

A close-up photograph of a man with light brown hair, wearing a grey and white horizontally striped shirt. He is looking down at a black smartphone held in his hands. His expression is focused and slightly weary. The background is blurred.

Chérie, j'ai rétréci
mes bundles JS



Antoine Kahn-Dubois

Lead Developper chez Theodo
github.com/antkahn

Au menu

- Pourquoi diminuer la taille de son JavaScript ?
- Comment mettre en place le Code Splitting
- Comment choisir ses librairies ?
- Pour aller plus loin: le Bundle Splitting

Au menu

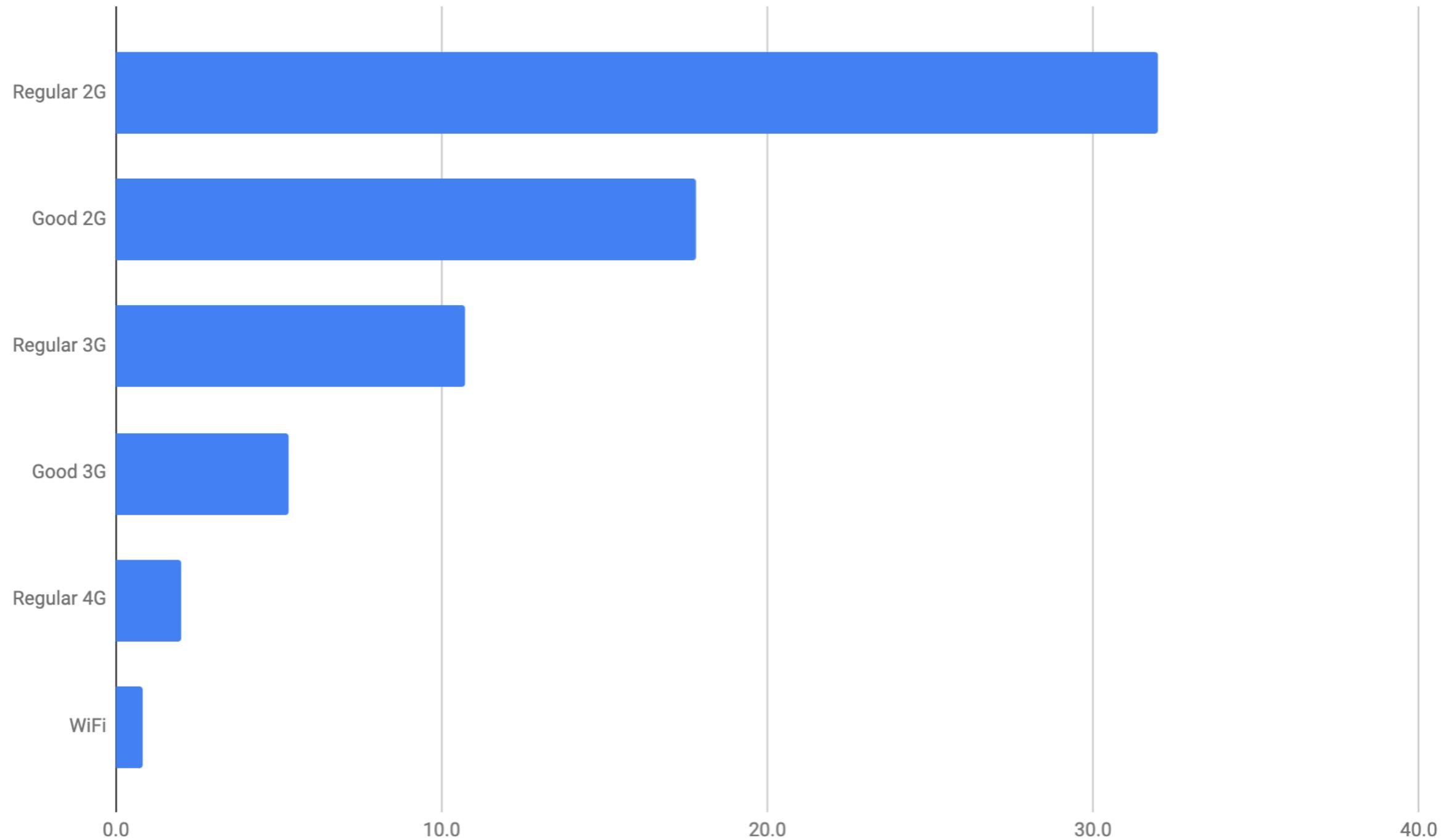
- Pourquoi diminuer la taille de son JavaScript ?
- Comment mettre en place le Code Splitting
- Comment choisir ses librairies ?
- Pour aller plus loin: le Bundle Splitting

Pourquoi réduire la taille de son JavaScript ?

Réduire le temps de:

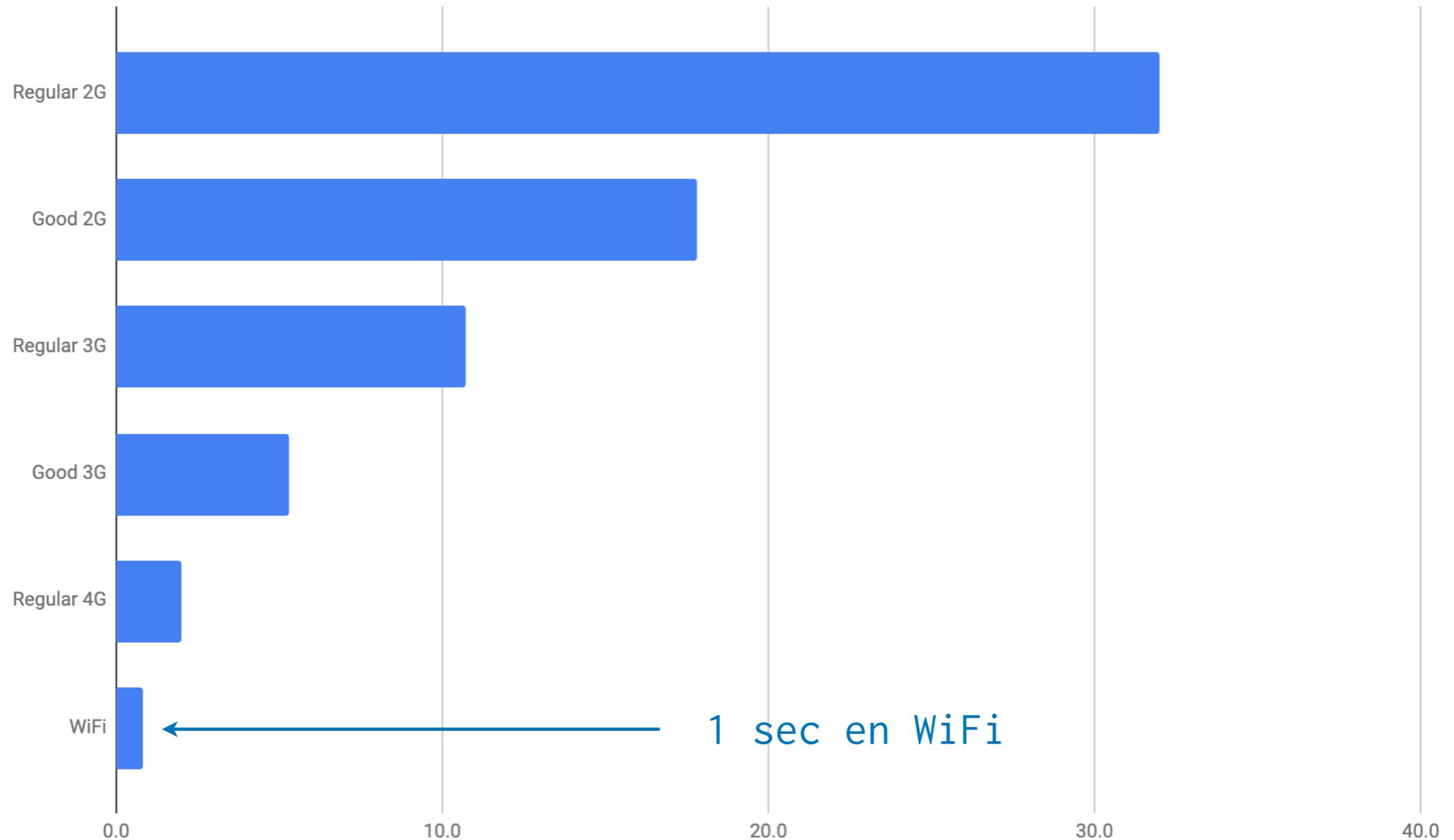
- Téléchargement

Temps pour télécharger 1MB de JavaScript



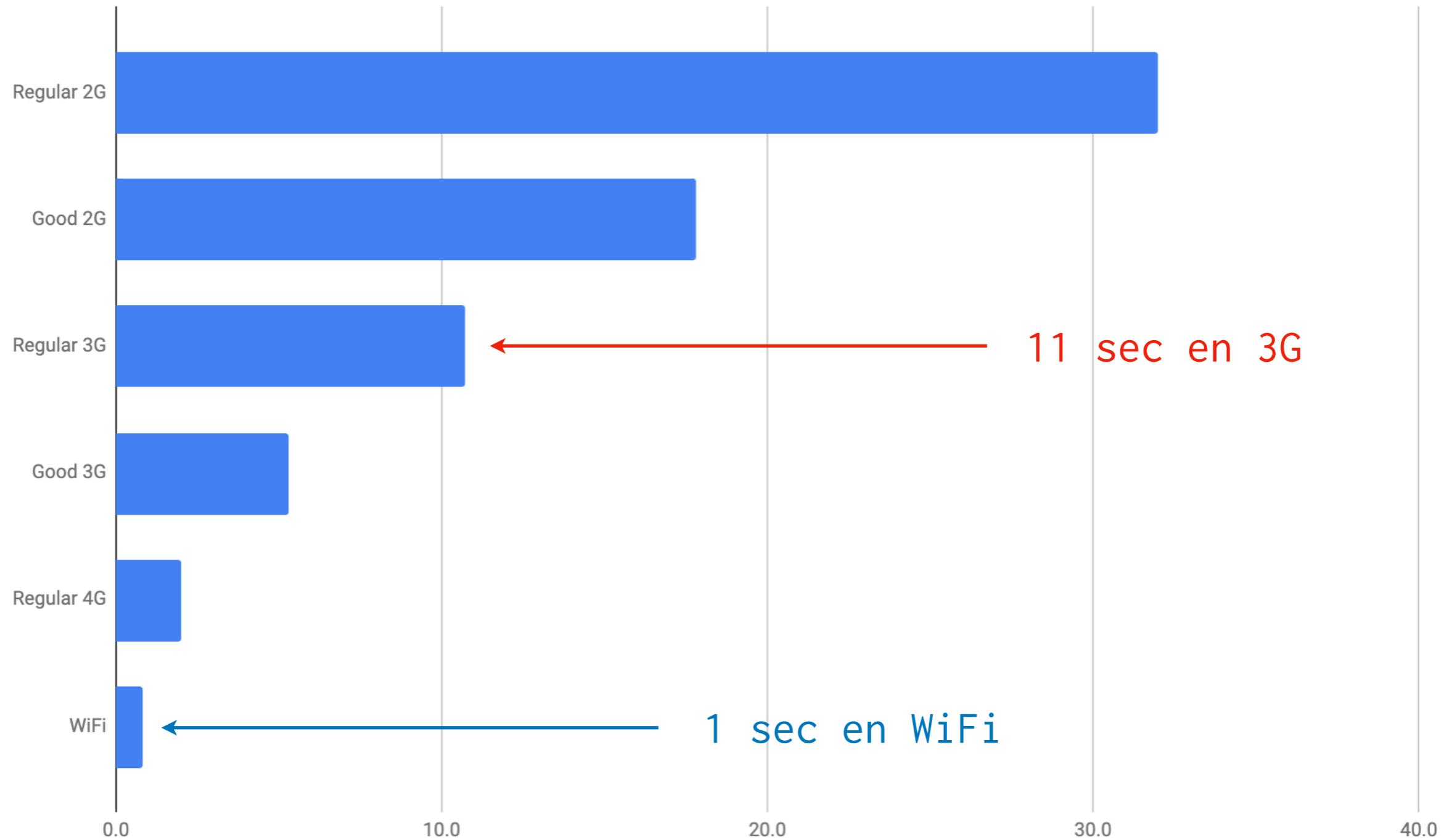
Source: ma calculette

Temps pour télécharger 1MB de JavaScript



Source: ma calculette

Temps pour télécharger 1MB de JavaScript



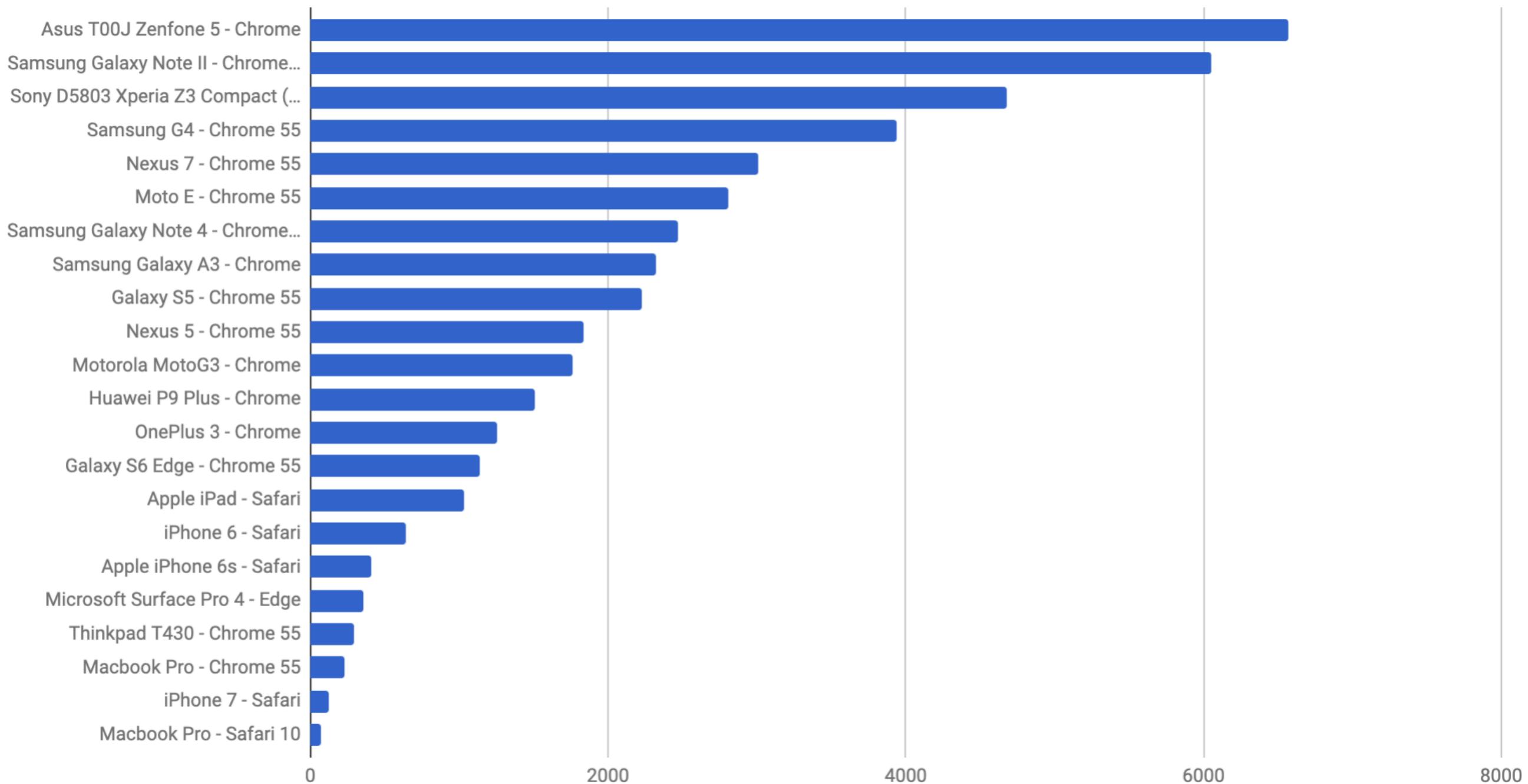
Source: ma calculette

Pourquoi réduire la taille de son JavaScript ?

Réduire le temps de:

