



THE MAGIC OF CONTROLLING ATTENTION TO SHAPE PERCEPTION

GRGUR de GRISOGONO



Sencha Community Days 2019
European Sencha ExtJS Conference

(VERY READABLE SLIDE)

私は日本語が話せません。私はできればいいのに。こんにちは。そして川崎。

私は日本の文化を深く賞賛します。次回訪問するときは、さらに文化を楽しむために全国を見学します。

あなたはそれを見ることができません、しかし私は中笑っています。私の顔を真剣に保つのは本当に難しいです。みんな笑顔になったら、これを読んだことを登録します。私を笑わせて笑わせないでください。

さて、今、私はこれが忙しいスライドのように感じさせるためにたくさんのテキストを追加しなければなりません。

2つの尻が通りを歩きます。おならもう一人は、「あなたは言葉を私の口から出した」とコメントした。

(VERY READABLE SLIDE – TRANSLATED)

I can't speak Japanese. I wish I could, though. I can only say Hello. And Kawasaki.

I deeply admire Japanese culture. Next time I visit I will tour across the country to relish the culture some more.

You can't see it, but I'm laughing inside. It's really hard to keep my face serious. If you all start smiling, I'll register that you've read this. Please don't make me burst laughing!

Ok, now I have to add a bunch of text to make this feel like a busy slide.

Two butts walk down the street. One farted. The other one commented: "You took the words right out of my mouth".



GOAL:
DELIVER A MOBILE WEB VIEW
OF ANY SIZE
OVER 3G
IN 1 SECOND

TIMING FACTORS

1. **D**OMAIN LOOKUP
2. **S**ERVER CONNECTION
3. **S**ERVER RESPONSE
4. **C**ONTENT DOWNLOAD
5. **B**ROWSER EXECUTION

PAGE LOAD UP

1s...10s

BOUNCE RATE UP

123%



100ms

INSTANT



1s

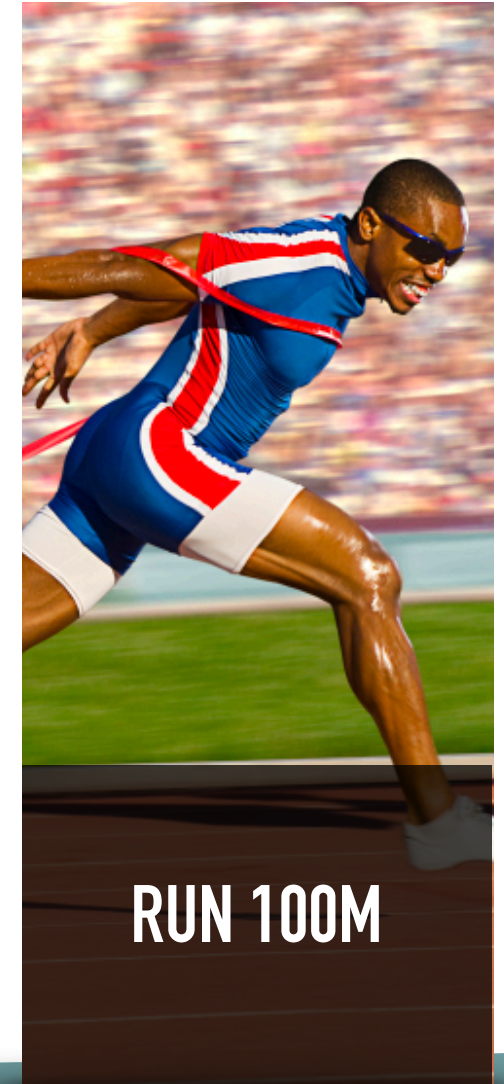
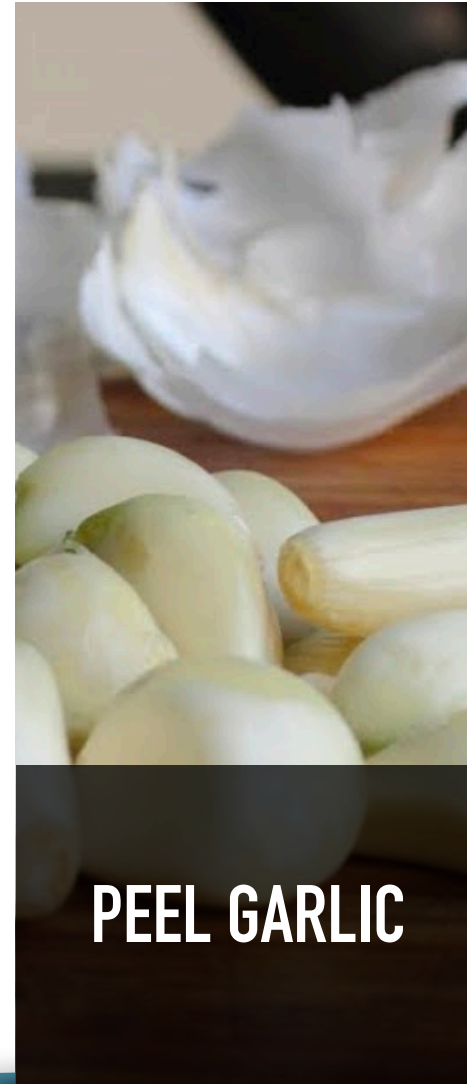
**CONTROL
CHALLENGED**



10s

**FOCUS
LOST**

WHAT OTHERS DO IN 10 SECONDS...



