 10: HOW OLD ARE YOU?

(怎么老是你)

```
let e = e => {  
  console.log(e)  
  for (let e of [1, 2, 3])  
    console.log(e)  
}
```



```
// Safari 10  
SyntaxError: Cannot declare a let variable twice: 'e'.
```

SAFARI 10: SYNTAXERROR.

```
const UglifyJsPlugin = require('uglifyjs-webpack-plugin')

module.exports = {
  plugins: [
    new UglifyJsPlugin({
      mangle: {
        safari10: true
      }
    })
  ]
}
```



FEATURE DETECTION FAILS...

IE	Edge *	Firefox	Chrome	Safari	Opera	iOS Safari	Opera Mini	Android *	Blackberry	Opera Mobile *	Chrome Android	Firefox Android	IE Mobile	UC for Android	Samsung Internet	QQ	Baidu
6		54	60	9	43	9.2		4									
7	12	² 55	61	9.1	46	9.3		4.1									
8	13	² 56	62	10	¹ 47	10.2		4.3									
9	14	² 57	63	⁴ 10.1	¹ 48	⁴ 10.3		4.4		12					4		
10	¹ 15	² 58	64	11	¹ 49	11.2		4.4.4	7	12.1			10		5		
11	16	² 59	65	11.1	¹ 50	11.3	all	62	10	37	64	² 57	11	11.8	6.2	1.2	7.12
	17	60	66	TP	¹ 51												
	18	61	67		¹ 52												
			68														

Notes

Known issues (0)

Resources (12)

Feedback

¹ Support can be enabled via `about:flags`

² Support can be enabled via `about:config`

⁴ Does not support the `nomodule` attribute



STAGED TO PRODUCTION

Hit Ratio

Version	Expected	Actual
ES5	< 40%	67%
ES2015	> 60%	33% 🙄

STAGED TO PRODUCTION

Business Code

Version	Size (minified)	Size (minified + gzipped)	Parse/eval time
ES5	288K	96.7K	636ms
ES2015	274K 👍	105K 👎	233ms 👍

STAGED TO PRODUCTION

Production Dependencies

Version	Size (minified)	Size (minified + gzipped)	Parse/eval time
ES5	307K	94.6K	75ms
ES2015	298K 🍑	92.3K 🍑	67ms 🍑

REQUESTING ES6 BUNDLES (#6937)

 vuejs / vue

 Watch ▾

4,809

★ Unstar

91,122



<> Code

🔔 Issues 117

🔗 Pull requests 51

📁 Projects 0

📊 Insights

Provide ES2015 bundle? #6937

Edit



Open

myst729 opened this issue on Oct 27, 2017 · 2 comments



myst729 commented on Oct 27, 2017 • edited ▾

Contributor



What problem does this feature solve?

Recently I'm working on a project that attempts to deploy ES2015 code to production. The idea was inspired by Philip Walton's [post](#).

I hacked [vuejs-templates/webpack](#) to [output dual bundles with feature detection to downgrade](#). The benefit is significant: ES2015 script execution time is only about 55% of ES5! And the bundle size reduces too. If vue (as well as vue-router, vuex, etc.) could provide ES2015 bundles, the performance boost can be even more significant.

According to the [es6 compatibility table](#), ES2015 compatible browsers have got a lot of market shares.