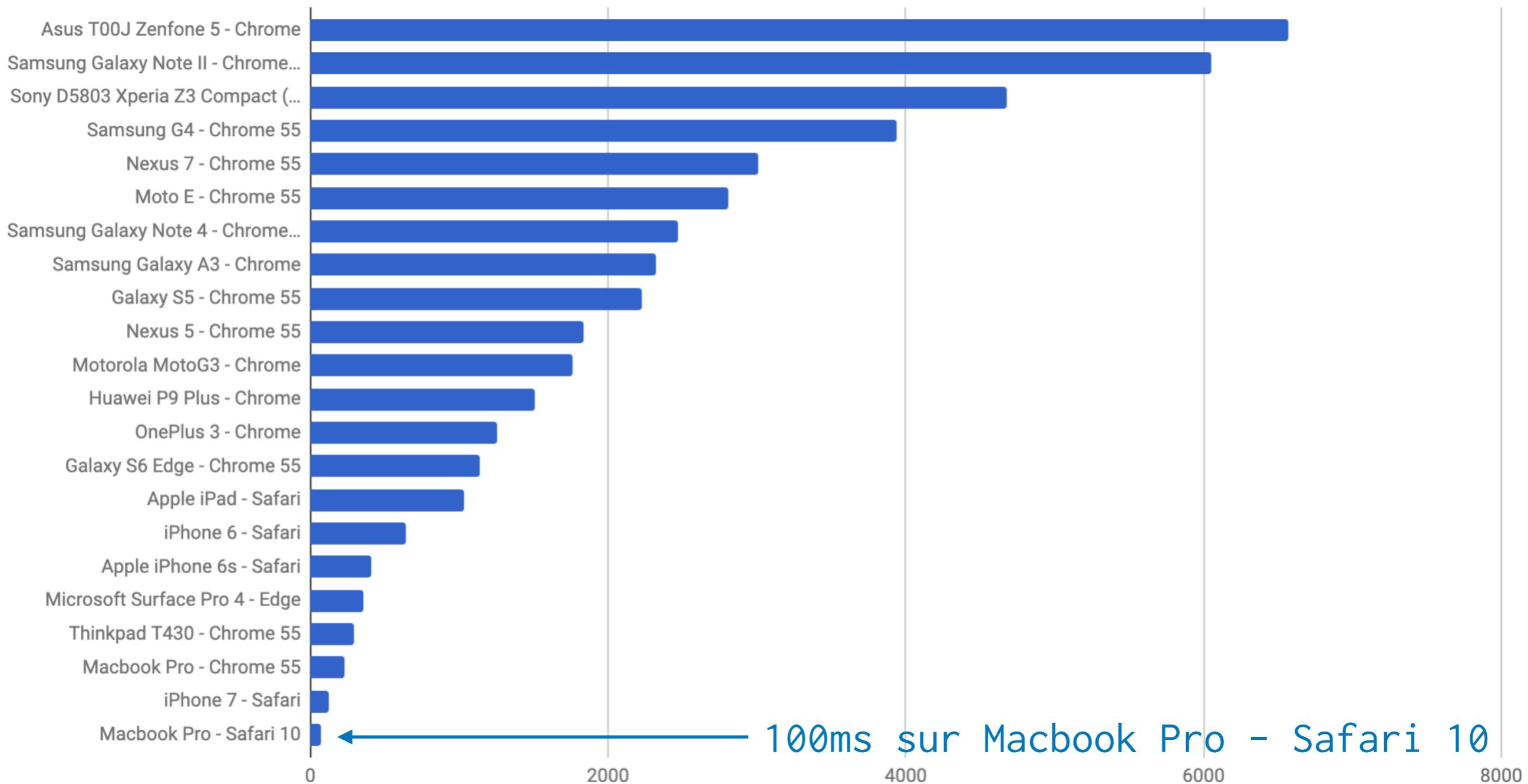
- Téléchargement
- Parsing

Temps pour parser 1MB de JavaScript



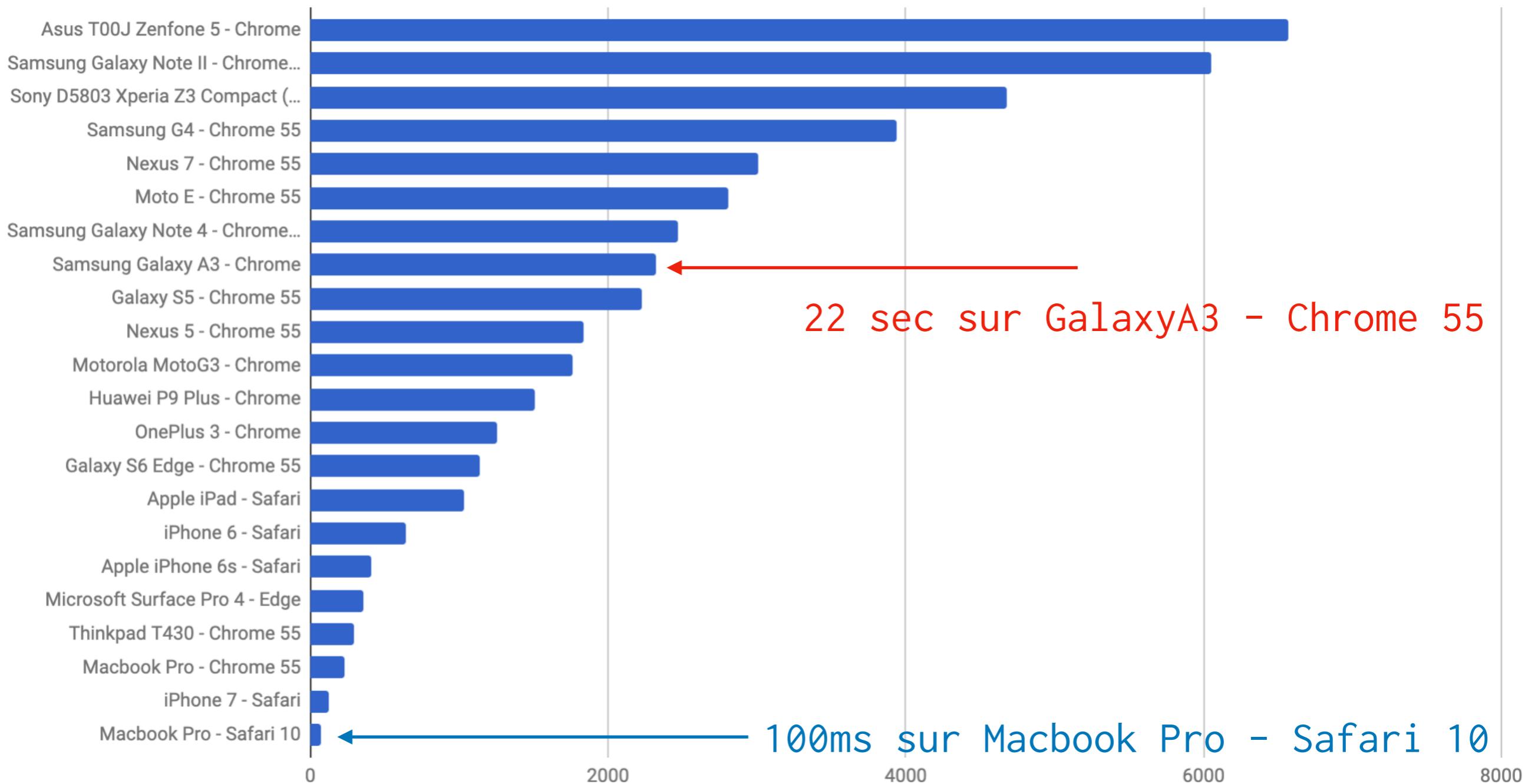
Source: @addyosmani, cost of JavaScript 2018

Temps pour parser 1MB de JavaScript



Source: @addyosmani, cost of JavaScript 2018

Temps pour parser 1MB de JavaScript



Source: @addyosmani, cost of JavaScript 2018

Pourquoi réduire la taille de son JavaScript ?

Réduire le temps de:

- Téléchargement
- Parsing
- Compilation & Exécution

Pourquoi réduire la taille de son JavaScript ?

Sauver la planète !

On va consommer moins d'énergie pour:

- Déplacer les fichiers du serveur au client
- Parser les fichiers chez le client

Pourquoi réduire la taille de son JavaScript ?

Taille conseillée: < 200KB Gzippé

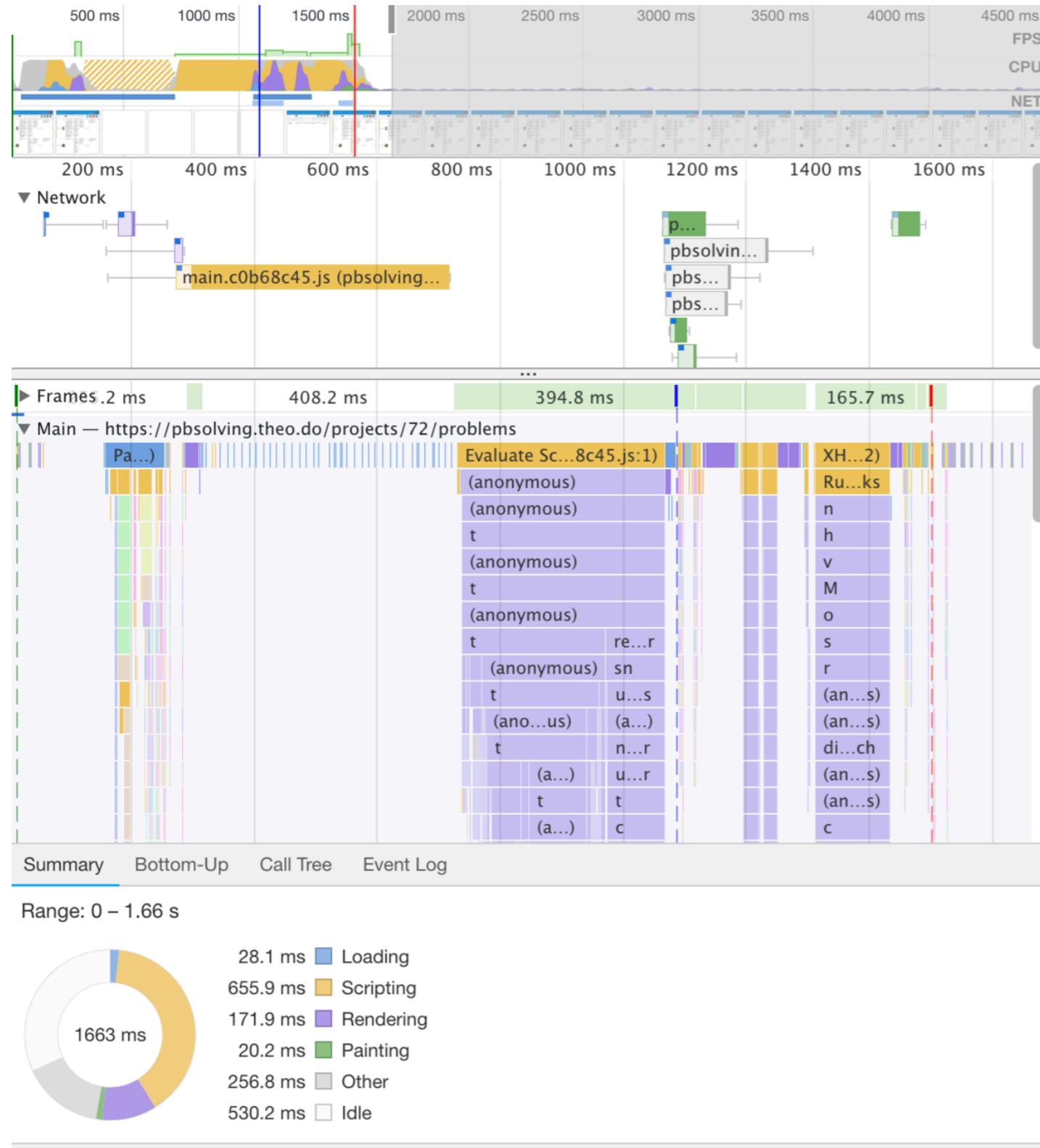
Au menu

- Pourquoi diminuer la taille de son JavaScript ?
- Comment mettre en place le Code Splitting
- Comment choisir ses librairies ?
- Pour aller plus loin: le Bundle Splitting

Application de test

- Single Page Application
- En React/Redux
- Une page de login
- Une page avec un tableau de data et des modales

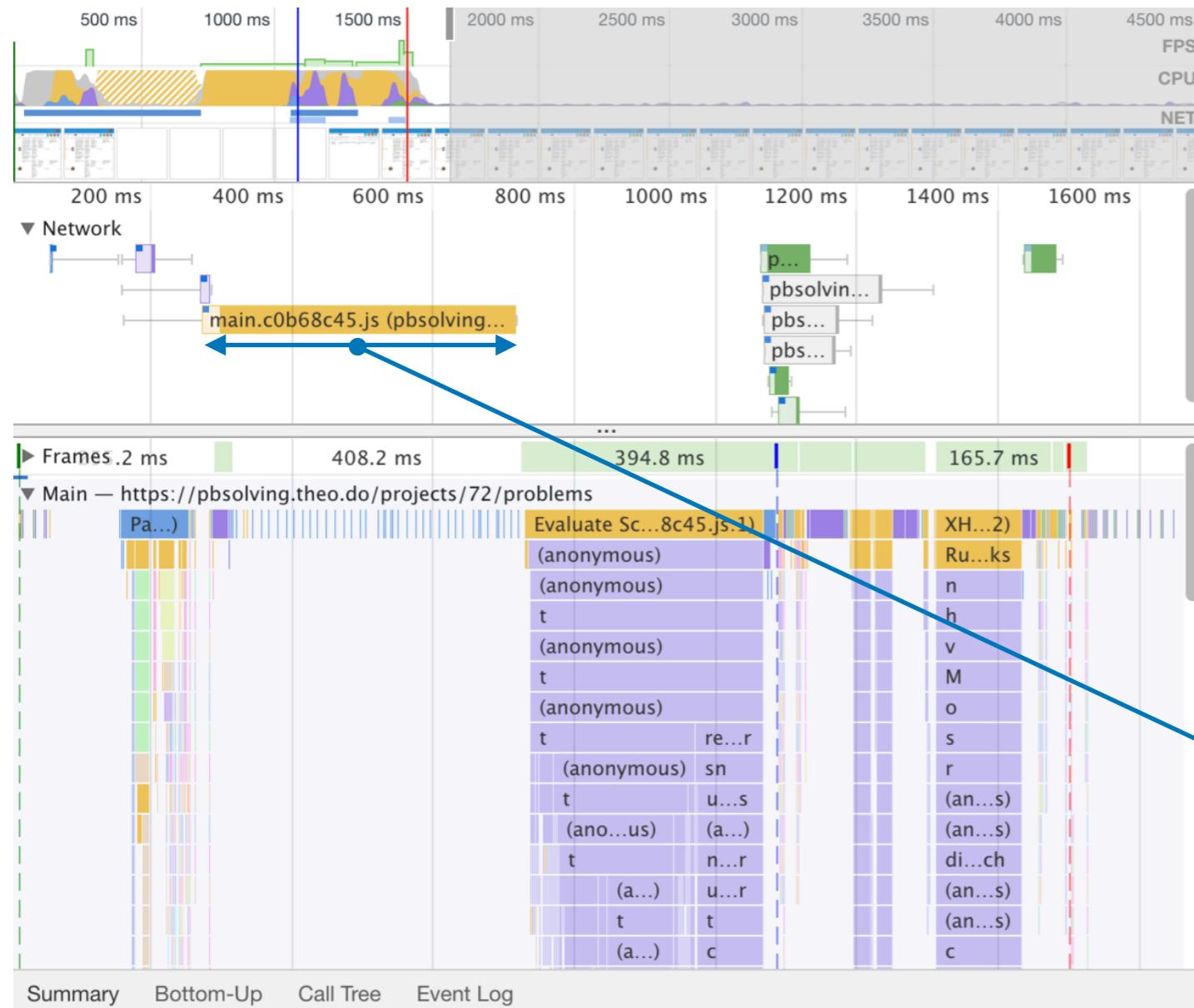
État initial (page principale)



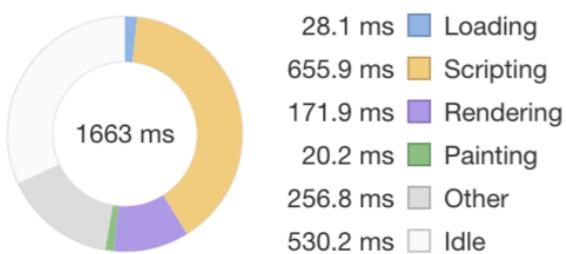
Temps de chargement: 1.3s

Sur Chrome, Macbook Pro

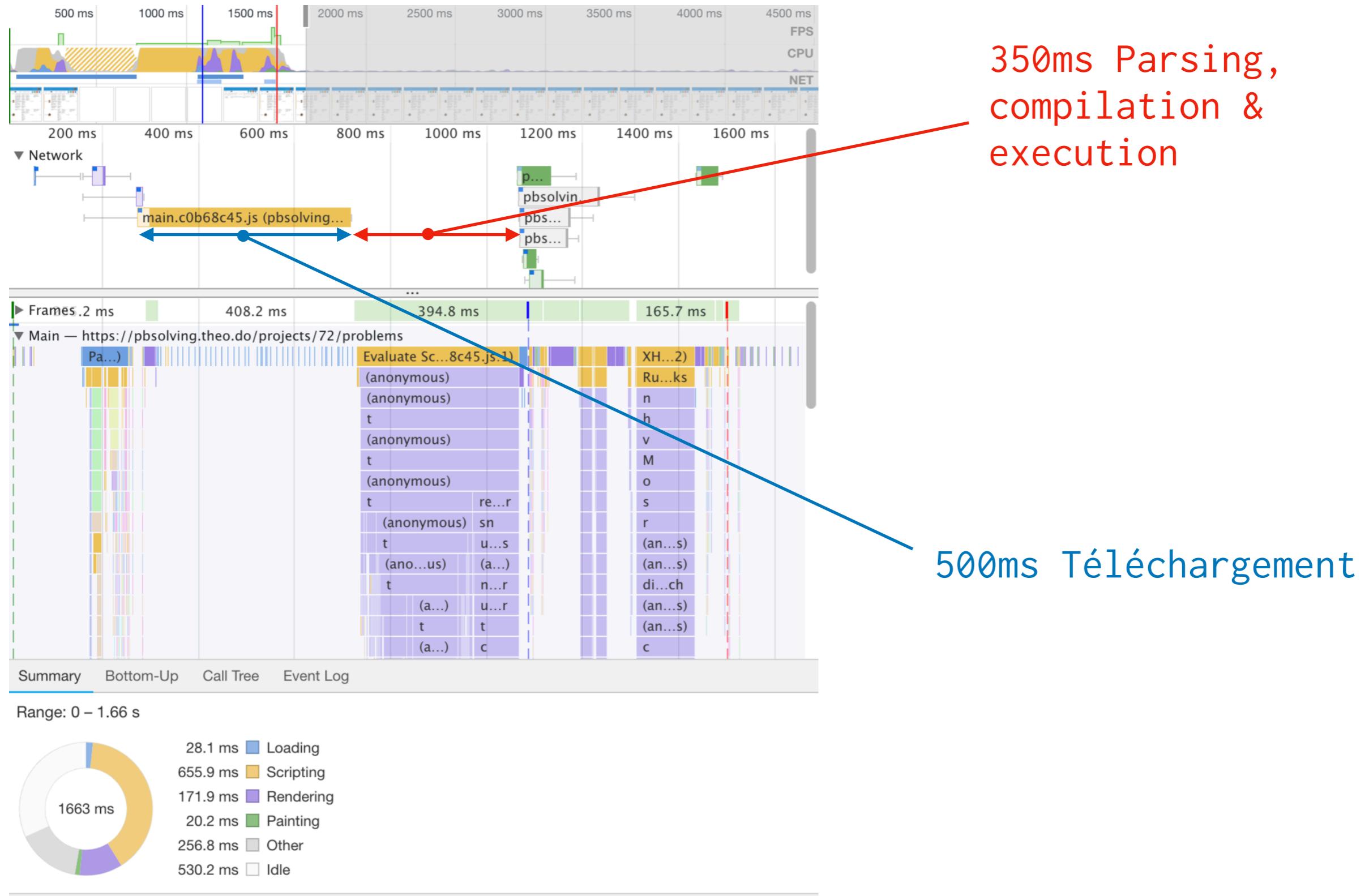
État initial (page principale)



500ms Téléchargement



État initial (page principale)