2G-FIRST DEVELOPMENT

LOW
BANDWIDTH

=

TRAIN
ELEVATOR
TUNNEL
BASEMENT

NATURE
AIR
SEA
HIGHWAY



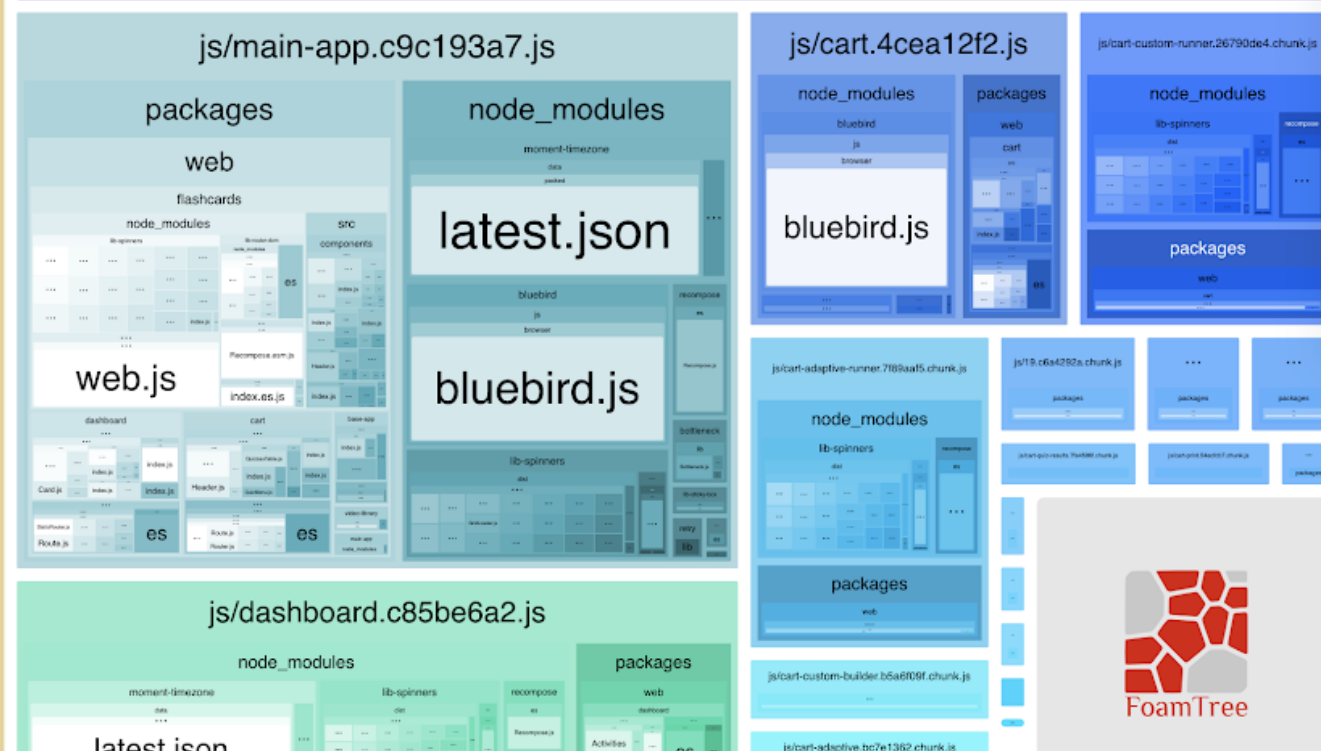