Assignees

No one assigned

Labels

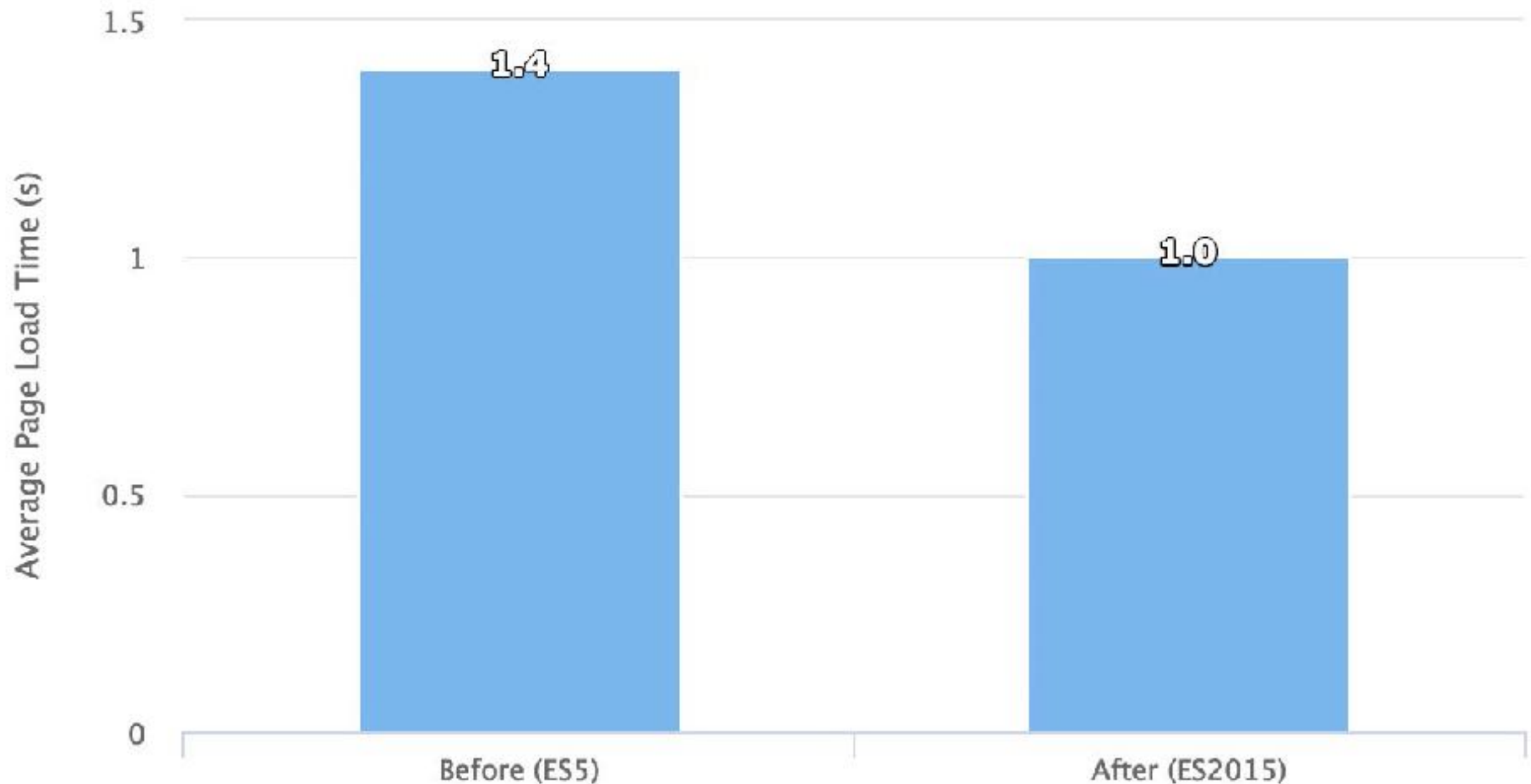
feature request

Projects

None yet

Milestone

PAGE LOAD TIME



MORE PITFALLS

MORE PITFALLS

实现 * * * * 伟大 * * 的 * * 梦



溜了溜了



SENTENCED?

SENTENCED?

CERTAINLY NOT



我在等红包

FEATURE SUBSET DETECTION

```
class Ƨ_Ƨ extends Array {
  constructor(j = 'a', ...c) {
    const q = (({u: e}) => {
      return { [`s${c}`]: Symbol(j) }
    })({})
    super(j, q, ...c)
  }
}

new Promise(f => {
  const a = function* () {
    return '\u{20BB7}'.match(/./u)[0].length === 2 || true
  }
  for (let r of a()) {
    const [s, ws, m, wm] = [new Set(), new WeakSet(), new Map(), new WeakMap()]
    break
  }
  f(new Proxy({}, {
    get: (h, k) => k in h ? h[k] : '42'.repeat(0o10)
  })))
}).then(t => new Ƨ_Ƨ(t.d))
```





FEATURE SUBSET DETECTION

```
<script>
  try {
    // Feature detection snippet
    var code = 'class Ø_Ø extends Array { constructor(j = "...')
    eval(code)
    // Feature detection succeeds
    document.write('\<script src="js/es2015.js"\></script\>')
  } catch (e) {
    // Feature detection fails
    document.write('\<script src="js/es5.js"\></script\>')
  }
</script>
```



FEATURE SUBSET DETECTION

PROS AND CONS

- ▶  HIGHER HIT RATIO
- ▶  LESS CODE DOWNLOADED ON OLD BROWSERS
- ▶  PERFORMANCE LOSS
- ▶  UNEXPECTED BABEL OUTPUT

DYNAMIC SERVING

THE **VARY** HTTP RESPONSE HEADER






`Vary: User-Agent`

- ▶ AVOID CACHE MISTAKENLY SERVED.
- ▶ HELP SEARCH ENGINES TO DISCOVER ALTERNATIVE VERSIONS.
- ▶ TELL SEARCH ENGINES NO CLOAKING INTENDED.



DYNAMIC SERVING

PROS AND CONS

- ▶  HIGHER HIT RATIO
- ▶  LESS CODE DOWNLOADED ON OLD BROWSERS
- ▶  PERFORMANCE BOOST
- ▶  MAINTENANCE OF UA WHITELIST
- ▶  FAULTY WHITELISTED

SEPARATE URLS

REL="CANONICAL" AND HTTP 301 REDIRECTS






```
<link href="//example.org/default" rel="canonical">
```

- ▶ TELL SEARCH ENGINES TO INDEX THE CANONICAL URL.
- ▶ TELL SEARCH ENGINES TO NOT INDEX NON-CANONICAL URLS.



SEPARATE URLS

PROS AND CONS

- ▶  HIGHER HIT RATIO
- ▶  LESS CODE DOWNLOADED ON OLD BROWSERS
- ▶  PERFORMANCE BOOST
- ▶  MAINTENANCE OF UA WHITELIST
- ▶  FAULTY WHITELISTED

REFERENCES

- ▶ DEPLOYING ES2015 CODE IN PRODUCTION TODAY
- ▶ ECMAScript 6 COMPATIBILITY TABLE
- ▶ ECMAScript 5 STRICT MODE
- ▶ UGLIFYJS WEBPACK PLUGIN
- ▶ SAFARI 10.1 NOMODULE SHIM
- ▶ W3C TECHNICAL REPORT: PRELOAD
- ▶ CAN I USE: ES6 MODULE
- ▶ SYNTAXERROR: CANNOT DECLARE A LET VARIABLE TWICE
- ▶ UGLIFYJS2: MANGLE OPTIONS
- ▶ ES6 FEATURE SUBSET DETECTION
- ▶ MDN: THE VARY HTTP RESPONSE HEADER
- ▶ WIKIPEDIA: CLOAKING
- ▶ BEST PRACTICES USING VARY HEADER
- ▶ CANONICAL LINK ELEMENT