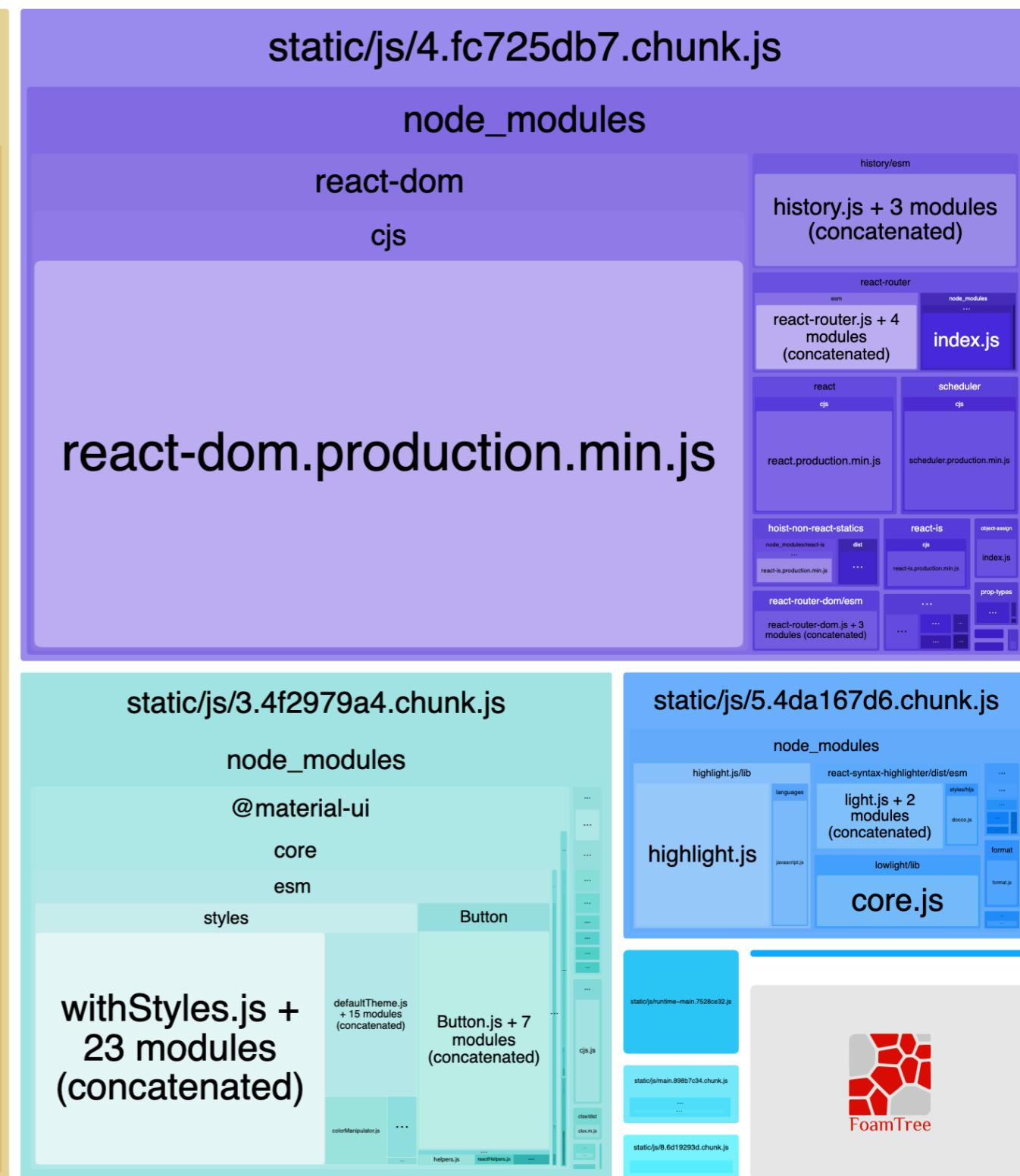
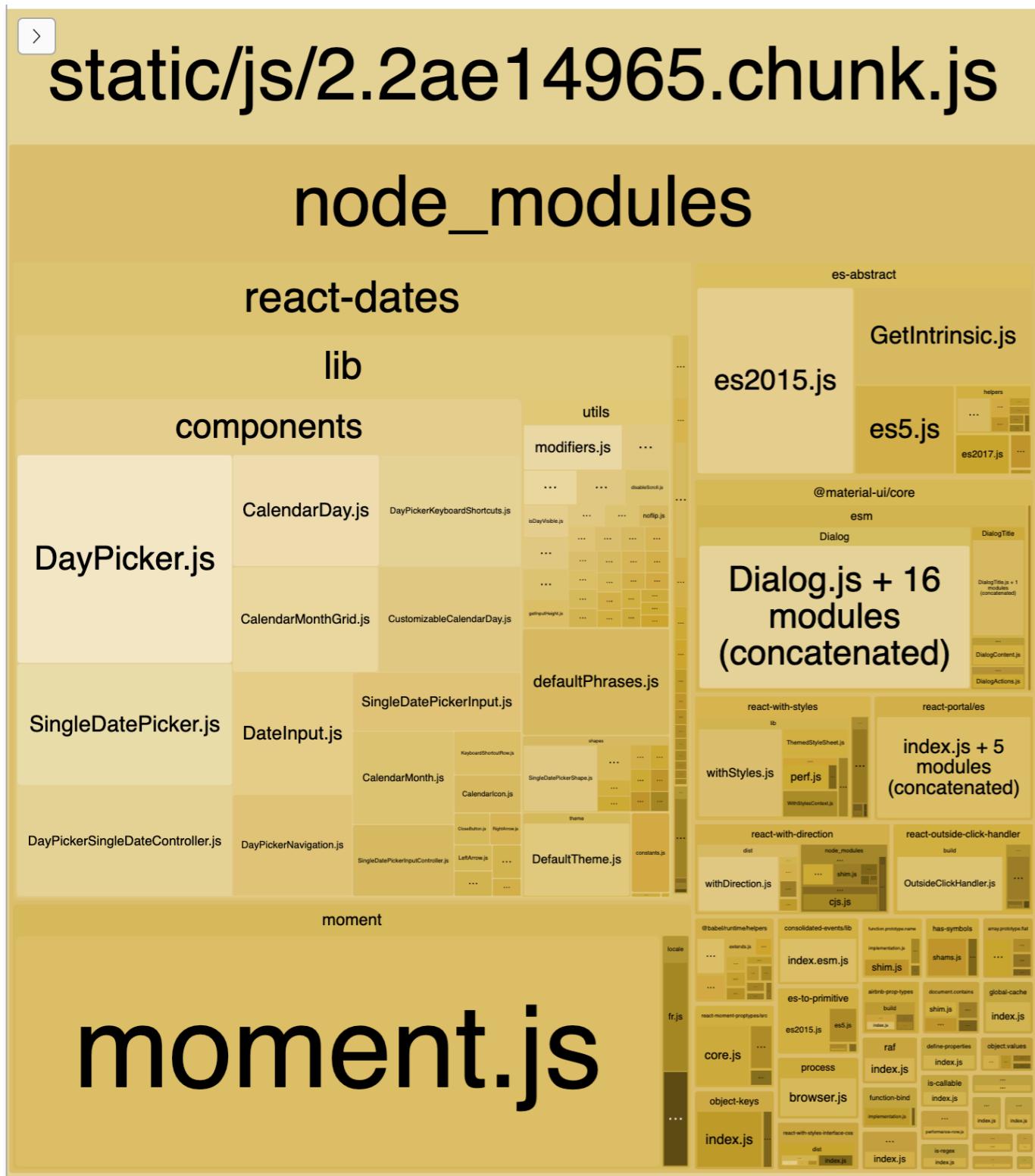
A close-up photograph of a yellow dog's face. The dog is wearing a blue bandana around its neck and is holding a magnifying glass with its front paws, examining a small, dark object. The background is dark and out of focus.

Qu'y a-t-il dans mon
Bundle?

Qu'y a-t-il dans mon bundle ?

L'outil idéal: webpack-bundle-analyzer
github.com/webpack-contrib/webpack-bundle-analyzer

C'est quoi webpack-bundle-analyzer ?



Ce qu'il y a dans mon bundle

600 KB

static/js/main.c0b68c45.js

node_modules

static/js/main.c0b68c45.js
Stat size: 4.55 MB
Parsed size: 2.32 MB
Gzipped size: 600.96 KB

highlight.js

lib

languages

mathematica.js

1c.js

maxima.js

sqf.js

x86asm.js

mel.js

stata.js

lsl.js

sql.js

gauss.js

vim.js

glsl.js

autoit.js

moment

irpf90.js verilog.js fortran.js julia.js arduino.js excel.js stylus.js

puppet.js swift.js processing.js hy.js scheme.js n1ql.js routeros.js lasso.js

mipsasm.js rust.js cs.js ruby.js d.js typescript.js kotlin.js matlab.js avrasm.js

clojure.js mercury.js xl.js vhdl.js coffeescript.js tcl.js django.js erlang.js

armasm.js ad.js tp.js xquery.js cmake.js dart.js pf.js nimrod.js python.js

gams.js php.js basic.js haxe.js stan.js rib.js groovy.js crmsh.js ocaml.js purebasic.js

crystal.js lua.js scalab.js cos.js nginx.js ... makefile.js rsl.js lisp.js dts.js

coq.js gradle.js elixir.js scala.js pony.js go.js xml.js vala.js css.js cal.js

oxygen.js bash.js asciidoc.js dns.js hamt.js nix.js thrift.js intercal.js

hsp.js objective.js vbnnet.js q.js asciidoc.js dns.js hamt.js nix.js thrift.js intercal.js

less.js opencat.js sharp.js monkey.js ceylon.js abnf.js fix.js http.js dtd.js profile.js

llvm.js moonscript.js monkey.js ceylon.js abnf.js fix.js http.js dtd.js profile.js

cpp.js dos.js smi.js ini.js prolog.js parser.js cop.js wml.js leaf.js

powershell.js perl.js aspect.js vbscript.js

qml.js dos.js smi.js ini.js prolog.js parser.js cop.js wml.js leaf.js

nsis.js nsislanguage.js javascript.js zephir.js java.js haskell.js twig.js gcode.js apache.js goto.js json.js wr.js

locales.min.js

moment.js

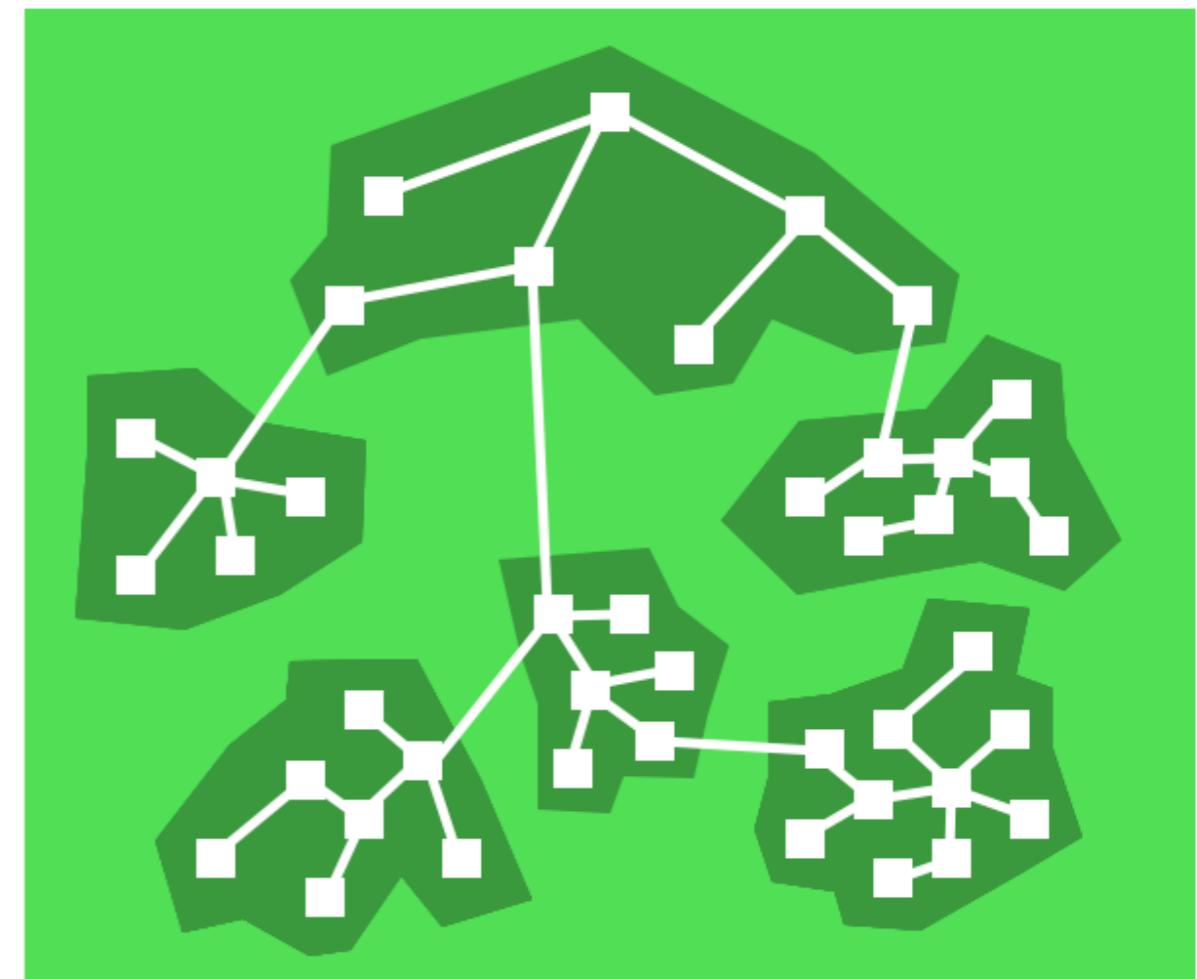


Code Splitting

Le but: apporter uniquement à l'utilisateur ce dont il a besoin, quand il en a besoin.

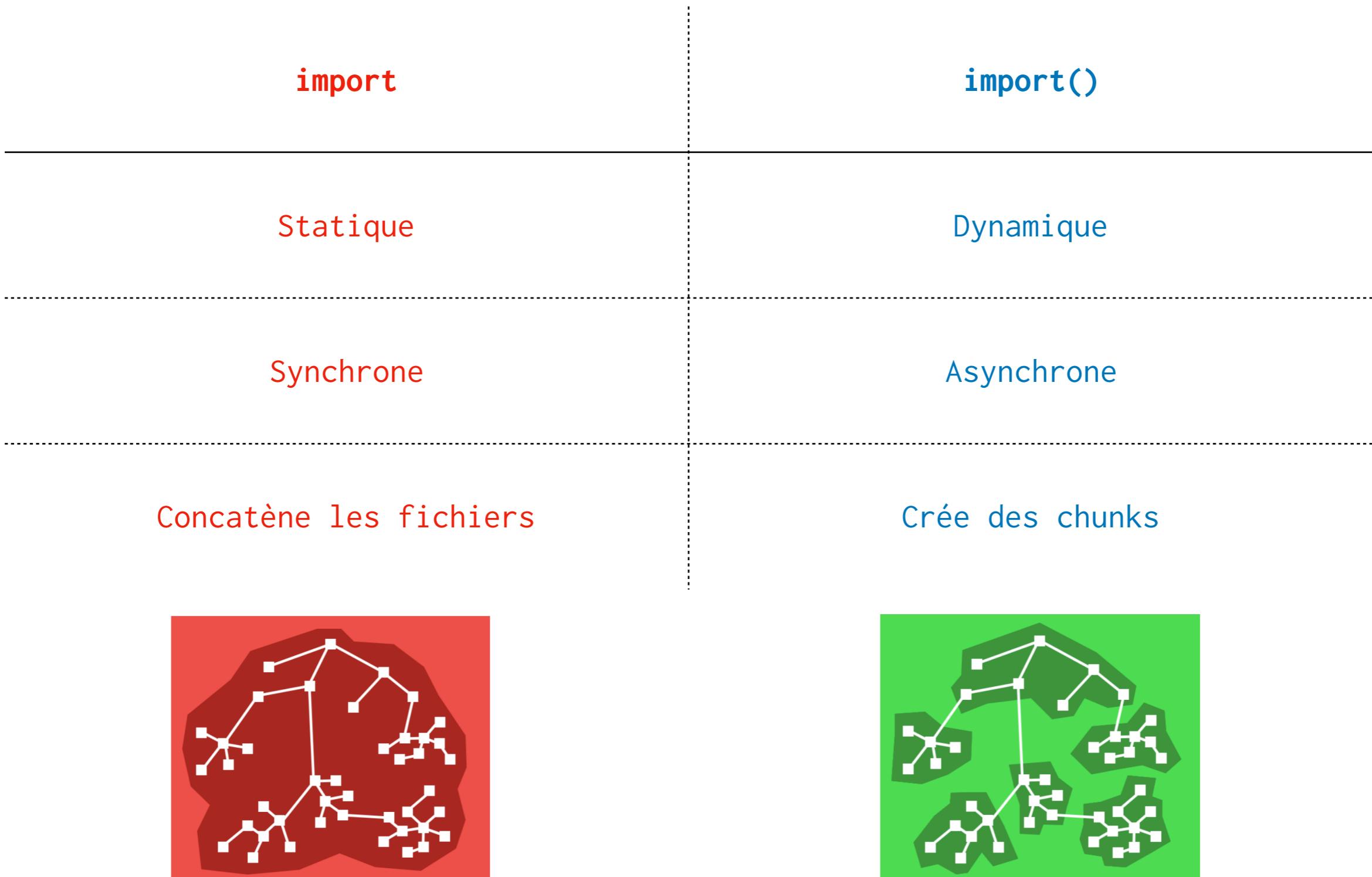
On va découper notre JavaScript en plusieurs chunks, et lorsque l'utilisateur voudra faire une action, on lui enverra seulement le JavaScript nécessaire.

Code Splitting



Source: github.com/jamiebuilds/react-loadable

Code Splitting



Code Splitting par page

```
const Callback = lazy(() => import('./pages/Callback'));
const Invite = lazy(() => import('./pages/Invite'));
const Login = lazy(() => import('./pages/Login'));
const NewProject = lazy(() => import('./pages/NewProject'));
const Problems = lazy(() => import('./pages/Problems'));
const Projects = lazy(() => import('./pages/Projects'));
```

```
const history = createBrowserHistory();
const { persistor, store } = configureStore(history);
@ -29,7 +28,7 @@ const App = () => (
```

```
  <Provider store={store}>
    <PersistGate loading={null} persistor={persistor}>
      <ConnectedRouter history={history}>
```

```
        <Fragment>
          <Suspense fallback={<div>Loading...</div>}>
```

```
            <Toastr />
            <Header />
            <Switch>
```

```
@ -49,7 +48,7 @@ const App = () => (
  <Route exact path="/login" component={Login} />
  <Route component={Login} />
```

```
  </Switch>
</Fragment>
</Suspense>
```

Avant Code Splitting par Page

600 KB

static/js/main.c0b68c45.js

node_modules

highlight.js

lib

languages

mathematica.js

x86asm.js

gauss.js

irpf90.js verilog.js fortran.js julia.js arduino.js excel.js stylus.js

puppet.js

swift.js

processing.js

hy.js

scheme.js

n1ql.js

routeros.js

lasso.js

vim.js

mipsasm.js

rust.js

cs.js

ruby.js

d.js

typescript.js

kotlin.js

matlab.js

avrasm.js

livecodeserver.js

clojure.js

gams.js

mercury.js

xl.js

vhdl.js

coffeescript.js

tcl.js

django.js

erlang.js

stata.js

coq.js

applescript.js

lua.js

gradle.js

basic.js

haxe.js

stan.js

rib.js

groovy.js

crmsh.js

ocaml.js

purebasic.js

glsl.js

coq.js

applescript.js

lua.js

gradle.js

basic.js

haxe.js

stan.js

rib.js

groovy.js

crmsh.js

ocaml.js

autoit.js

hsp.js

objective.js

bash.js

elixir.js

scala.js

pony.js

go.js

xml.js

vala.js

css.js

cal.js

maxima.js

lsl.js

cpp.js

llvm.js

delphi.js

opencl.js

sharp.js

monkey.js

ceylon.js

yaml.js

ini.js

proto.js

http.js

dos.js

sml.js

aspects.js

vbscript.js

qml.js

powershell.js

perl.js

qml.js

aspects.js

vbscript.js

dos.js

sml.js

aspects.js

qml.js

aspects.js

vbscript.js

dos.js

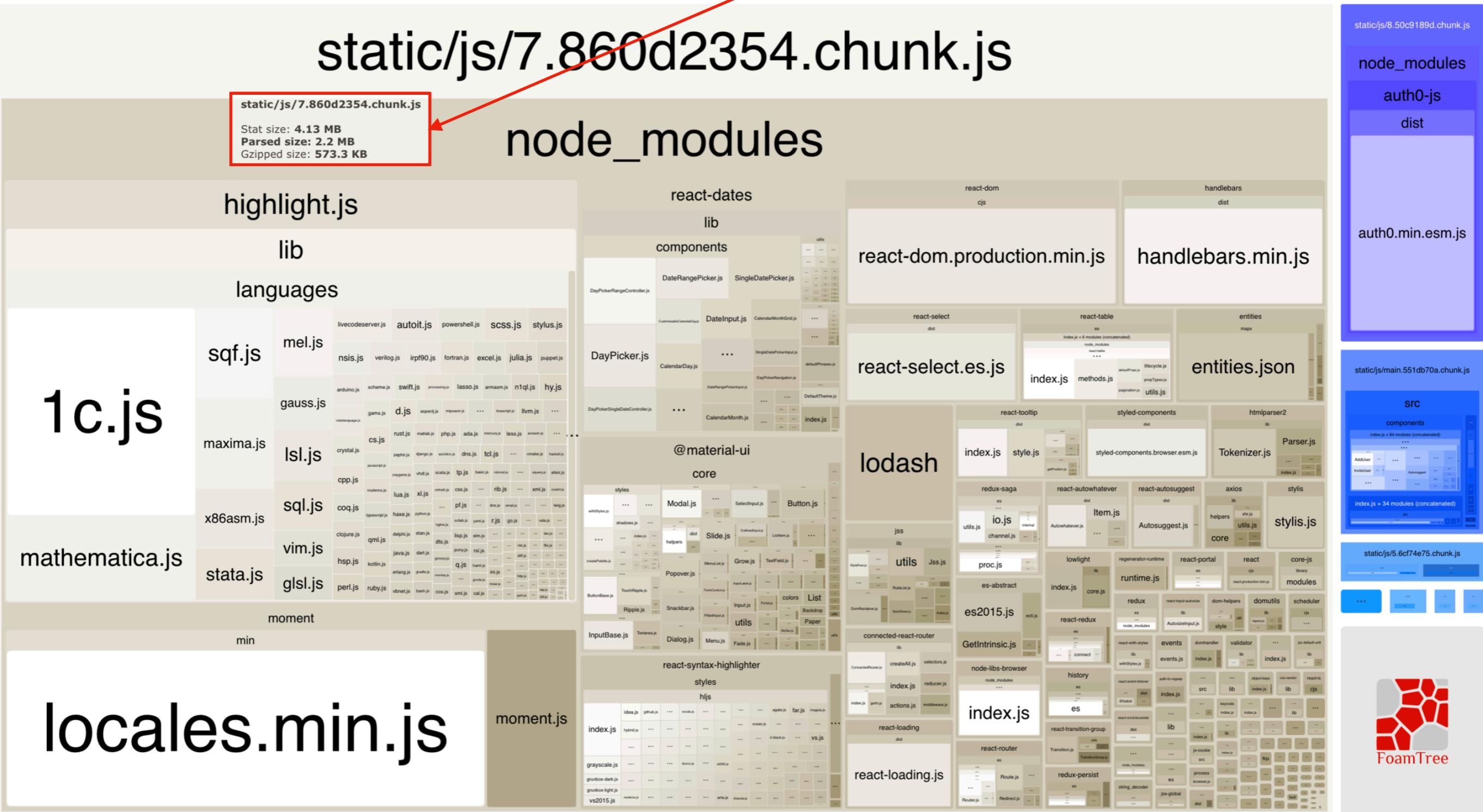
sml.js

aspects.js

vbscript.js

Après Code Splitting par Page

570 KB (-30 KB)



highlight.js

static/js/7.860d2354.chunk.js

static/js/7.860d2354.chunk.js

Stat size: **4.13 MB**
Parsed size: **2.2 MB**
Gzipped size: **573.3 KB**

node_modules

highlight.js

lib

languages

sqf.js

mel.js

gauss.js

maxima.js

isl.js

sql.js

x86asm.js

coq.js

clojure.js

vim.js

mathematica.js

stata.js

moment

min

1c.js

locales.min.js

moment.js

react-dates

lib

components



@material-ui

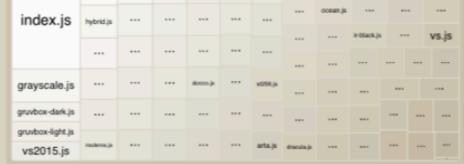
core



react-syntax-highlighter

styles

hljs



react-dom

cjs

react-dom.production.min.js

handlebars

dist

handlebars.min.js

react-select

dist

react-select.es.js

react-table

es

index.js methods.js lifecycle.js propTypes.js pagination.js utils.js

entities

maps

entities.json

lodash

lib

index.js style.js getValues.js

redux-saga

es

io.js internal Autowrapper.js ...