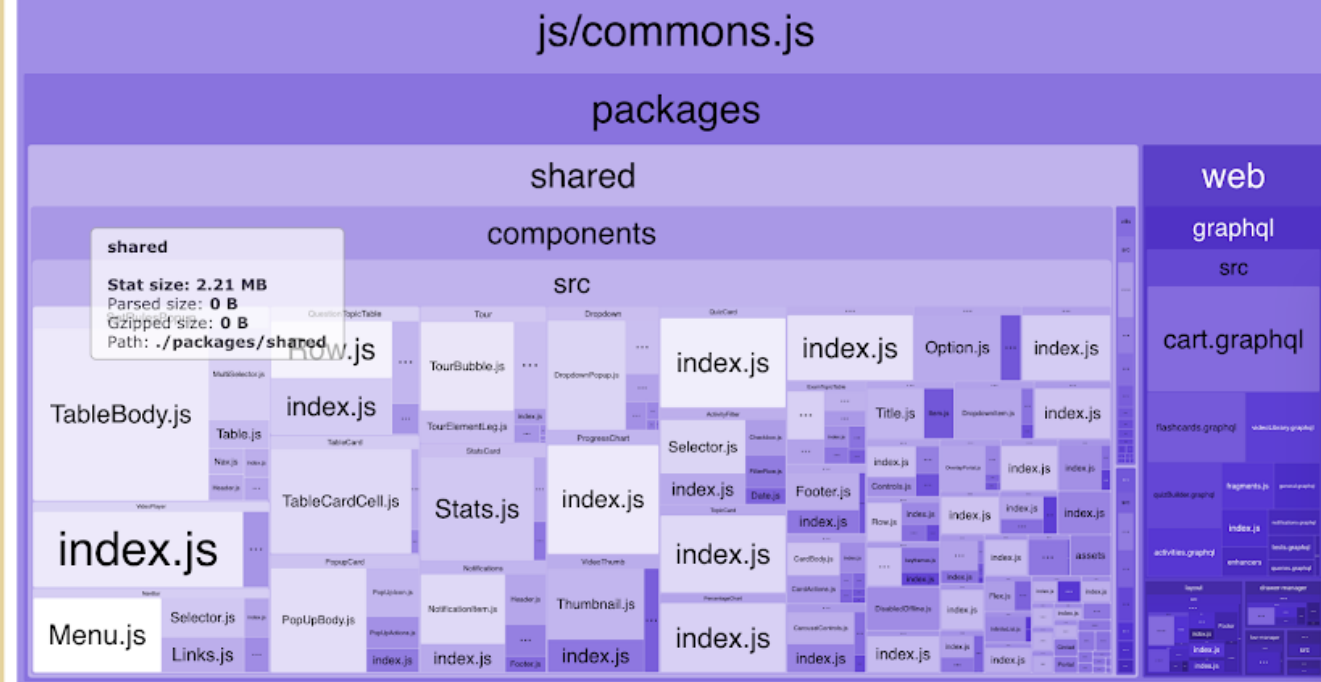
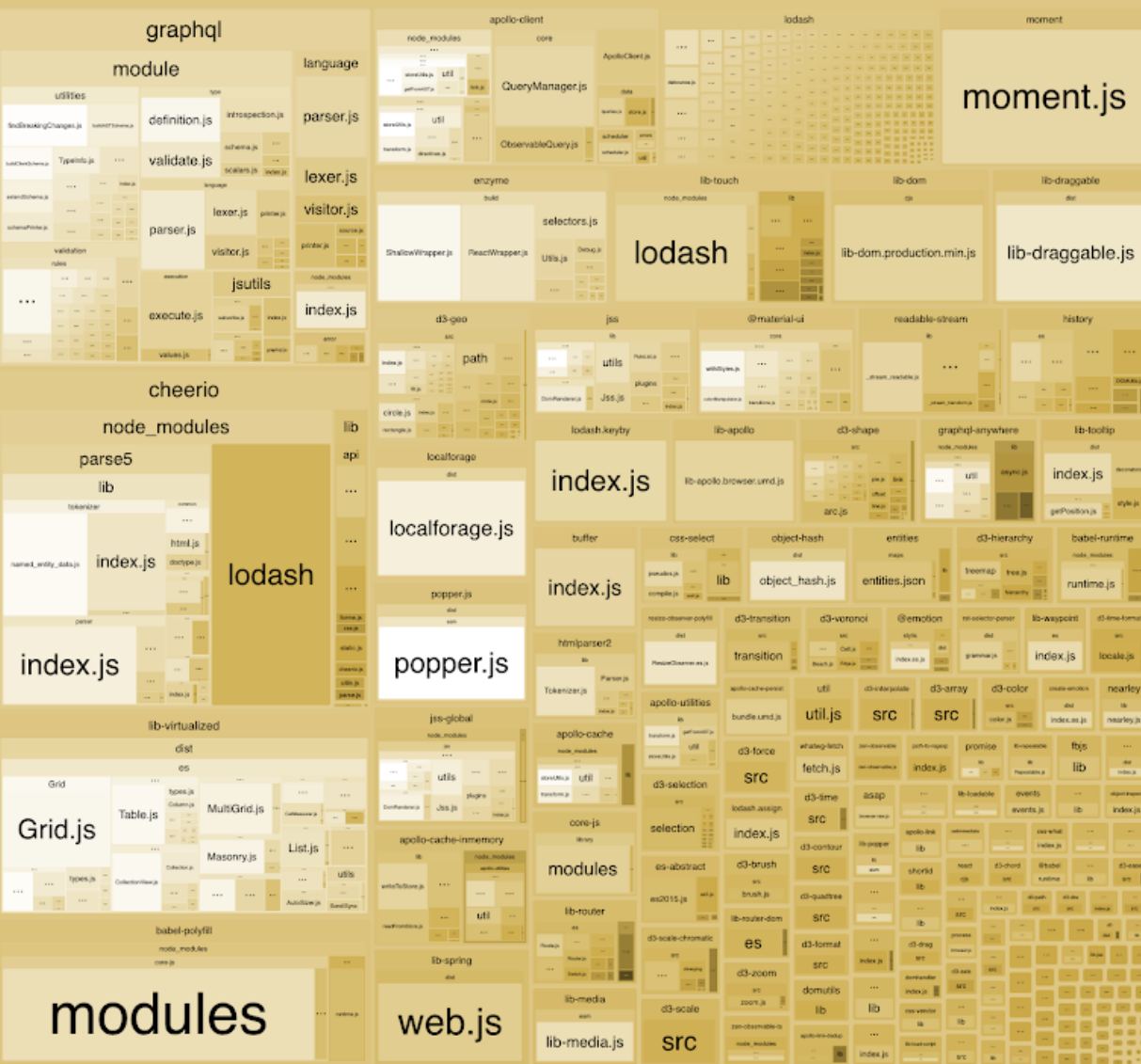
Harry Roberts ✓
@csswizardry



