

Conducting Performance Audits



Kamran Ayub

Helping React developers create snappy apps

@kamranayub | www.kamranayub.com

Live Video: Introduction

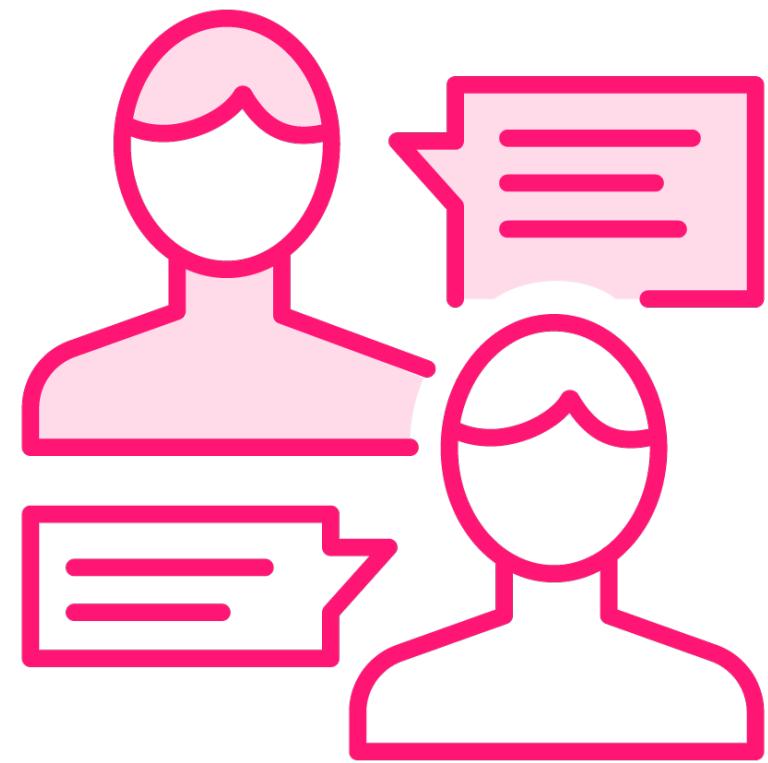


Performance Audit

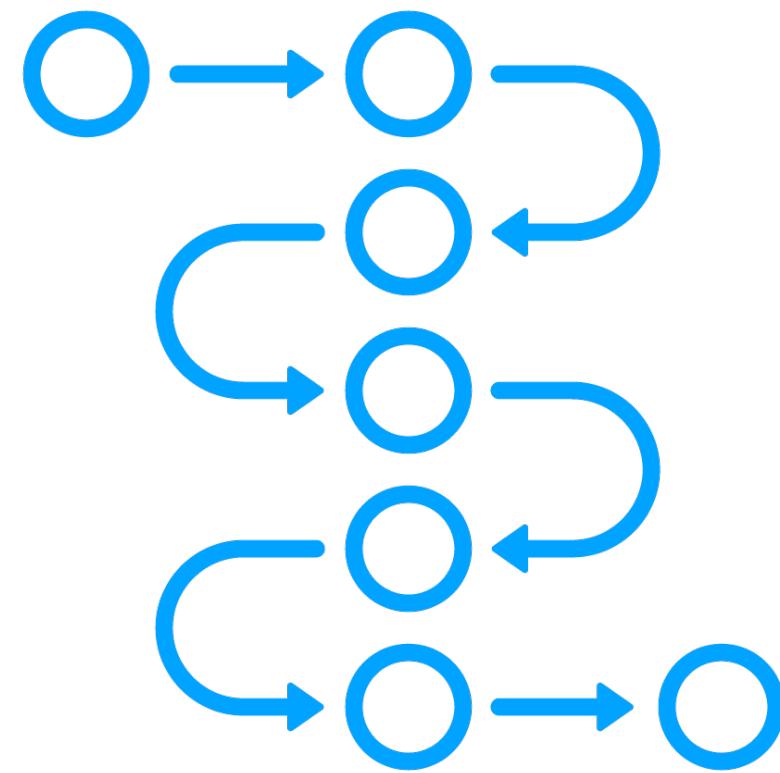
A systematized, repeatable process anyone on your team can follow to measure your application's performance, create a report, and prioritize issues to address.