react-autowhatever

dist

Item.js Autowrapper.js ...

react-autosuggest

dist

Autosuggest.js ...

axios

ib

helpers xhr.js utils.js core ...

stylis

stylist.js

Tokenizer.js index.js ...

htmlparser2

lib

index.js styled-components.browser.esm.js

react-tooltip

dist

index.js style.js getValues.js

styled-components

dist

index.js ...

runtime.js

es

index.js core.js regenerator-runtime ...

react-portal

es

index.js react ...

react

cjs

index.js react.production.min.js

core-js

library

index.js modules ...

dom-helpers

ib

index.js dom-helpers ...

domutilis

ib

index.js domutilis ...

scheduler

cjs

index.js scheduler ...

static/js/8.50c9189d.chunk.js

node_modules

auth0-js

dist

auth0.min.esm.js

static/js/main.551db70a.chunk.js

src

components

index.js + 84 modules (concatenated)

AddUser.js ...

InstaUser.js ...

... Autologin.js ...

... ...

... ...

... ...

... ...

... ...

... ...

... ...

... ...

... ...

... ...

... ...

... ...

... ...

... ...

... ...

... ...

... ...

... ...

... ...

... ...

... ...



FoamTree

highlight.js n'est utilisé que dans une modale

Problem solving

Antoine Kahn-Dubois



PICK OTHER
PROJECT

Date Owner Problem with impact

05/08  First label Second la...
I failed rgrrvvbg

06/08  First label
I cannot solve my problems

Markdown export

I failed rgrrvvbg

Hypothèses de cause :

1. liste
2. très
3. numérotée

italique

gras

[link](lol)

`code` fgb

Actions : (responsable Antoine - 27/07/2018)

h1
h2
h3
h4
h5
h6

> blockquote

texte sur plusieurs

COPY TO CLIPBOARD

Expected result	Check date	Result	Status
test a a a	03/04	✓	

I can solve my problems	28/07	✓	
-------------------------	-------	---	--



On Code Split la modale

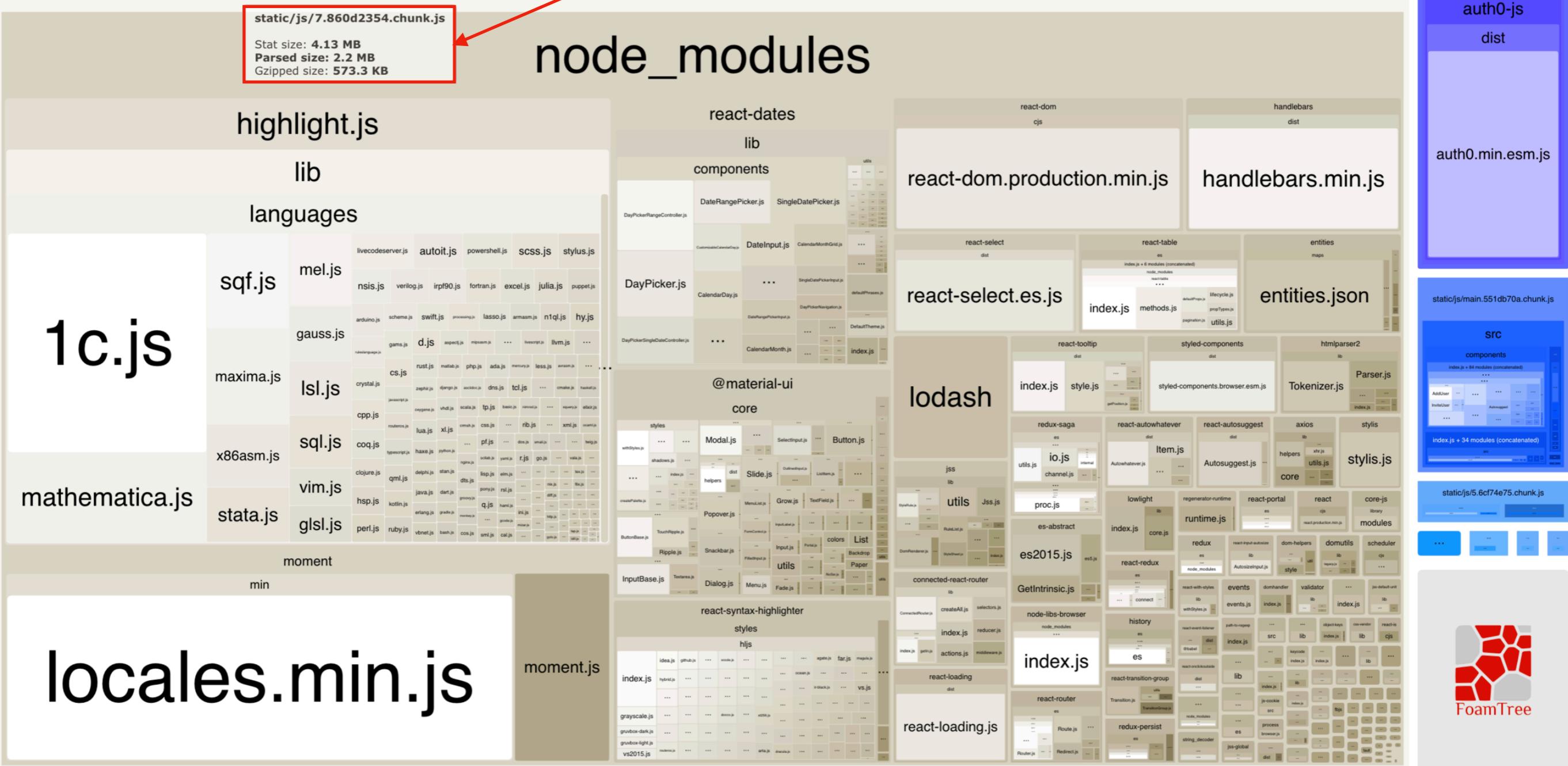
```
const DailyMailExportModal = lazy(() => import('../DailyMailExportModal'));
```

```
type PropsType = {
    problem: ProblemType,
}@ -93,11 +95,15 @@ class ProblemTableCellAction extends Component<PropsType, State> {
    onClick={this.handleOpen}
    />
    <ReactTooltip place="left" type="info" effect="solid" id="paste" />
    <DailyMailExportModal
        problem={problem}
        onClose={this.handleClose}
        open={this.state.isModalOpened}
    />
    {this.state.isModalOpened && (
        <Suspense fallback={null}>
            <DailyMailExportModal
                problem={problem}
                onClose={this.handleClose}
                open={this.state.isModalOpened}
            />
        </Suspense>
    )}
}
```

Avant le code split de la modale

570 KB

static/js/7.860d2354.chunk.js

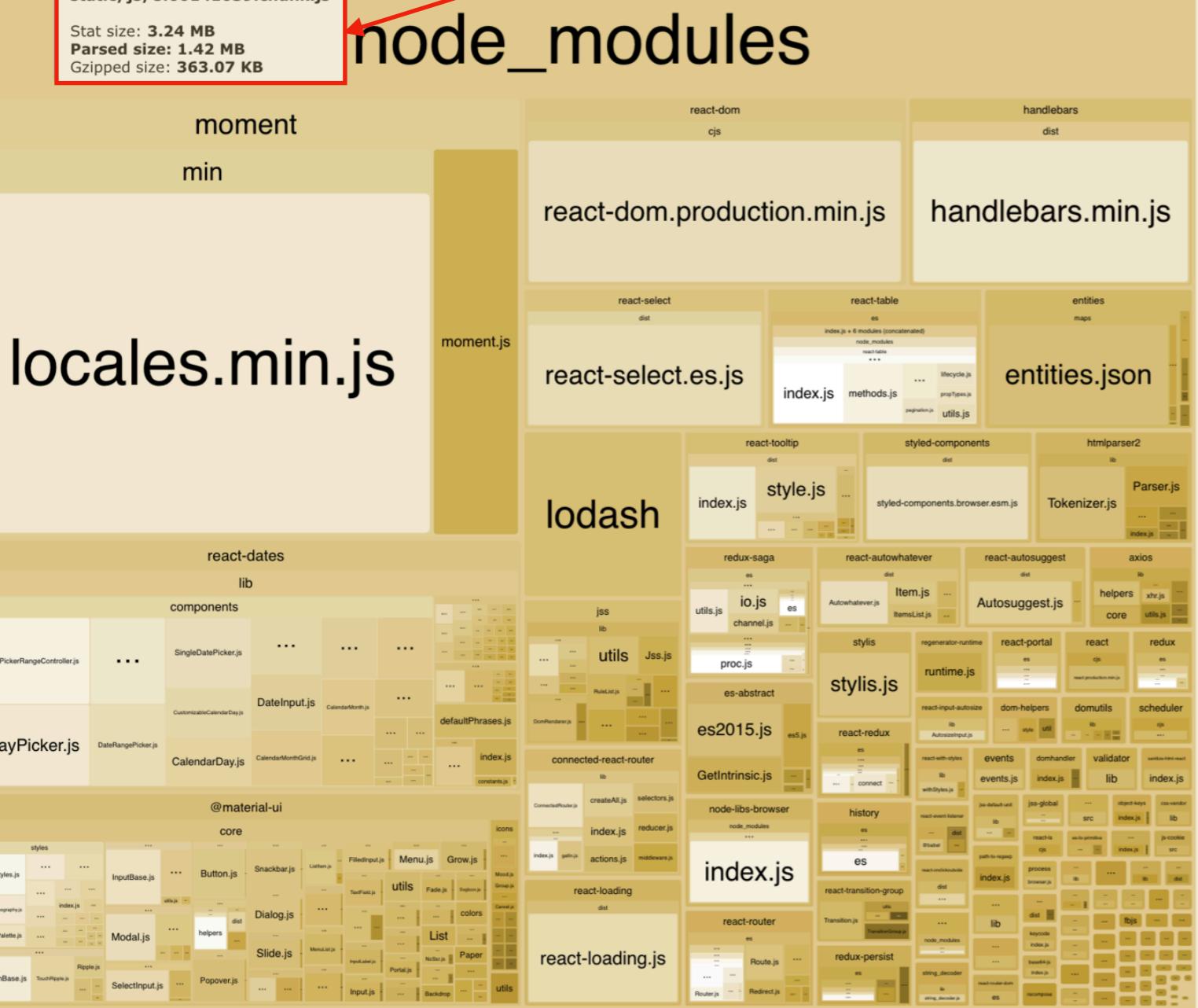


Après le code split de la modale

360 KB (-200 KB)

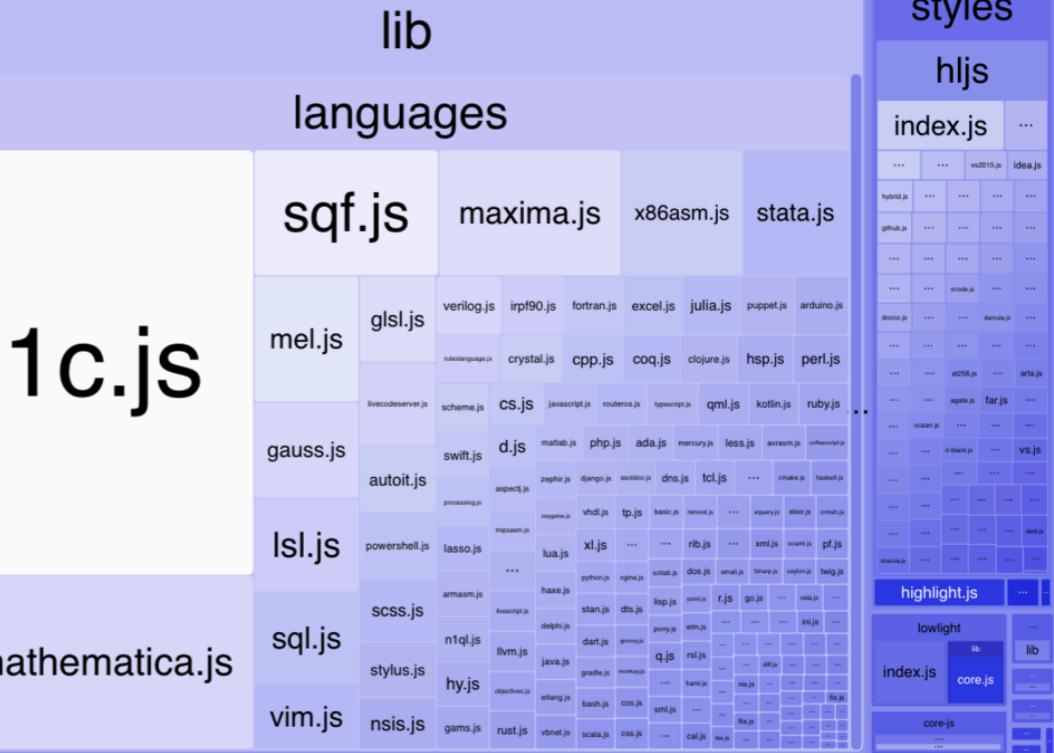
static/js/8.60141039.chunk.js

Stat size: 3.24 MB
 Parsed size: 1.42 MB
 Gzipped size: 363.07 KB



static/js/9.b0f1e469.chunk.js

node_modules
 highlight.js



static/js/10.639ccb8d.chunk.js

node_modules

auth0.min.esm.js



moment / highlight

static/js/8.60141039.chunk.js

static/js/8.60141039.chunk.js
Stat size: 3.24 MB
Parsed size: 1.42 MB
Gzipped size: 363.07 KB

node_modules

moment
min

locales.min.js

react-dom.production.min.js handlebars.min.js

react-select.es.js

entities.json

lodash

react-dates

DayPickerRangeController.js
DayPicker.js
DayPickerRangePicker.js
CustomizableCalendarDay.js
CalendarDay.js
CalendarMonthGrid.js

@material-ui

core
styles
withStyles.js
makeTypeProp.js
createPalette.js
ButtonBase.js
InputBase.js
Button.js
Snackbar.js
FilledInput.js
Menu.js
Grow.js
Icon.js
InputBase.js
index.js
Modal.js
Dialog.js
Slide.js
List.js
Paper.js
Input.js
Ripple.js
SelectInput.js
Popover.js
Popover.js
Backdrop.js
Input.js
Ripple.js

react-loading

react-loading.js

static/js/9.b0f1e469.chunk.js

node_modules

highlight.js

lib

languages

sqf.js maxima.js x86asm.js stata.js

mel.js

gauss.js

lsl.js

sql.js

vim.js

1c.js

mathematica.js

static/js/10.639ccb8d.chunk.js

node_modules

auth0.js

auth0.min.esm.js



moment / highlight

Les **grosses** librairies contiennent beaucoup de fichiers spécifiques.

On importe seulement de ces librairies **les bouts qui nous intéressent**. Pour moment, seulement le français et l'anglais; pour highlight, seulement le markdown.

Après l'allégement de moment/highlight

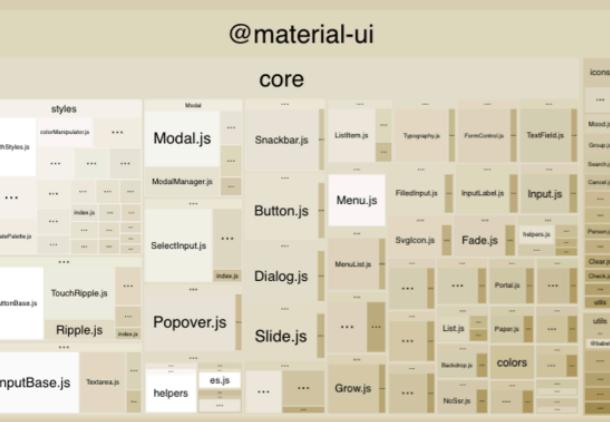
310 KB (-50 KB)

10 KB (-190 KB)

static/js/8.f58616cc.chunk.js

static/js/8.f58616cc.chunk.js
Stat size: 2.92 MB
Parsed size: 1.16 MB
Gzipped size: 313.84 KB

node_modules



handlebars.min.js

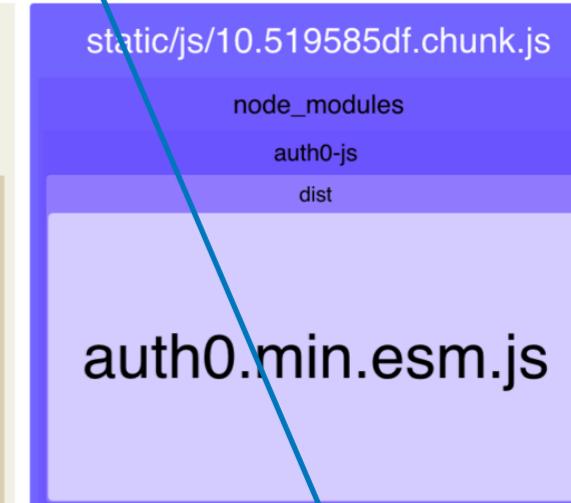
moment

moment.js

entities.json

react-select.es.js

react-dom.production.min.js



auth0.min.esm.js



static/js/9.f9755b18.chunk.js



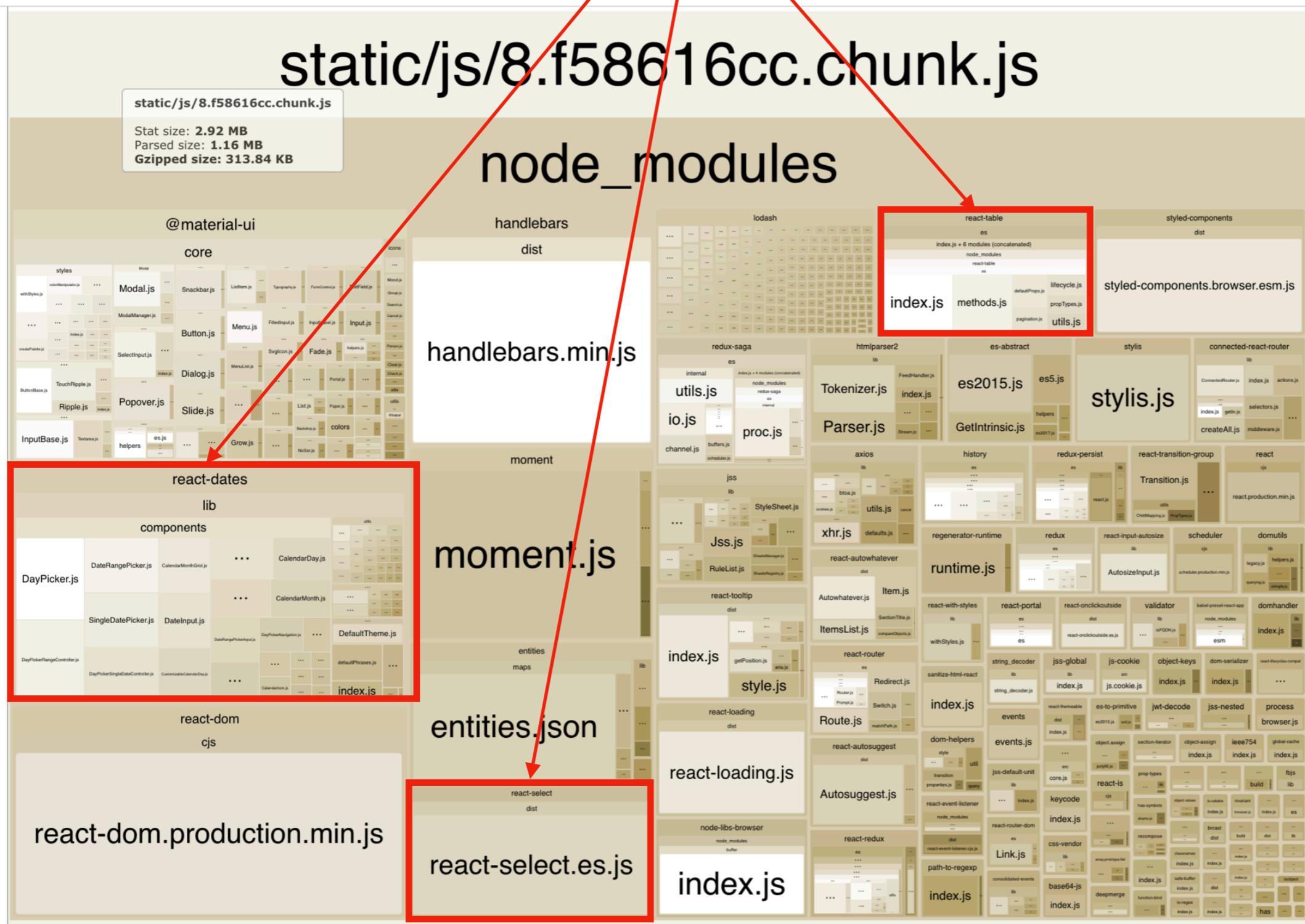
static/js/5.320d1f99.chunk.js



Bizarre...