Welcome to Ukraine! Where the average webpage will cost you £12. [#perfmatters](#)

Welcome to UKRAINE. While you're here calls back to the UK will be £2.00 pm. To receive a call it will be £2.00 pm. Texts are 50p to send and free to receive. Data is £7.20/MB

node_modules

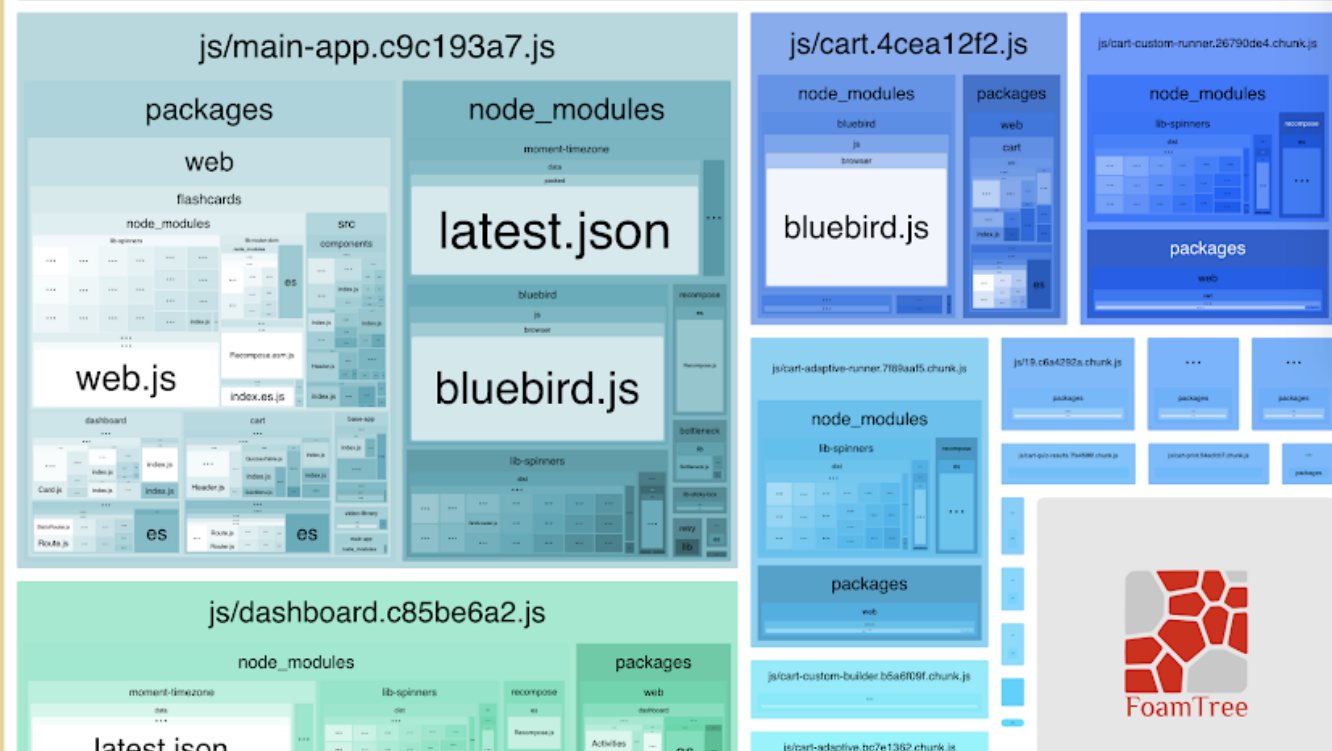
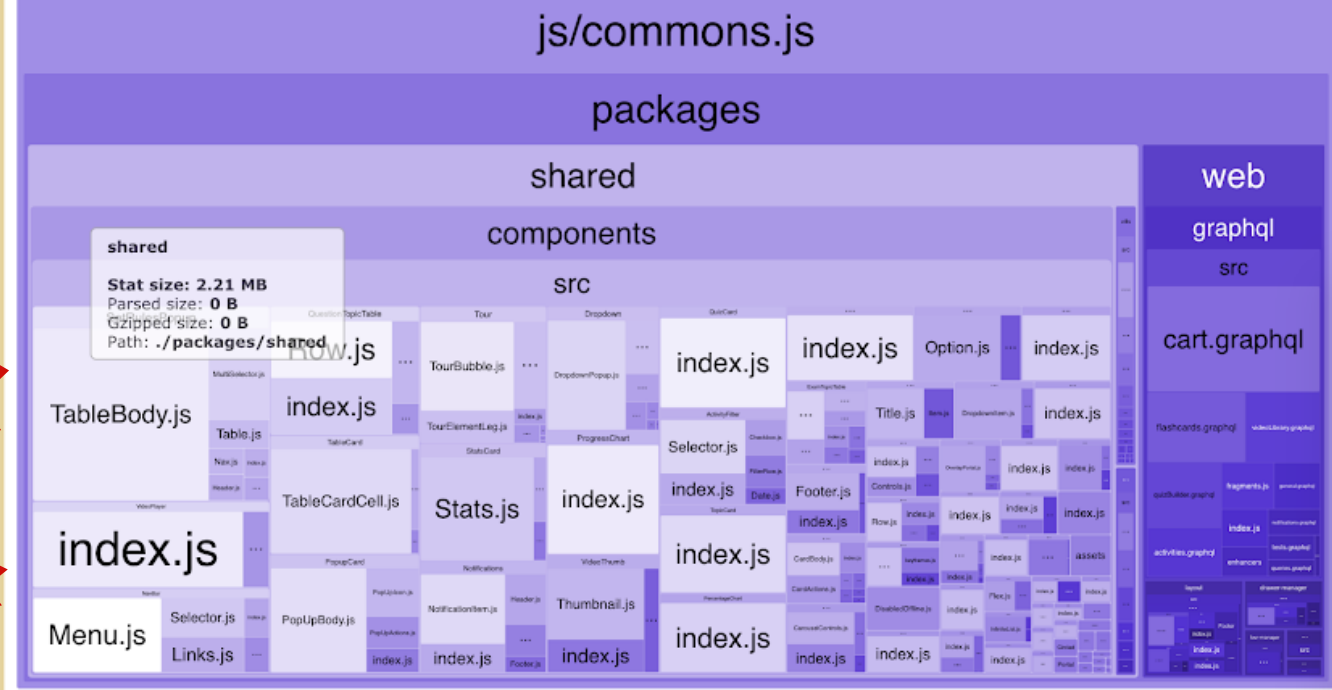


node_modules

APPLICATION CORE



node_modules



**WE OVER-VALUE HUMAN
ABILITY TO SEE LARGE
AMOUNTS OF CONTENT AT ONCE**

PROGRESSIVE RENDERING

PRIORITIZE RENDERING & LOADING

BATCHED RENDERING

PROGRESSIVE RENDERING

I can't speak Japanese. I wish I could,
though. I can only say Hello. And Kawasaki.