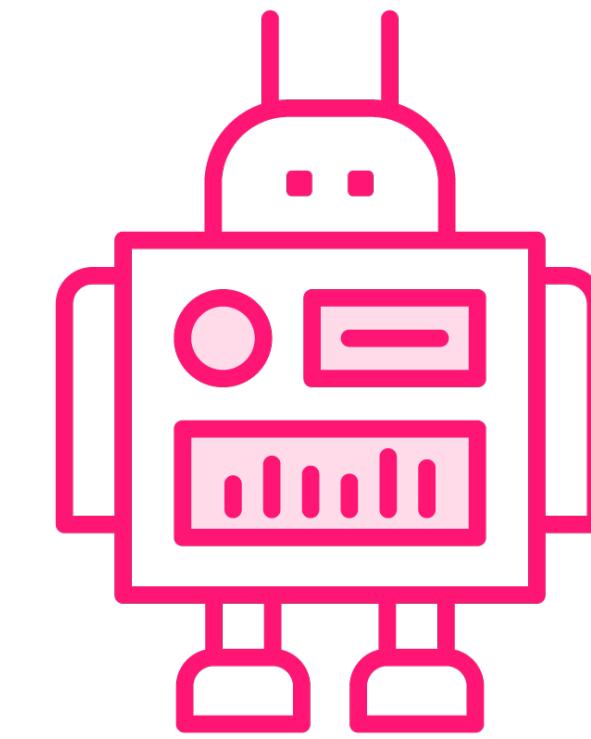
Benefits of Performance Audits



Tool to make better decisions



Standardized and repeatable process



Provides a path to automation



Dr. Russel Ackoff

**“Until managers take into account
the systemic nature of their
organizations, most of their efforts
to improve their performance are
doomed to failure.”**



Live Example



What Does an Audit Look Like?



Defining a Strategy: Scope and Objectives



Defining a Strategy: Scope and Objectives

**What is the goal? Why are we
doing this?**



Defining a Strategy: Scope and Objectives

What is the goal? Why are we doing this?

How does this impact the business? Have we defined any KPIs?



Defining a Strategy: Scope and Objectives

What is the goal? Why are we doing this?

How does this impact the business? Have we defined any KPIs?

Which routes and components are included vs. not included?



Defining a Strategy: Scope and Objectives

What is the goal? Why are we doing this?

How does this impact the business? Have we defined any KPIs?

Which routes and components are included vs. not included?

What are the various assumptions or risks?



Defining a Strategy: Scope and Objectives

What is the goal? Why are we doing this?

How does this impact the business? Have we defined any KPIs?

Which routes and components are included vs. not included?

What are the various assumptions or risks?



Defining a Strategy: Tools and Methodology



Defining a Strategy: Tools and Methodology

List the tools and versions



Defining a Strategy: Tools and Methodology

List the tools and versions

Document your steps (SOP)



Defining a Strategy: Tools and Methodology

List the tools and versions

Document your steps (SOP)

You will come back and refine
these steps over time



Defining a Strategy: Tools and Methodology

List the tools and versions

Document your steps (SOP)

**You will come back and refine
these steps over time**

Ensures you can repeat it



Live Video: Systematize!



Performance Audit Step-by-Step



Example: Improving Add to Cart (A2C) Conversion

Context

E-commerce retailer, getting ready for the holiday season



Example: Improving Add to Cart (A2C) Conversion

Context

E-commerce retailer, getting ready for the holiday season

Scope

Holiday landing pages and product pages, add to cart



Example: Improving Add to Cart (A2C) Conversion

Context

E-commerce retailer, getting ready for the holiday season

Scope

Holiday landing pages and product pages, add to cart

Assumptions

Majority anonymous users, checking out as guests.
Will be an image-heavy experience. Mobile emphasis.



Example: Improving Add to Cart (A2C) Conversion

Context

E-commerce retailer, getting ready for the holiday season

Scope

Holiday landing pages and product pages, add to cart

Assumptions

Majority anonymous users, checking out as guests.
Will be an image-heavy experience. Mobile emphasis.

Risks

Images will have an outsized impact on UX.



Example: Improving Add to Cart (A2C) Conversion



Example: Improving Add to Cart (A2C) Conversion

Objective

**Measure and focus on the impact
of imagery on the add to cart
experience**



Example: Improving Add to Cart (A2C) Conversion

Objective

Measure and focus on the impact of imagery on the add to cart experience

Impact

We expect any improvements will lead to increased holiday sales



Example: Improving Add to Cart (A2C) Conversion

Objective

Measure and focus on the impact of imagery on the add to cart experience

Impact

We expect any improvements will lead to increased holiday sales

Primary KPI to measure will be A2C conversion rate from holiday landing pages

KPIs



Example: Improving Add to Cart (A2C) Conversion

Objective

Measure and focus on the impact of imagery on the add to cart experience

Impact

We expect any improvements will lead to increased holiday sales

Primary KPI to measure will be A2C conversion rate from holiday landing pages

KPIs

We will run an A/B experiment using an email promotion to compare with new changes

Methodology



Strategy Connects the Dots



Define strategy

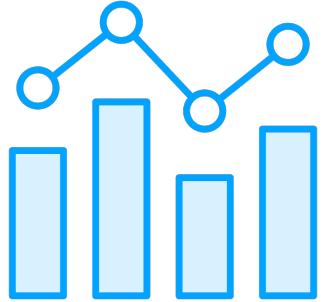
- **Scope**
- **Objectives**
- **Tools**
- **Methodology**