Librairies utilisées sur une seule page

static/js/8.f58616cc.chunk.js



static/js/10.519585df.chunk.js

node_modules

auth0.js

dist

auth0.min.esm.js

static/js/9.f9755b18.chunk.js

node_modules



static/js/5.320d1f99.chunk.js

node_modules

src



Lorsque j'exporte mes composants en global...

Tree: 9edae94503 ▾

[problem-solving](#) / [client](#) / [src](#) / [components](#) / [index.js](#)



antkahn allow user to change his mail export

2 contributors



19 lines (17 sloc) | 869 Bytes

```
1 // @flow
2
3 export { default as Autosuggest } from './Autosuggest';
4 export { default as Header } from './Header';
5 export { default as Loading } from './Loading';
6 export { default as MailExport } from './MailExport';
7 export { default as PageContent } from './PageContent';
8 export { default as PrivateRoute } from './PrivateRoute';
9 export { default as ProblemTable } from './ProblemTable';
10 export { default as ProblemTableCell } from './ProblemTableCell';
11 export {
12     default as ProblemTableCellDatePicker,
13 } from './ProblemTableCellDatePicker';
14 export { default as ProblemTableCellOwner } from './ProblemTableCellOwner';
15 export { default as ProblemTableCellResult } from './ProblemTableCellResult';
16 export { default as ProfilePicture } from './ProfilePicture';
17 export { default as ProjectMembers } from './ProjectMembers';
18 export { default as UserCard } from './UserCard';
```

J'importe mes composants en global !

```
// @flow

import React, { Component } from 'react';

import type { Match } from 'react-router-dom';
import type { Action } from '../../../../../types/redux';
import { Loading } from '../../../../../components';

export type PropsType = {
    inviteUser: (token: string) => Action,
    match: Match,
    push: (pathname: string) => Action,
};

export default class Invite extends Component<PropsType> {
    props: PropsType;
```

J'importe seulement les composants dont j'ai besoin

2 ████ client/src/pages/Invite/Invite.component.js

```
diff --git a/client/src/pages/Invite/Invite.component.js b/client/src/pages/Invite/Invite.component.js
@@ -4,7 +4,7 @@ import React, { Component } from 'react';
4   4
5   5   import type { Match } from 'react-router-dom';
6   6   import type { Action } from '../../types/redux';
7 - 7   - import { Loading } from '../../../../../components';
7 + 7   + import Loading from '../../../../../components>Loading';
8   8
9   9   export type PropsType = {
10 10     inviteUser: (token: string) => Action,
```

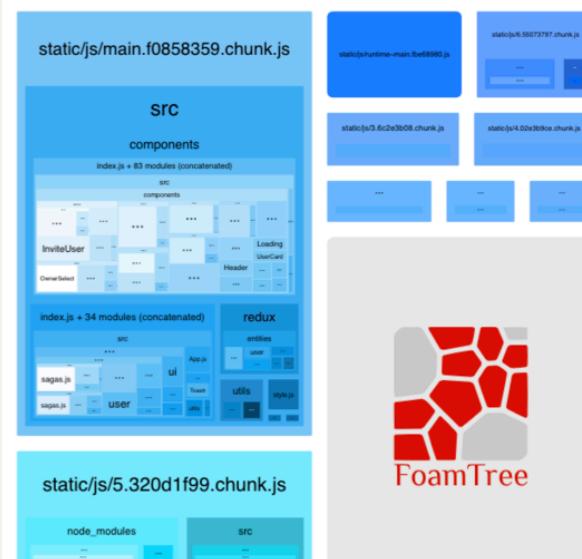
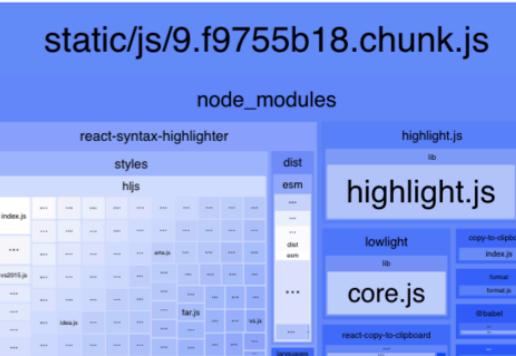
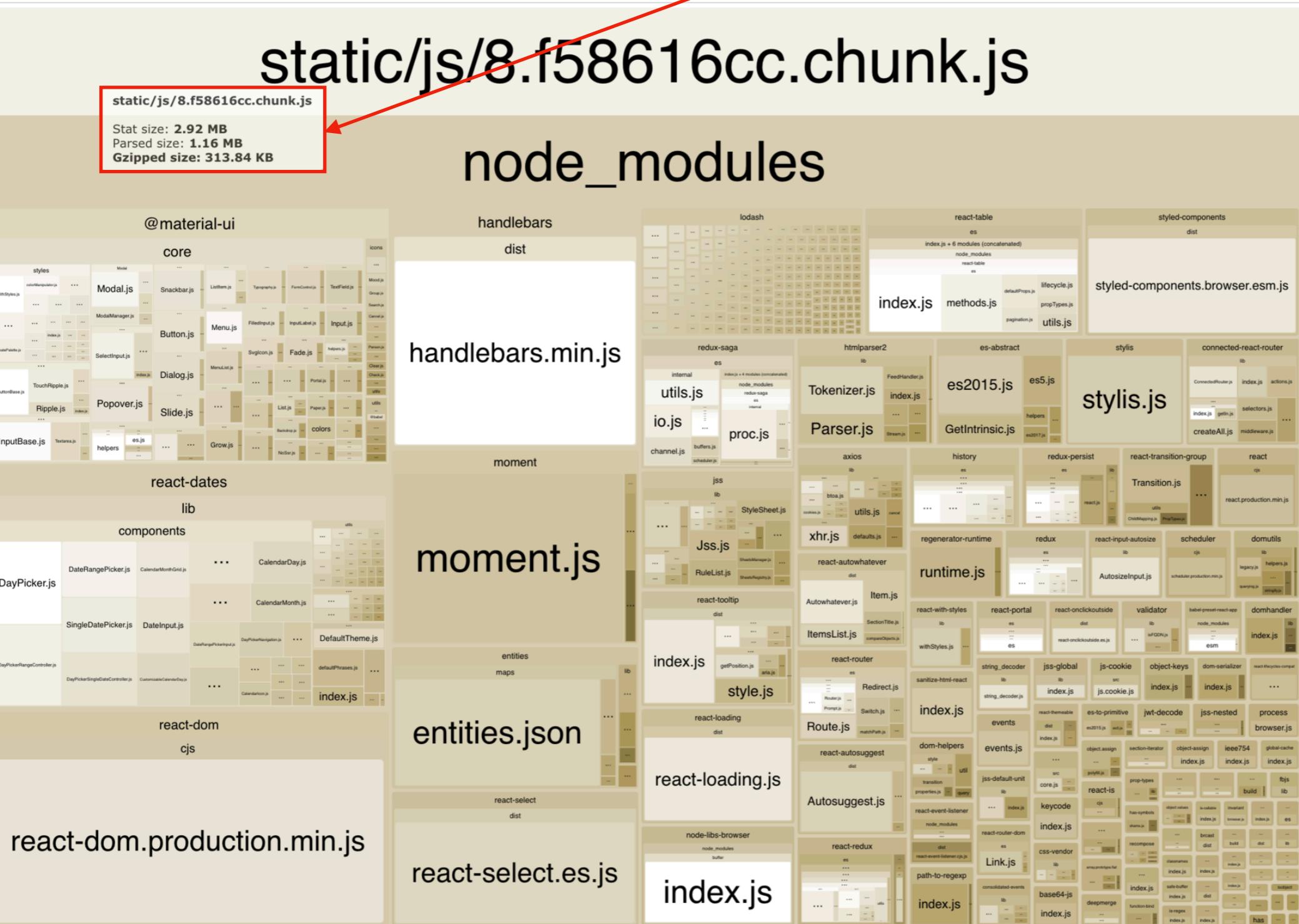


Avant la suppression des imports globaux

310 KB

~~static/js/8.f58616cc.chunk.js~~

static/js/8.f58616cc.chunk.js
Stat size: 2.92 MB
Parsed size: 1.16 MB
Gzipped size: 313.84 KB

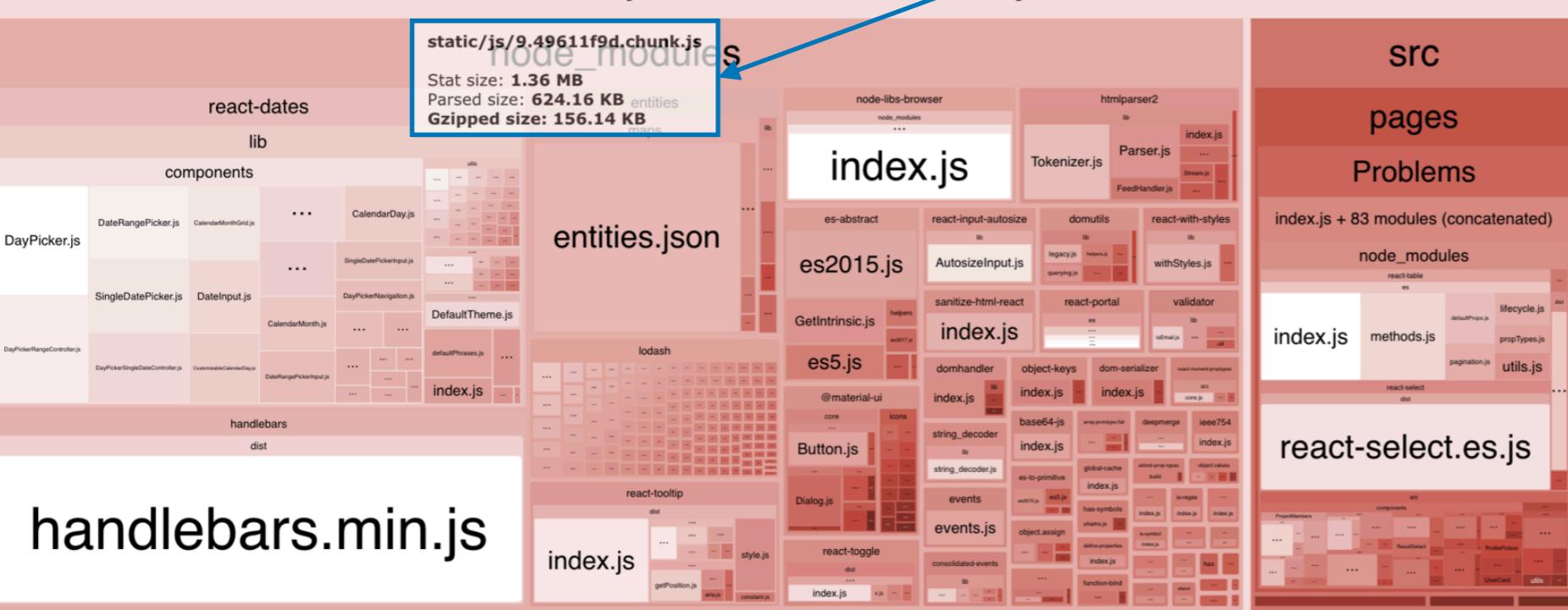


Après la suppression des imports globaux

150 KB (-160 KB)

static/js/9.49611f9d.chunk.js

node_modules
Stat size: 1.36 MB
Parsed size: 624.16 KB entities
Gzipped size: 156.14 KB



handlebars.min.js

static/js/12.a2a59edc.chunk.js

node_modules

moment
moment.js

react-dom

cjs

react-dom.production.min.js

static/js/14.708e5714.chunk.js

node_modules
auth0.js
dist

auth0.min.esm.js

static/js/2.e1d228e1.chunk.js

node_modules
@material-ui
react-autowhatever

static/js/13.9e54f761.chunk.js

node_modules

react-syntax-highlighter

styles

highlight.js

dist

esm

core.js

lowlight

highlight.js

Sur quels composants utiliser le code splitting ?

Sur quels composants utiliser le code splitting ?

Dès que l'élément **n'est pas visible** à l'arrivée sur la page:

- Les autres pages
- Les modales
- Les date pickers
- Les composants non affichés dans le viewport

A photograph of a group of people, likely of African descent, standing outdoors. They are wearing traditional, colorful clothing, including various patterns of robes, shawls, and headwraps. The colors range from earthy tones like browns and yellows to more vibrant blues and reds. Some individuals have beaded necklaces or bracelets. The background is a plain, light-colored wall.

On split le reste...

Avant Code Splitting

150 KB

static/js/9.49611f9d.chunk.js

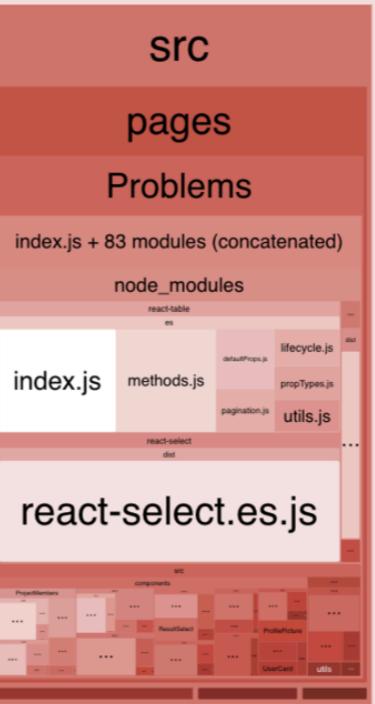
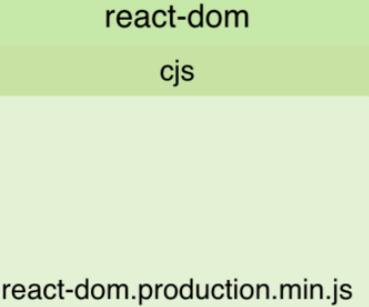
node_modules
Stat size: 1.36 MB
Parsed size: 624.16 KB entities
Gzipped size: 156.14 KB