.



~~Two butts walk down the street. One farted.
The other one commented: "You took the
words right out of my mouth".~~

I can't speak Japanese. I wish I could,
though. I can only say Hello. And Kawasaki.

.



~~Two butts walk down the street. One farted.
The other one commented: "You took the
words right out of my mouth".~~

3. Your eye glanced
here next

4. Then you saw this area



1. You saw this first

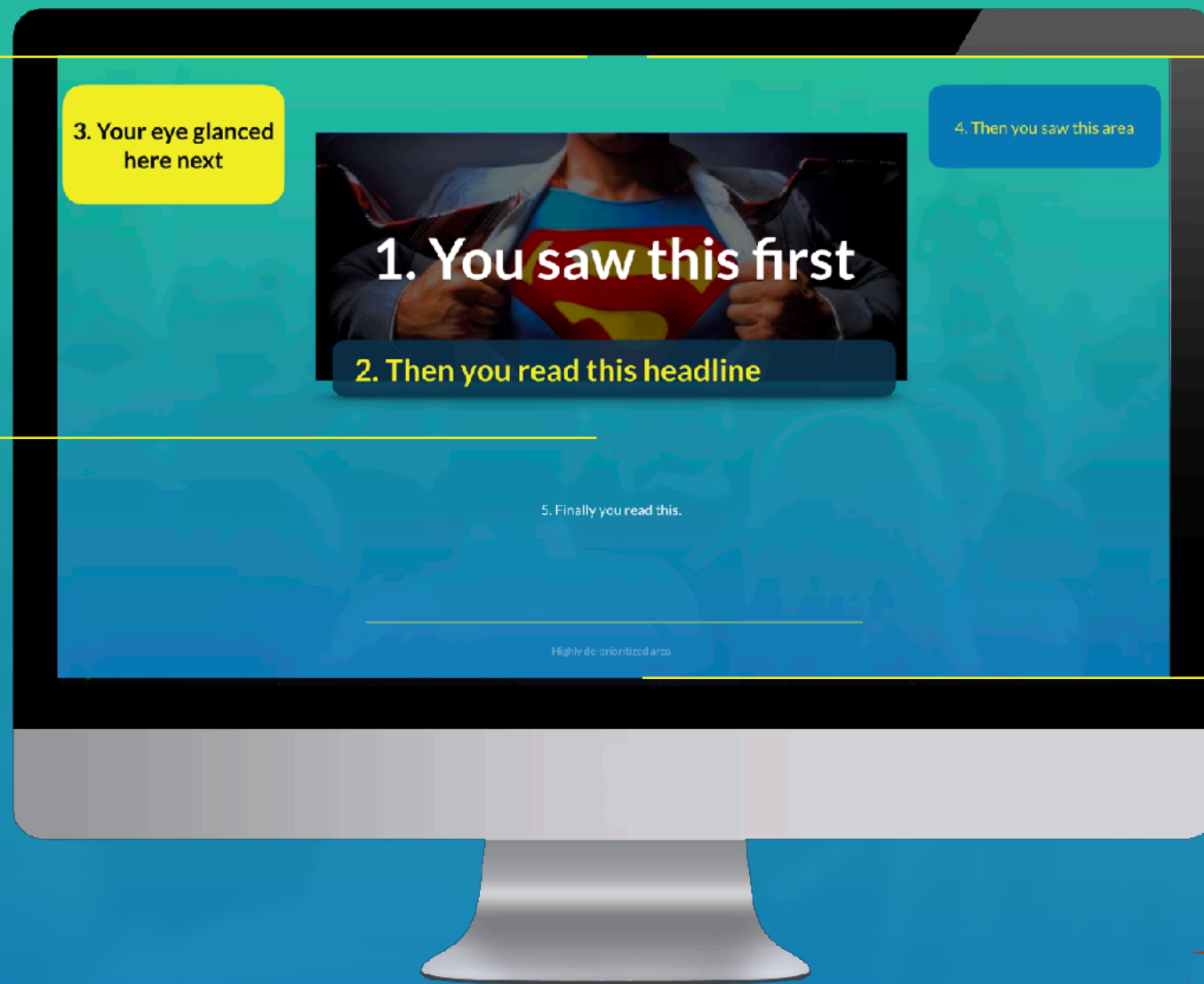
2. Then you read this headline

5. Finally you read this.

Highly de-prioritized area