static/js/12.a2a59edc.chunk.js

node_modules

moment.js



static/js/13.9e54f761.chunk.js

node_modules

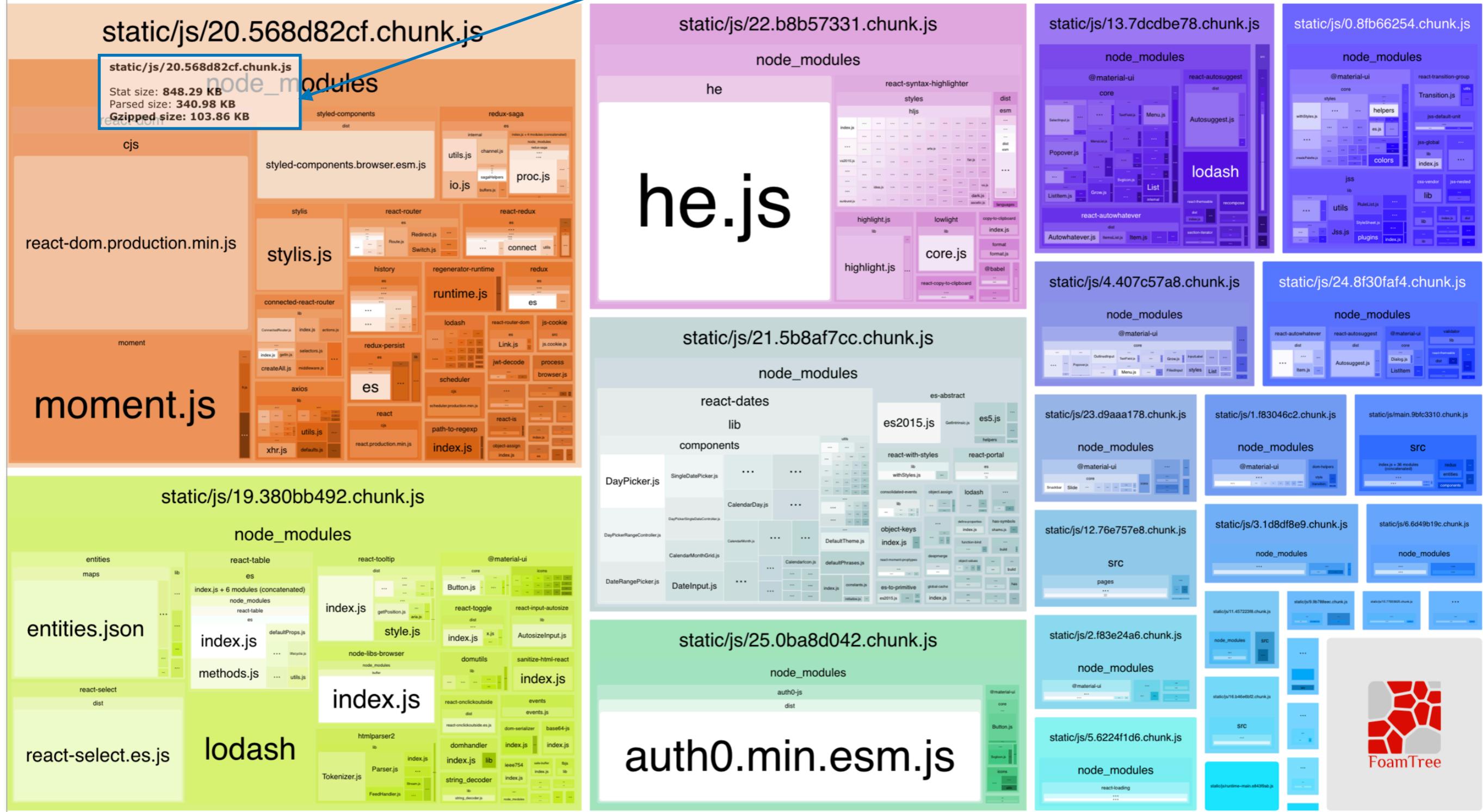
static/js/0.303d381e.chunk.js

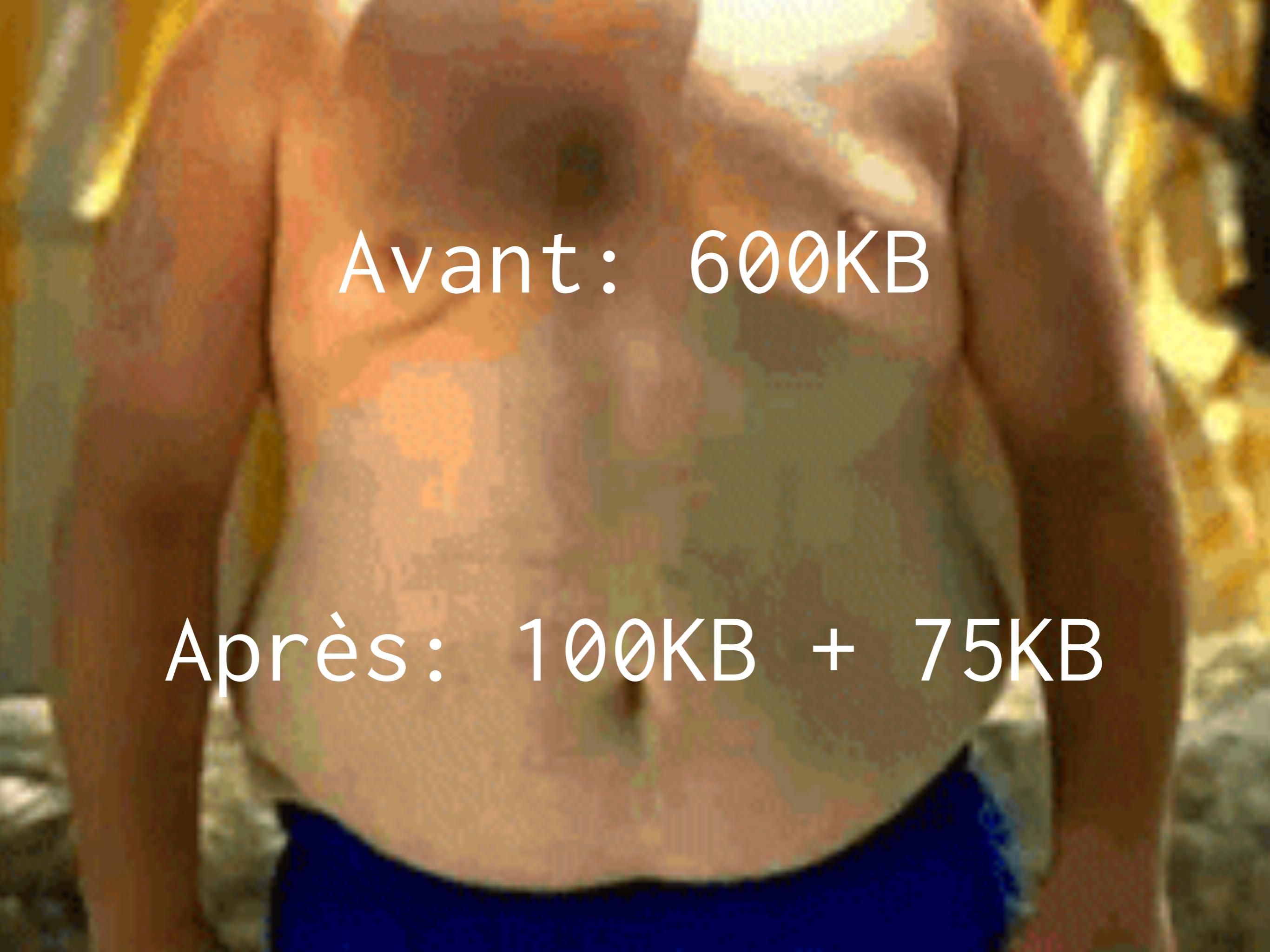
static/js/3.e96ae25f.chunk.js



Après Code Splitting

100 KB (-50KB)



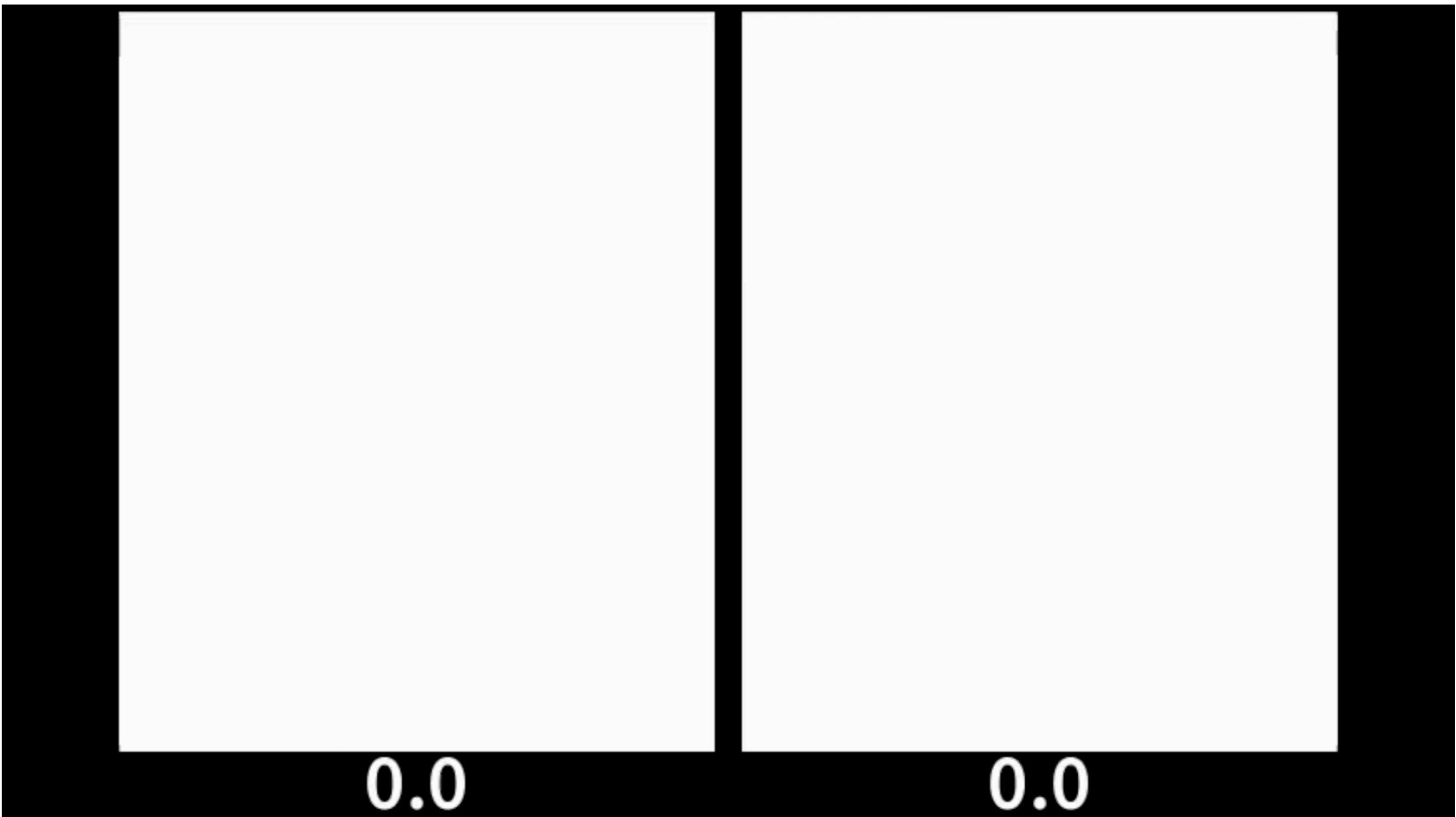


Avant: 600KB

Après: 100KB + 75KB

Comparaison (page principale)

Ipad, iOS12, 3G



Comparaison (page login)

Desktop, Chrome, 3G

ba62dd1b.ngrok.io/

ba62dd1b.ngrok.io/

0.0

0.0

Au menu

- Pourquoi diminuer la taille de son JavaScript ?
- Comment mettre en place le Code Splitting
- **Comment choisir ses librairies ?**
- Pour aller plus loin: le Bundle Splitting

Comment choisir la meilleure
librairie de graphes ?

bundlephobia.com

BUNDLEPHOBIA

BADGES OPENCOLLECTIVE SCAN PACKAGE.JSON ^B



echarts@4.3.0



A powerful charting and visualization library for browser

1 dependency



BUNDLE SIZE

784.7 kB

MINIFIED

261.3 kB

MINIFIED + GZIPPED

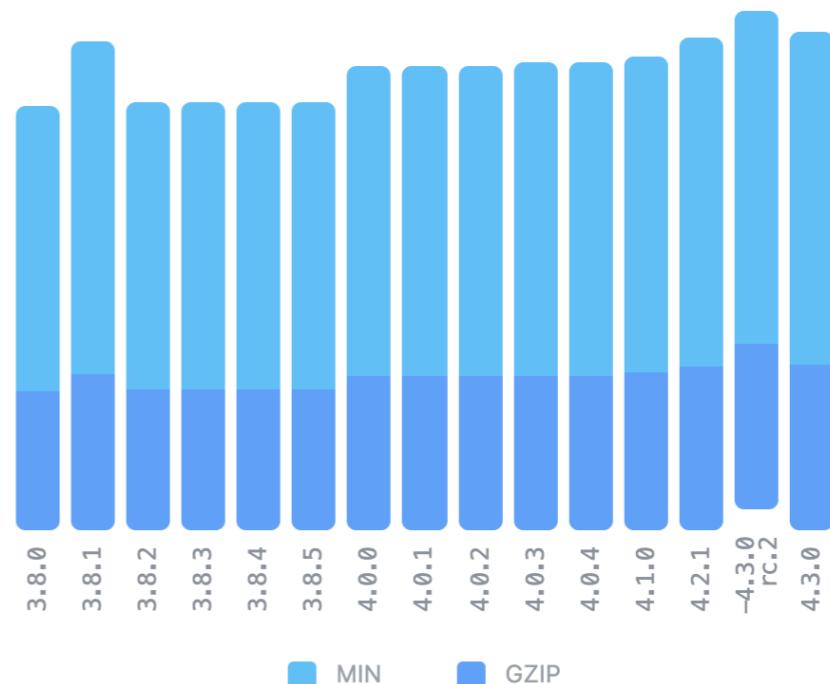
DOWNLOAD TIME

8.71s

2G EDGE ¹

5.23s

EMERGING 3G ¹



Composition

Composition



Permet de voir si la librairie utilise des dépendances que vous avez déjà sur votre projet

Meilleur choix ?

Similar Packages

General purpose Charting libraries

d3



Data-Driven Documents

70 %
SMALLER **79.36 kB**
MIN + GZIP



chart.js



Simple HTML5 charts using the
canvas element.