CRITICAL
RENDERING
PATH

You control this



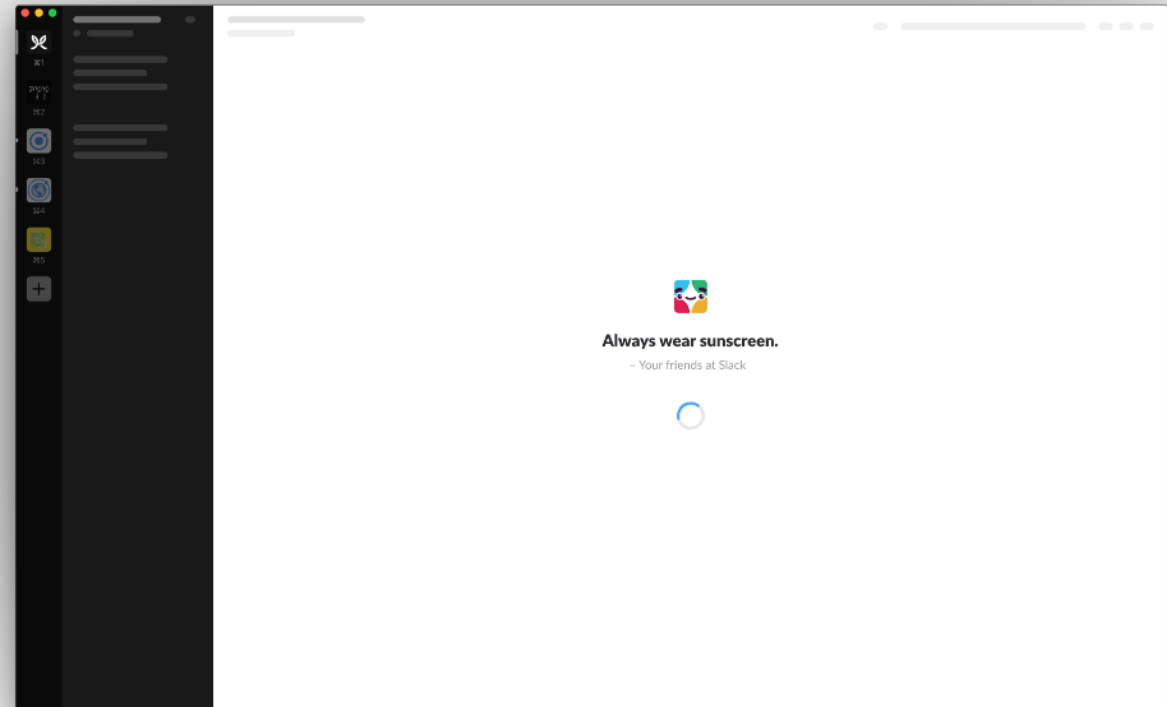
ABOVE
THE FOLD
CONTENT

Expensive deviation

APP CORE

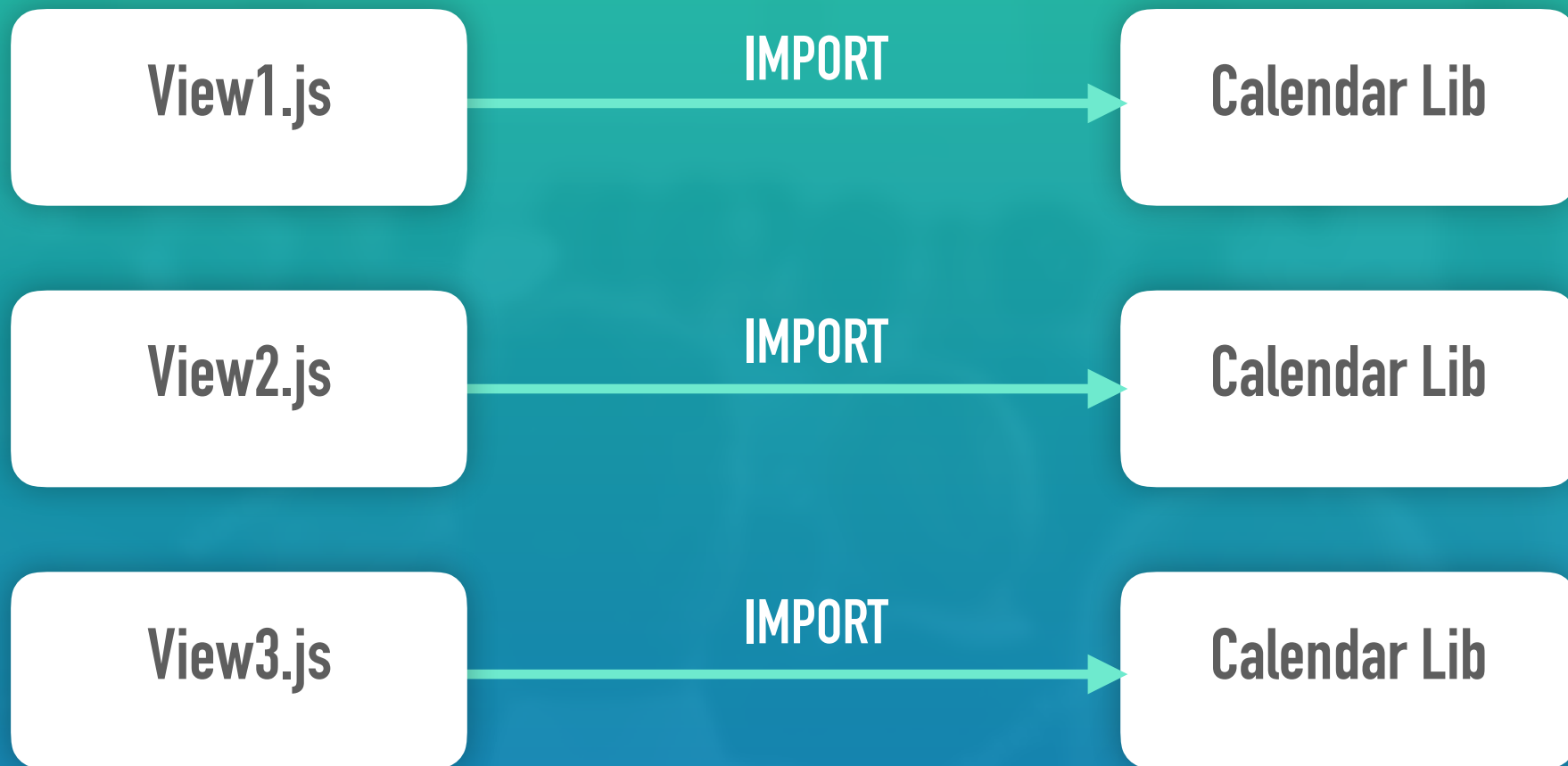
MINIMAL

COMMON LOGIC



WRAPPER MODULES

CONTAINERIZE CODE





CalWrapper.js

```
import React, { lazy, Suspense } from 'react';

const ReactCalendar = lazy(() => import('react-calendar'));

const Calendar = props => (
  <Suspense fallback={<div>Loading Calendar...</div>}>
    <ReactCalendar {...props} />
  </Suspense>
);

export default Calendar;
```

CalWrapper.js

```
import React, { lazy, Suspense } from 'react';

const ReactCalendar = lazy(() => import('react-calendar'));

const Calendar = props => (
  <Suspense fallback={<div>Loading Calendar...</div>}>
    <ReactCalendar {...props} />
  </Suspense>
);

export default Calendar;
```

CalWrapper.js

```
import React, { lazy, Suspense } from 'react';

const ReactCalendar = lazy(() => import('react-calendar'));

const Calendar = props => (
  <Suspense fallback={<div>Loading Calendar...</div>}>
    <ReactCalendar {...props} />
  </Suspense>
);

export default Calendar;
```

CalWrapper.js

```
import React, { lazy, Suspense } from 'react';

const ReactCalendar = lazy(() => import('react-calendar'));

const Calendar = props => (
  <Suspense fallback={<div>Loading Calendar...</div>}>
    <ReactCalendar {...props} />
  </Suspense>
);

export default Calendar;
```


CalWrapper.js

```
import React, { lazy, Suspense } from 'react';

const ReactCalendar = lazy(() => import('react-calendar'));

const Calendar = props => (
  <Suspense fallback={<div>Loading Calendar...</div>}>
    <ReactCalendar {...props} />
  </Suspense>
);

export default Calendar;
```

NAMED ROUTES

```
// We want to lazy-load the burger now  
// import { slide as Burger } from 'react-burger-menu';  
  
const Burger = lazy(() =>  
  import('react-burger-menu').then(module => ({  
    default: module.slide,  
  })))  
);
```

TIME LAZY-LOADING WITH ANOTHER MODULE

```
const Burger = lazy(() => {  
  import( './burger.css' );  
  return import( 'react-burger-menu' ).then( module => ({  
    default: module.slide,  
  }));  
});
```

NAMED CHUNKS

```
const Burger = lazy(() => {  
  import( './burger.css' );  
  return import( 'react-burger-menu' /* webpackChunkName: "burger" */ ).then(  
    module => ({  
      default: module.slide,  
    })  
  );  
});
```

COMBINING MODULES

```
const BurgerSlide = lazy(() => {  
  import('./burger.css');  
  return import('react-burger-menu' /* webpackChunkName: "burger" */).then(  
    module => ({  
      default: module.slide,  
    })  
  );  
});
```

```
const BurgerElastic = lazy(() => {  
  import('./burger.css');  
  return import('react-burger-menu' /* webpackChunkName: "burger" */).then(  
    module => ({  
      default: module.elastic,  
    })  
  );  
});
```

LAZY-LOAD ANY LIBRARY

```
async function asyncMoment() {  
  const moment = await import('moment' /* webpackChunkName: "moment" */);  
  return moment.default;  
}  
  
async function getTime() {  
  const moment = await asyncMoment();  
  console.log(moment().format('HH:ii'));  
}  
  
getTime();
```

WRAPPER MODULES

CREATE REUSABLE SPLIT POINTS

WRAPPER MODULES

HELP CONTAIN TECHNICAL DEBT

WRAPPER MODULES

SIMPLIFY UPGRADE PATHS

0.6s



1. You saw this first

2. Then you read this headline

1-2s

3. Your eye glanced
here next



1. You saw this first

2. Then you read this headline

2s

3. Your eye glanced
here next

4. Then you saw this area



1. You saw this first

2. Then you read this headline

2.5s

3. Your eye glanced
here next

4. Then you saw this area



1. You saw this first

2. Then you read this headline

5. Finally you read this.

2.8s

3. Your eye glanced
here next

4. Then you saw this area



1. You saw this first

2. Then you read this headline

5. Finally you read this.

Highly de-prioritized area

3s+

BROWSERS ❤️ JUST-IN-TIME RENDERING

IntersectionObserver 📄 - WD

Usage

% of all users

Global

72.55% + 4.02% = 76.57%

API that can be used to understand the visibility and position of DOM elements relative to a containing element or to the top-level viewport. The position is delivered asynchronously and is useful for understanding the visibility of elements and implementing pre-loading and deferred loading of DOM content.

Current aligned Usage relative Date relative Apply filters Show all ?

Edge *	Firefox	Chrome	Safari	iOS Safari *	Android Browser *	Chrome for Android	UC Browser for Android
16	64	⁴ 71	11.1	11.2	4.4		
17	65	⁴ 72	12	11.4	4.4.4		
18	66	⁴ 73	12.1	12.1	67	71	³ 11.8
	67	⁴ 74	TP	12.2			
	68	⁴ 75					
		⁴ 76					

REDUCE THE LAZY-LOADING EFFECT

I can't speak Japanese. I wish I could, though. I can only say Hello. And Kawasaki.

I deeply admire Japanese culture. Next time I visit I will tour across the country to relish the culture some more.

You can't see it, but I'm laughing inside. It's really hard to keep my face serious. If you all start smiling, I'll register that you've read this. Please don't make me burst laughing!

Ok, now I have to add a bunch of text to make this feel like a busy slide.

~~Two butts walk down the street. One farted. The other one commented: "You took the words right out of my mouth".~~

A black hole with a bright, glowing accretion disk. The center is a solid black circle. The text "COMPRESSING TIME" is written in white, bold, sans-serif capital letters in the center of the black circle. The background is a dark space filled with stars and a blueish-white nebula on the left side.