57 %
SMALLER **112.06 kB**
MIN + GZIP



frappe-charts



<https://frappe.github.io/charts>

93 %
SMALLER **17.10 kB**
MIN + GZIP



highcharts



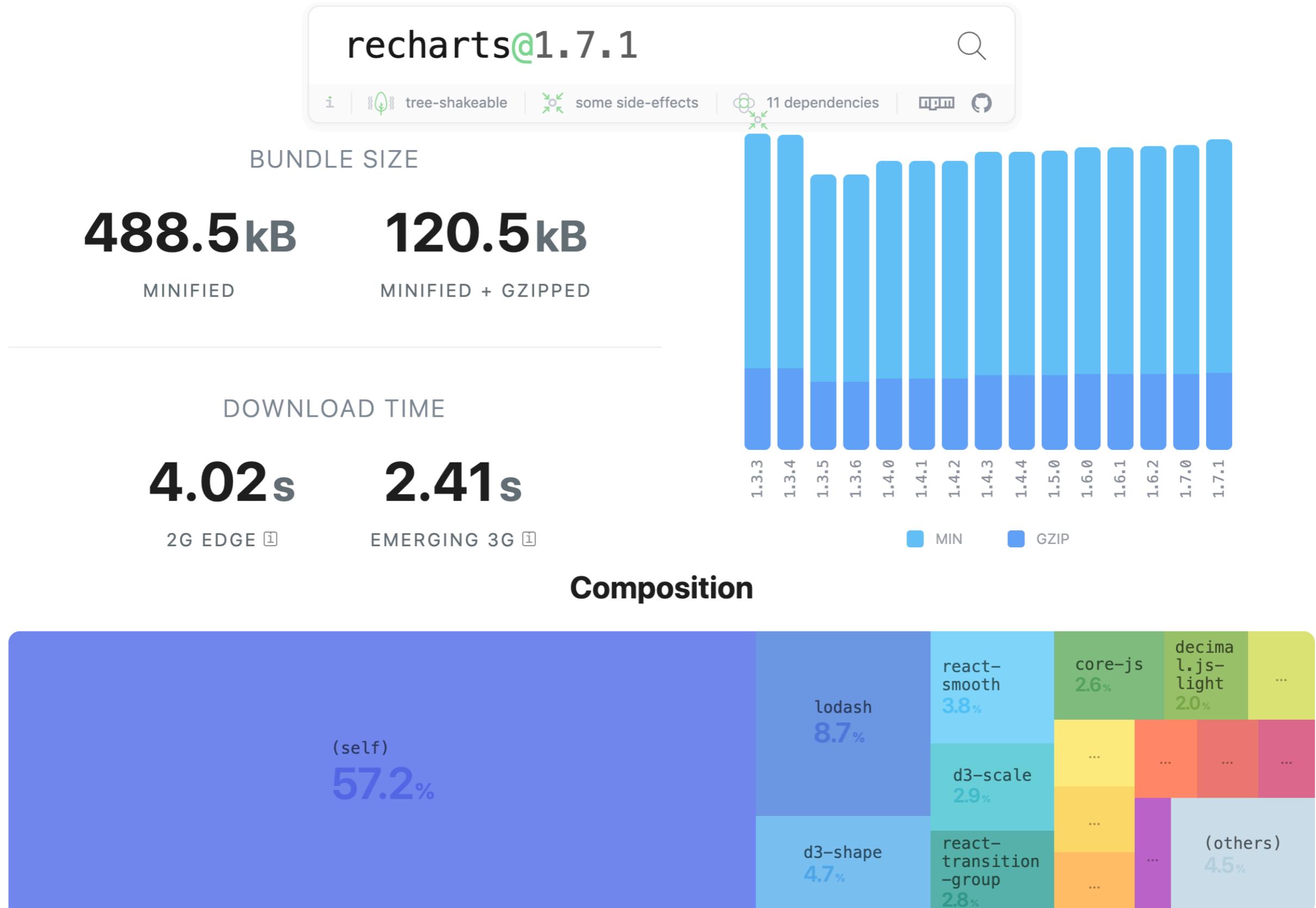
JavaScript charting framework

70 %
SMALLER **79.57 kB**
MIN + GZIP



SUGGEST ANOTHER

Mon choix



Mon choix

Tree shakeable



BUNDLE SIZE

488.5 kB

MINIFIED

120.5 kB

MINIFIED + GZIPPED

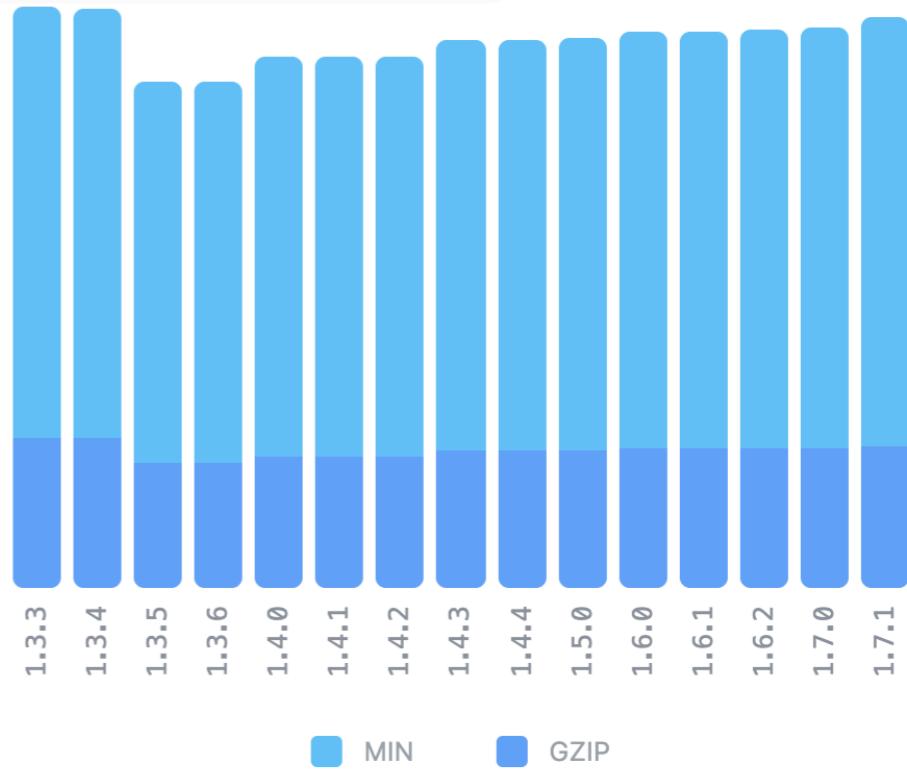
DOWNLOAD TIME

4.02s

2G EDGE ⓘ

2.41s

EMERGING 3G ⓘ



Composition



Mon choix

Tree shakeable



BUNDLE SIZE

488.5 kB

MINIFIED

120.5 kB

MINIFIED + GZIPPED

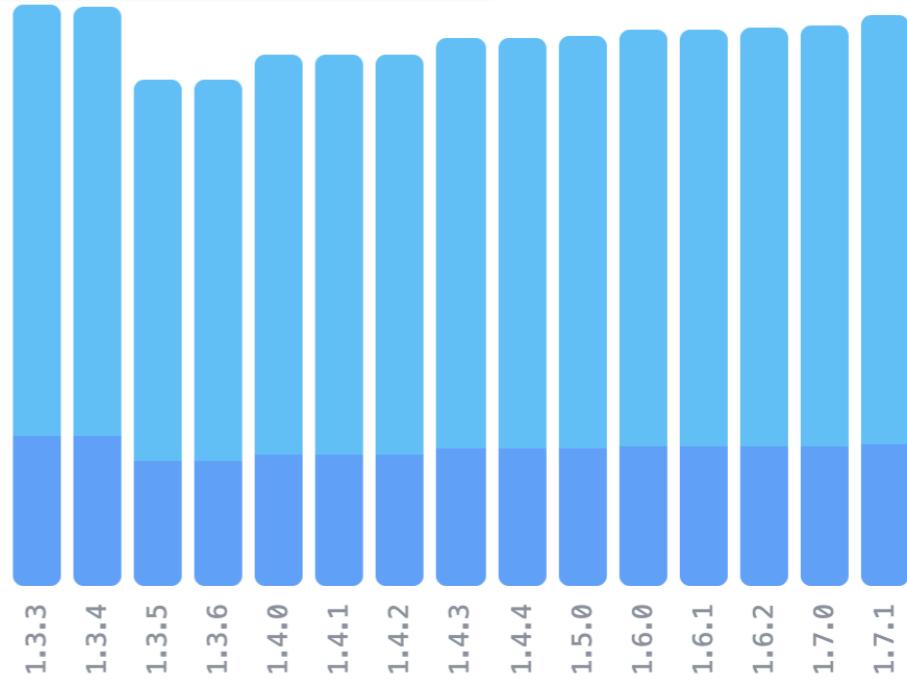
DOWNLOAD TIME

4.02s

2G EDGE ⓘ

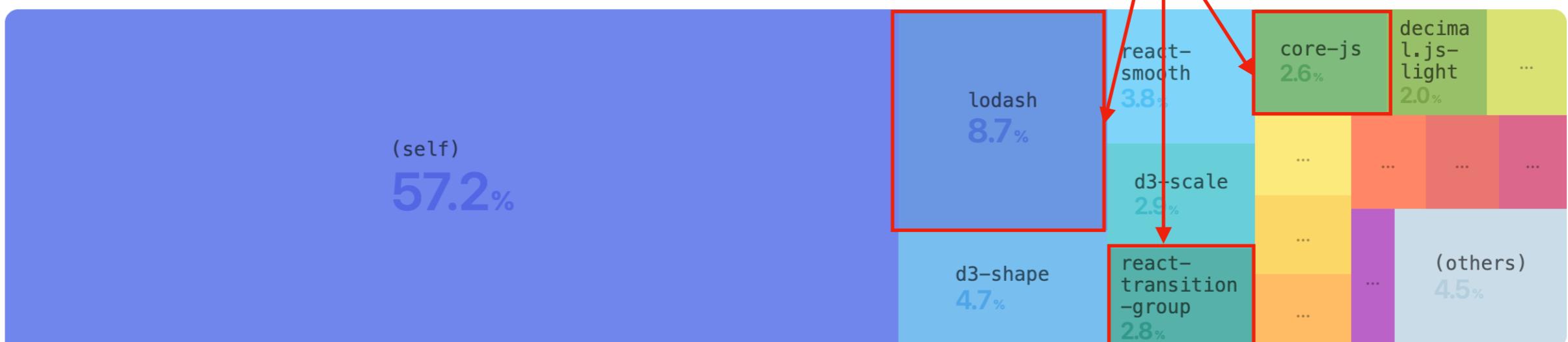
2.41s

EMERGING 3G ⓘ



Composition

Librairies déjà utilisées



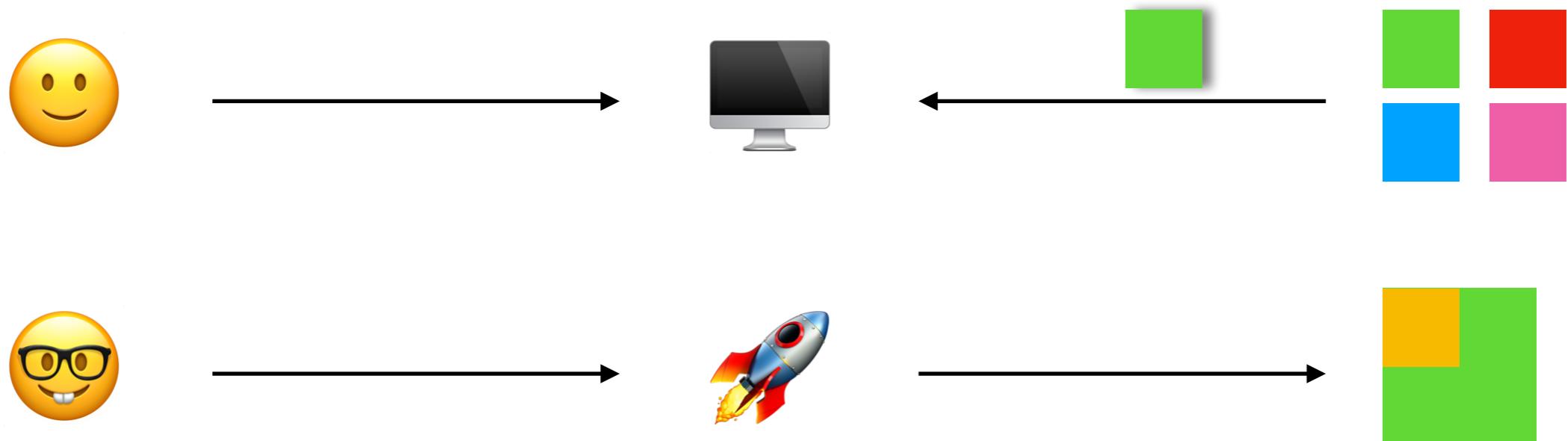
Au menu

- Pourquoi diminuer la taille de son JavaScript ?
- Comment mettre en place le Code Splitting
- Comment choisir ses librairies ?
- Pour aller plus loin: le Bundle Splitting

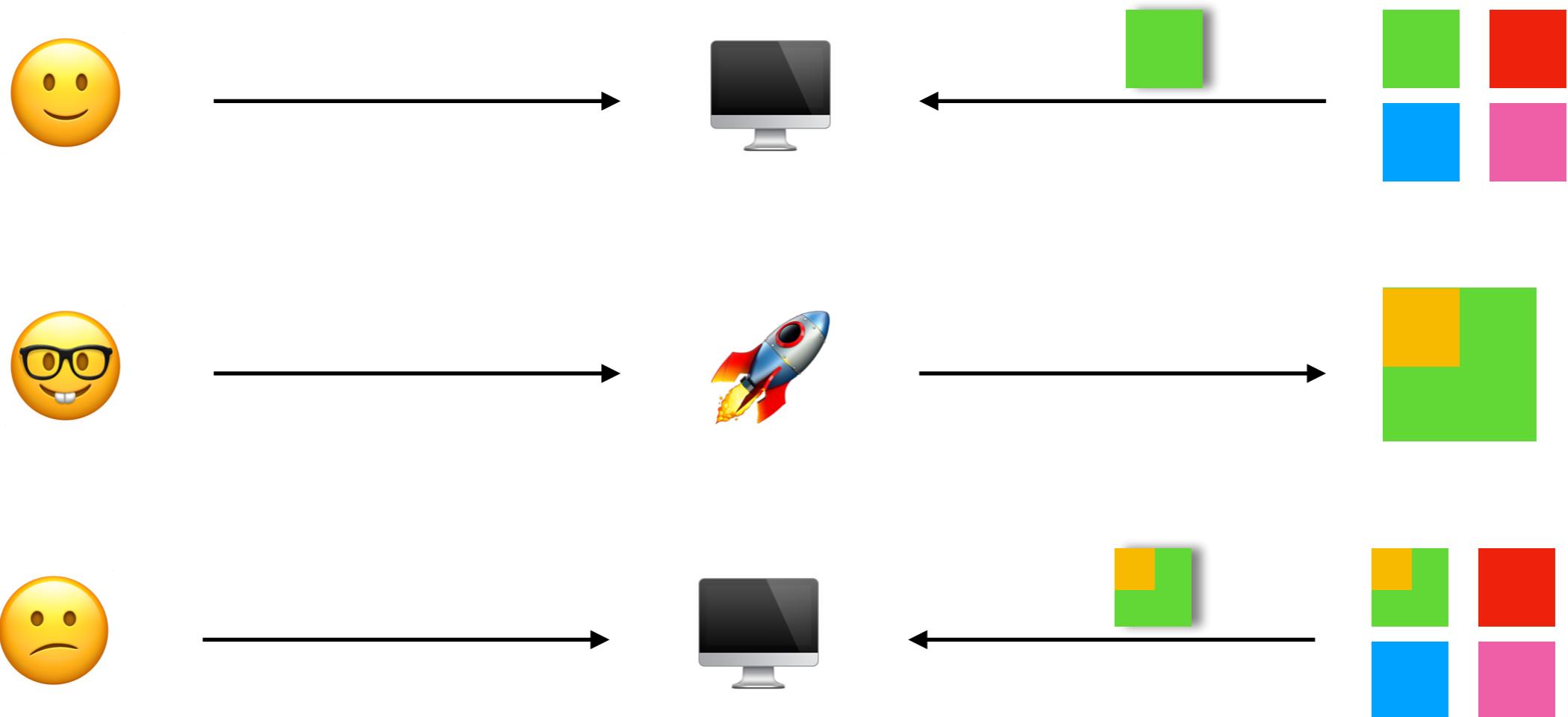
Bundle Splitting



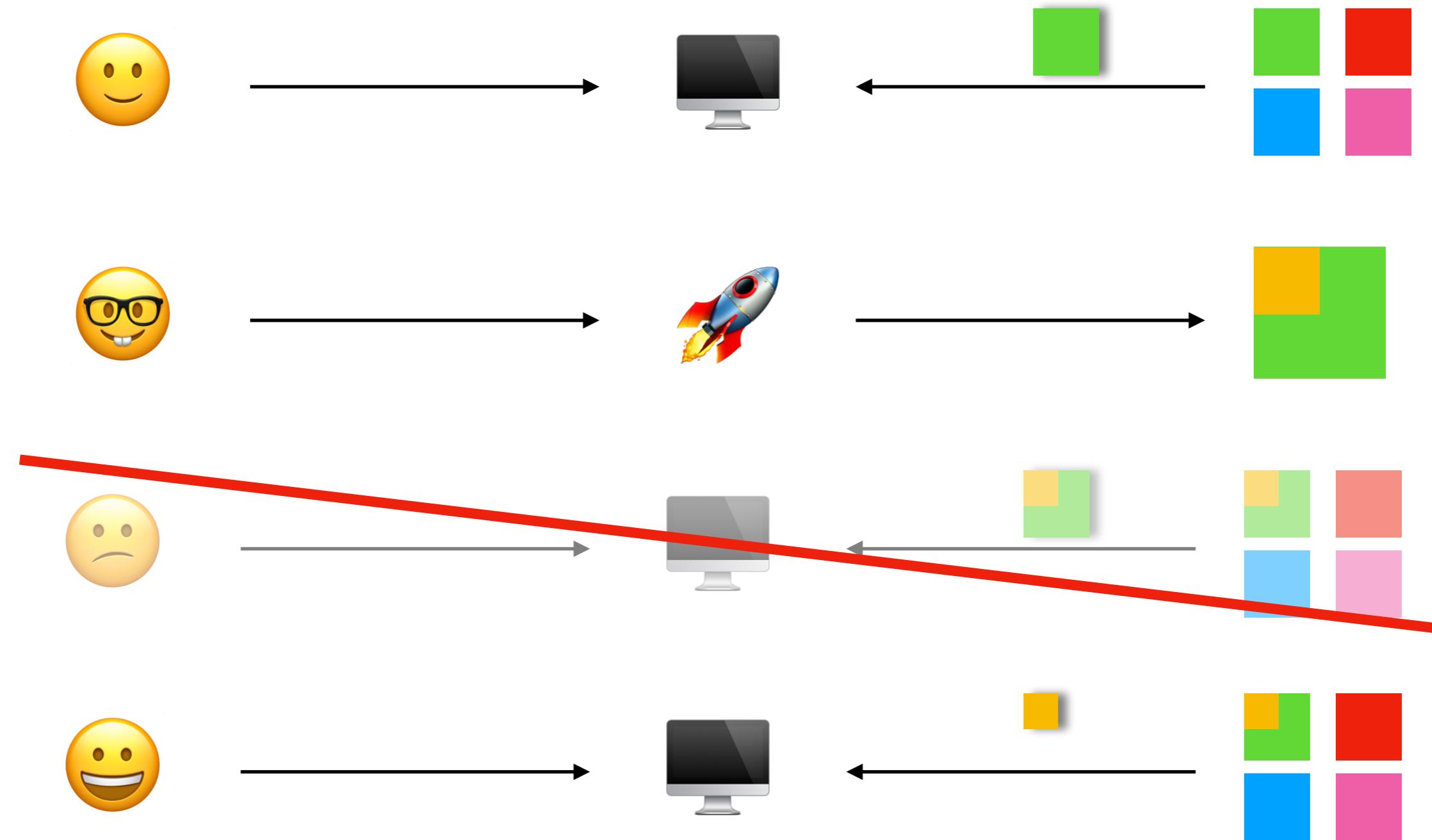
Bundle Splitting



Bundle Splitting



Bundle Splitting



Bundle Splitting

- Le **nom** d'un chunk dépend de son **contenu**

Bundle Splitting

- Le **nom** d'un chunk dépend de son **contenu**
- Un chunk déjà téléchargé va rester dans le **cache navigateur** de l'utilisateur

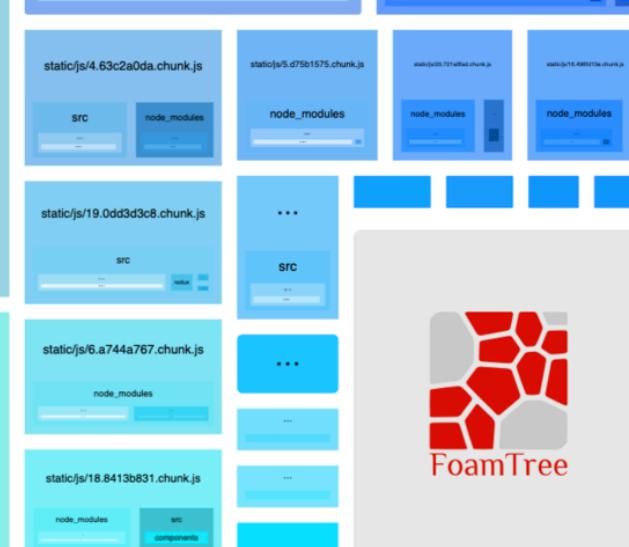
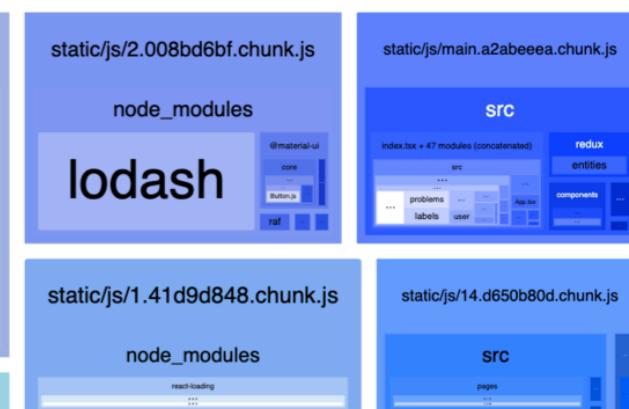
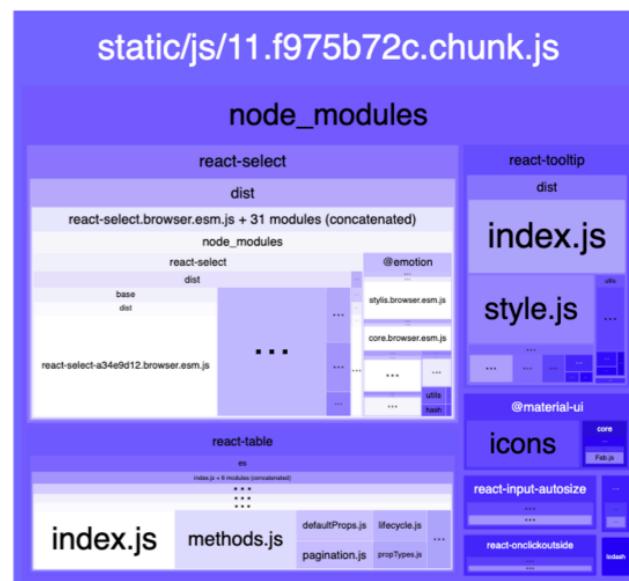
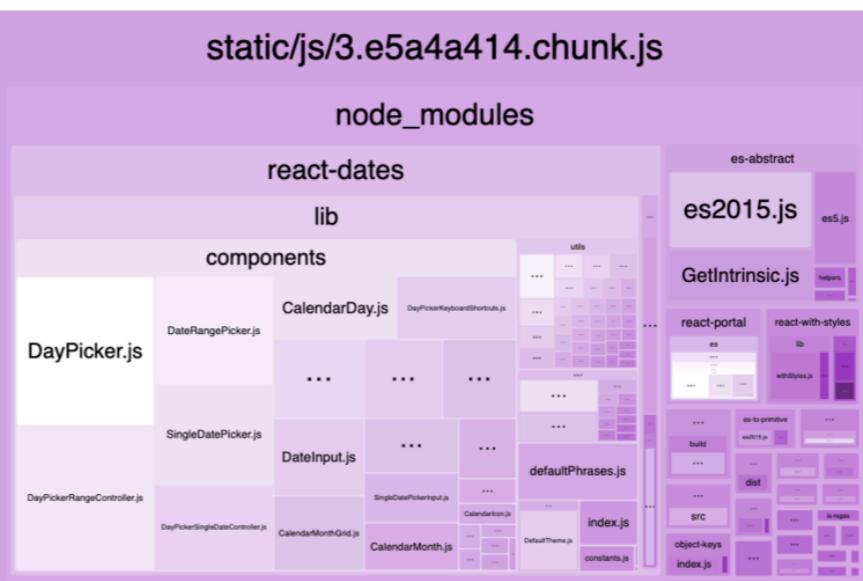
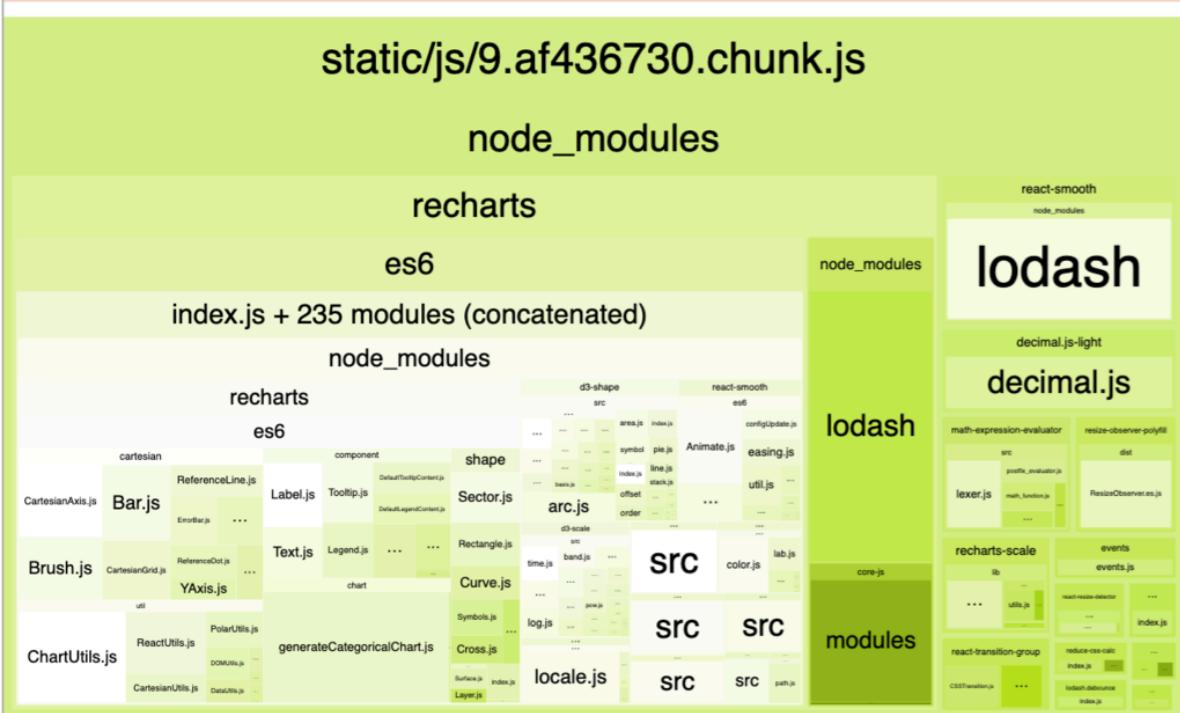
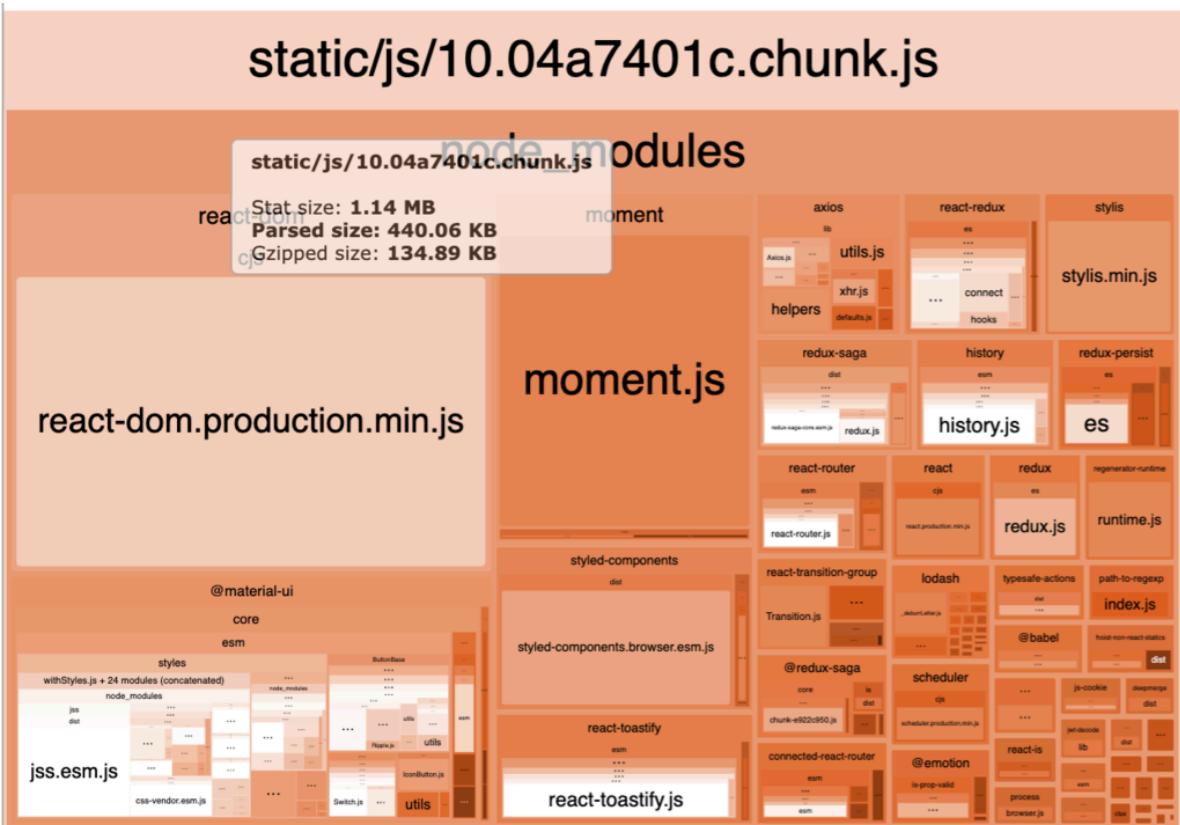
Bundle Splitting

- Le **nom** d'un chunk dépend de son **contenu**
- Un chunk déjà téléchargé va rester dans le **cache navigateur** de l'utilisateur
- On va créer un chunk par **librairie externe**

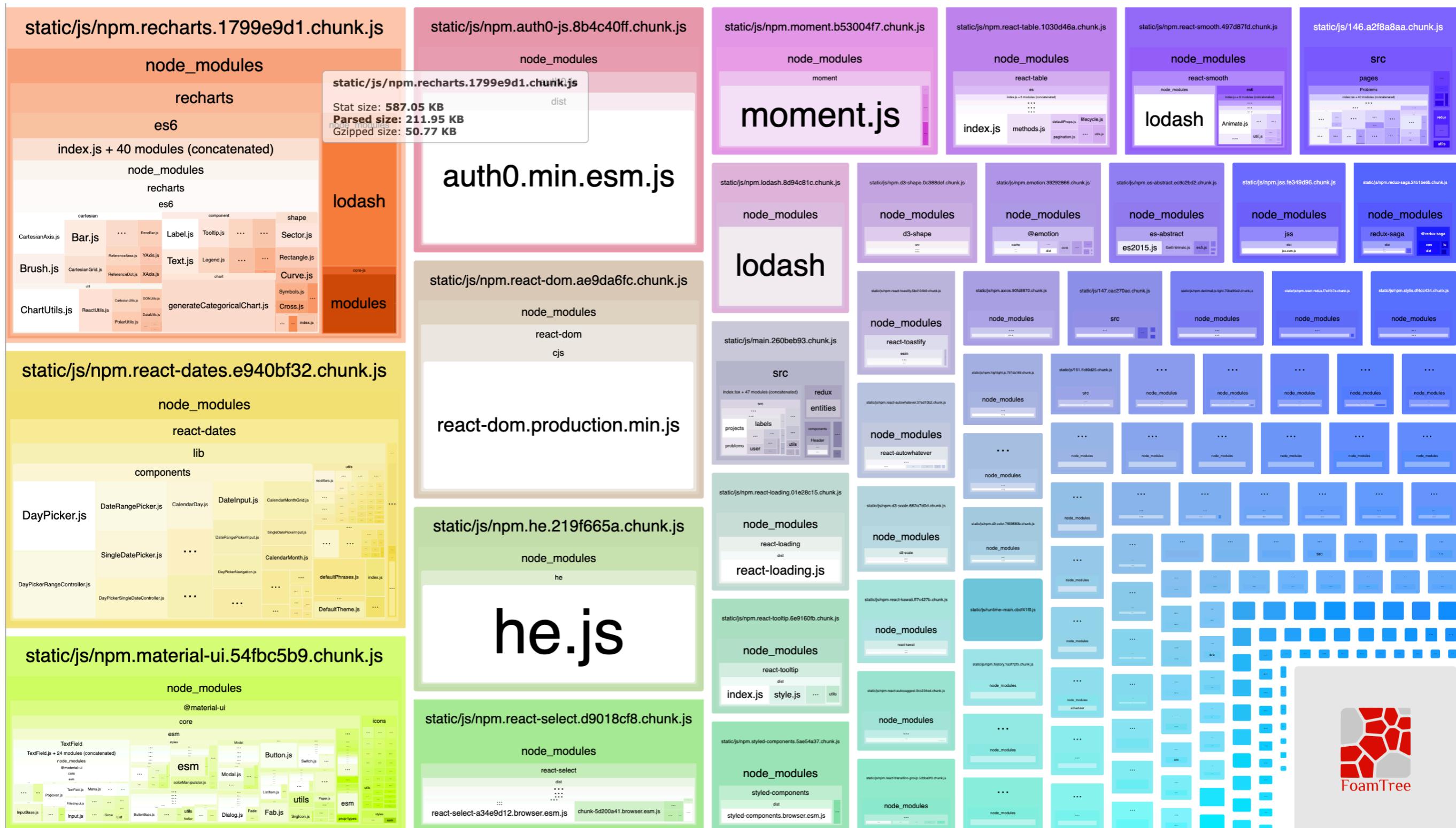
Bundle Splitting

- Le **nom** d'un chunk dépend de son **contenu**
- Un chunk déjà téléchargé va rester dans le **cache navigateur** de l'utilisateur
- On va créer un chunk par **librairie externe**
- Cela nécessite **HTTP/2**, car on va faire une **cinquantaine de requêtes** par page

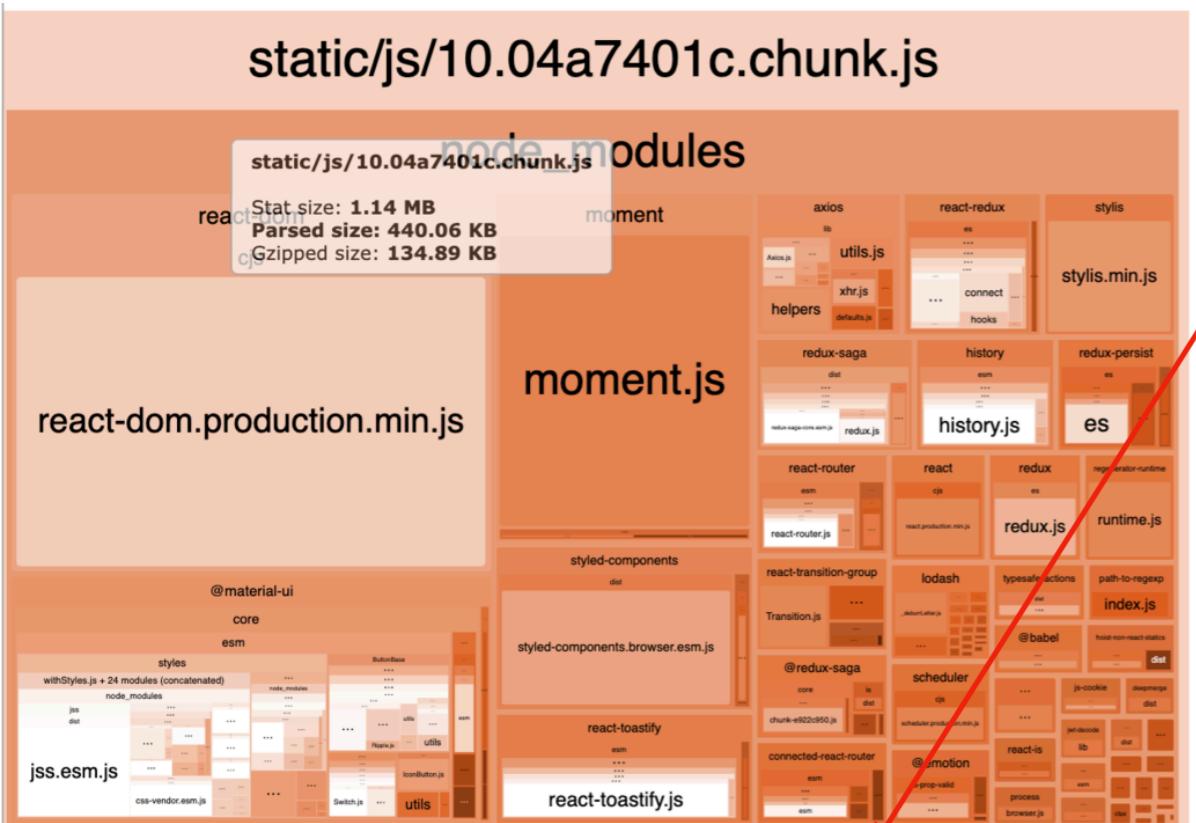
Avant Bundle Splitting



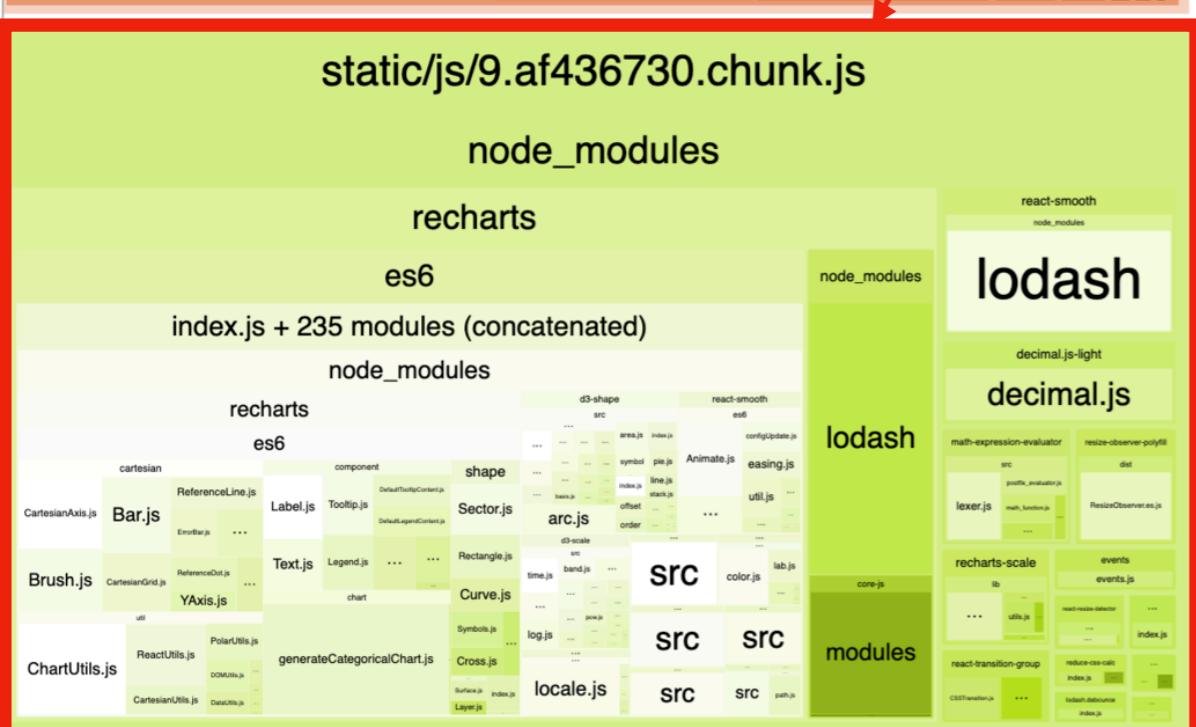
Après Bundle Splitting



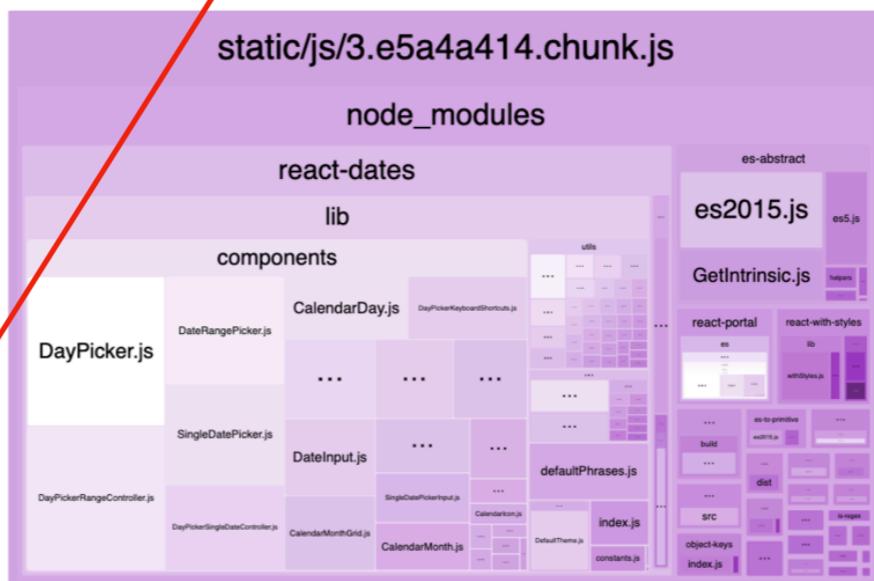
Modification de code, avant Bundle Splitting



static/js/9.af436730.chunk.js



75KB



static/js/15.25d540c0.chunk.js

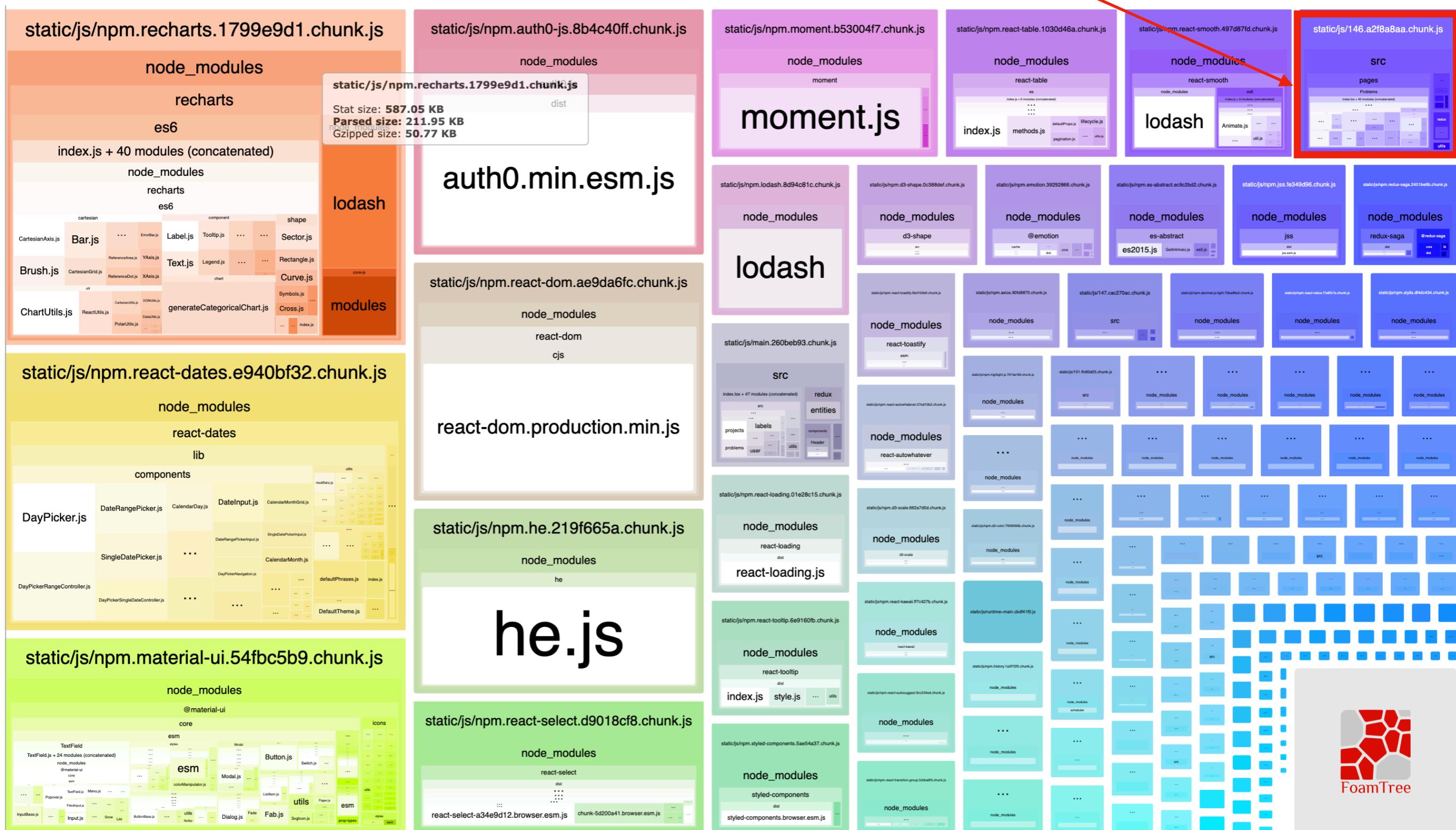


auth0.min.esm.js

he.js

Modification de code, après Bundle Splitting

5KB (-70KB)



On fait le bilan

On fait le bilan

Grâce au Code Splitting: on apporte uniquement à l'utilisateur ce dont il a besoin, quand il en a besoin.

On fait le bilan

Grâce au Code Splitting: on apporte uniquement à l'utilisateur ce dont il a besoin, quand il en a besoin.

Grâce au Bundle Splitting: on apporte uniquement à l'utilisateur ce qui a changé depuis sa dernière visite.

Pour résumer

Pour résumer

- Garder son JavaScript à moins de 200KB pour rendre une page

Pour résumer

- Garder son JavaScript à moins de 200KB pour rendre une page
- Utiliser le Code Splitting pour chaque composant non nécessaire au rendu initial

Pour résumer

- Garder son JavaScript à moins de 200KB pour rendre une page
- Utiliser le Code Splitting pour chaque composant non nécessaire au rendu initial
- Utiliser les imports allégés des grosses librairies

Pour résumer

- Garder son JavaScript à moins de 200KB pour rendre une page
- Utiliser le Code Splitting pour chaque composant non nécessaire au rendu initial
- Utiliser les imports allégés des grosses librairies
- Ne pas exporter globalement ses composants ou fonctions

Pour résumer

- Garder son JavaScript à moins de 200KB pour rendre une page
- Utiliser le Code Splitting pour chaque composant non nécessaire au rendu initial
- Utiliser les imports allégés des grosses librairies
- Ne pas exporter globalement ses composants ou fonctions
- Utiliser le Bundle Splitting si on est en HTTP/2

Pour s'entraîner

J'ai créé un exercice qui reprend tous les gestes que je viens de présenter

Pour s'entraîner



github.com/theodo/code-splitting-dojo



Merci !

Des questions ?