**COMPRESSING
TIME**

**TIME IS A MEASURE OF
CYCLES OF CHANGE
WITHIN THE WORLD.**

SIR ISAAC NEWTON

LOADING INDICATORS

WAIT = 1 - 2 SECONDS



PRELOAD EAGERLY
PREFETCH COMFORTABLY

RESOURCE HINTING

```
async function asyncMoment() {  
  const moment = await import(  
    'moment' /* webpackChunkName: "moment", webpackPrefetch: true */  
  );  
  return moment.default;  
}
```

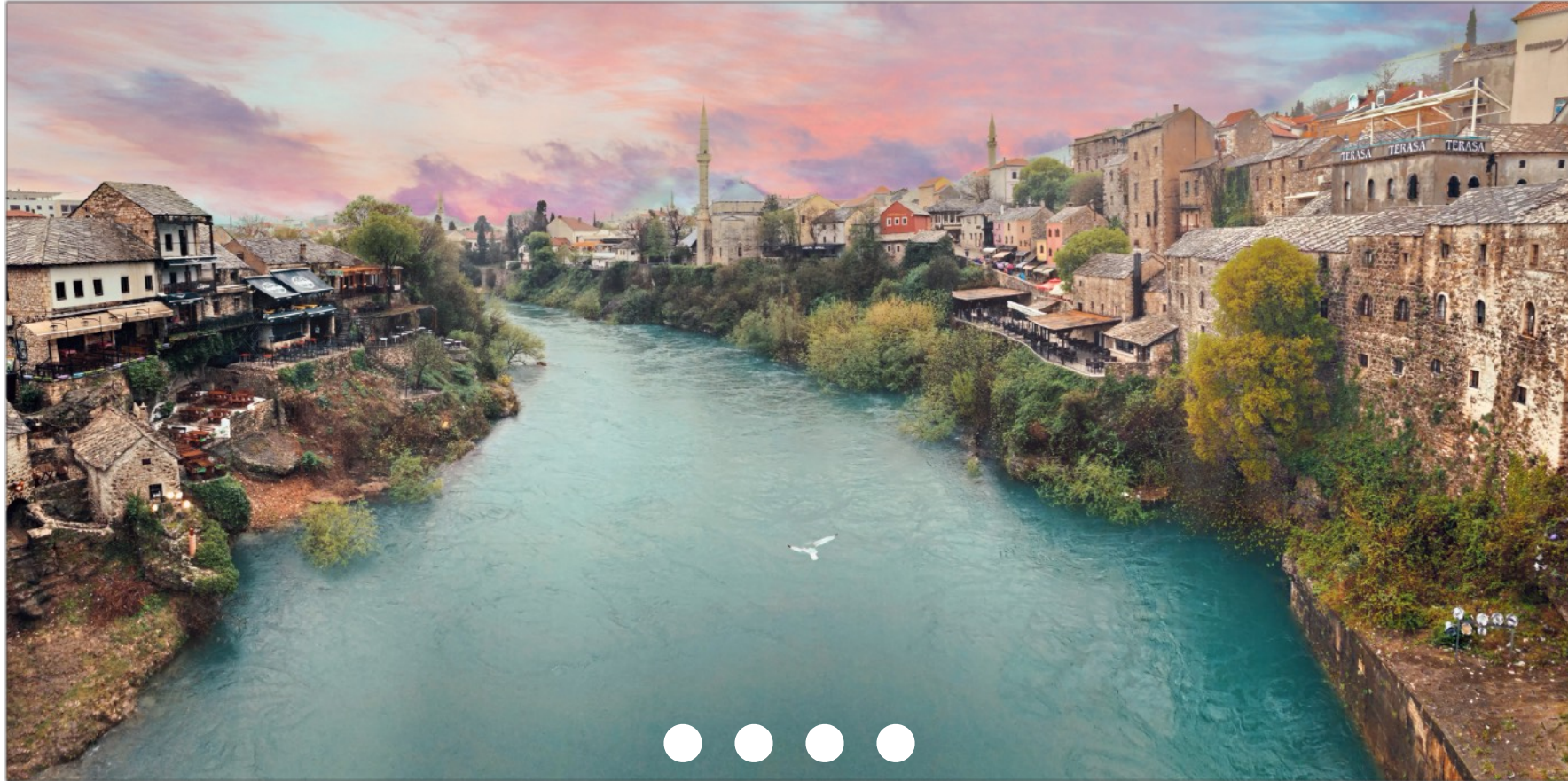
INTRODUCE NOVELTY



ANIMATION

IS A FORM OF STORYTELLING

EXAMPLE: CAROUSEL



TIMING IMAGE LOADING

```
<!-- Prioritized (eager) loading -->  
  
  
<!-- Deferred (lazy) loading -->  

```

CHROME 75

EXPOSURE TO BRIGHT LIGHT THEN DIMMING DOWN

TEASES BRAIN'S PERCEPTION OF TIME

JS

TS

JS

TS

CONTRAST SETS FOCUS

JS

TS

JS

TS

CHANGE OF FOCUS

COSTS



300ms



OPTIMISTIC INTERACTION

TAKE CARE OF THE DEFECTS IF THEY OCCUR

BUDGET ALL THE THINGS

APP CORE SIZE

120KB GZIPPED

CHUNK SIZE

170KB GZIPPED

FIRST MEANINGFUL PAINT (FMP)

2S ON 3G

TIME TO INTERACTIVE (TTI)

5S ON 3G

AUTOMATE BUDGETING

WEBPACK CONFIG

BUNDLESIZE

LIGHTHOUSE CI

HEADLESS CHROME & DEV TOOLS

@GGRGUR

@MODUSCREATE