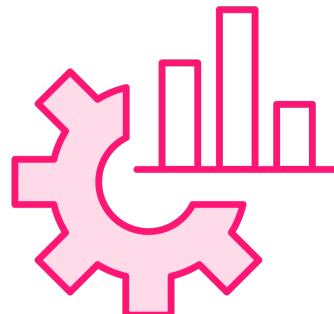
Prepping for the Audit



Measure using a consistent environment



Make sure KPI data measurement is in place



This may mean adding new collection mechanisms



Performing the Audit



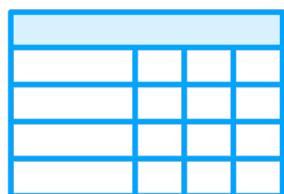
Measure your routes and/or component flows



Document the steps as you go



Repeat tests a few times to avoid outliers



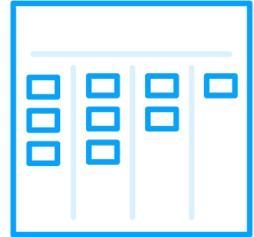
Record data and upload capture files



Organizing the Report



Dive into the (messy) data to dig up insights

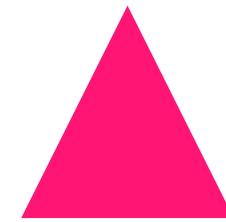


Bucket issues you find by priority/severity



3-Tier Prioritization System

Help organize and prioritize potential performance improvements discovered during the audit



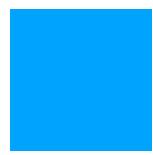
Highly Recommended

Urgent change that will significantly improve performance



Suggested

Needs more analysis but could likely to improve performance



Maybe Later

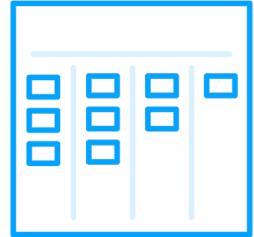
Potential for minor improvements if we have time



Organizing the Report



Dive into the (messy) data to dig up insights



Bucket issues you find by priority/severity



Add notes and comments to provide context



Summarize the key insights at the top



Demo: Sample Report & Template

Sample Performance Audit

For Globomantics FY2024 Holiday Season

Initiated: Aug 8, 2024

Prepared by: Kamran Ayub

Team

Kamran (Principal Engineer, Web)
Farrah (Product Owner, Web)
Shelly (PM, Web)
Javi (VP of Engineering, Web)
Sally (Promo Team, Digital Marketing)
Curt (VP, Digital Marketing)
Hussein (Sr Data Analyst, Demand Sales)

----- Page Break -----

Context

We are prepping for the holiday shopping season (Oct 15-Jan 5) and will be running gift promotions. We want to make sure that the performance of our landing pages is optimized so that we can increase holiday sales. FY2023 Q4 sales were \$678MM from our digital web channel.

Holiday landing pages (LPs) are customized by marketing team in CMS, then promoted through deals & promo mailing automation. We currently see about 1.6MM visitors pushed after each email campaign.

Why are we doing this?

Objective: The goal is to make sure it's a smooth experience to add products to the cart from the LPs.

Impact: We expect higher LP to A2C conversion with better performance which will lead to increased sales for Q4 2024.

KPI: A2C conversion attributed to holiday LPs

Scope

Testing will be performed on each of the holiday landing pages provided below:

- /holiday-gift-guide
- /thanksgiving-cooking-ideas
- /holiday-gift-guide
- /cyber-monday-deals

We will focus on **mobile** and **tablet** experiences (76% of historical holiday traffic). **Desktop** is in-scope but de-prioritized unless it's a quick win.

Methodology

Testing will be performed in staging preview environment (<https://www-stage.globomantics.com>).

Over FY2024, we have released numerous improvements and changes to the site. To isolate the changes made in response to the audit, we will use a feature flag (AUDIT_FY2024_Q4) to opt-in traffic to changes.

For any changes that are prioritized and released for this audit, we will measure the before vs. after KPI attributed via an email A/B test:

- On Oct 18, email will go out to 50% of recipients with non-optimized "Halloween Gift Guide" LP, and 50% of recipients will get the optimized version
- On Oct 22, email will go out with segments reversed, to help account for differences in open rate

Tools

- CMS (holiday content) – <https://cms.globomantics.com>
- ConvertKit (email automation)
- LaunchDarkly (feature flags)
- Google Lighthouse (measurement tool) – v3.25.6
- React Dev Tools v4.27.7
- WebPageTest (mobile device simulation)

Timeline

Shelly and Farrah to refine/confirm with partners.

- **Aug 1-Sep 5:** Prep, define strategy, identify & take care of prerequisites
- **Sep 10:** Audit with staging/production-ready CMS data
- **Sep 15-Oct 16:** Implement prioritized code changes before freeze using feature flag
- **Oct 18:** Marketing promo A test
- **Oct 22:** Marketing promo B test
- **Oct 22-28:** Look at the data, report on expected results
- **Oct 26:** GO/NO-GO decision (leave feature flag on/off)
- **Oct 28:** CODE FREEZE – P1 fixes only (with VP approval)

Assumptions & Risks

- **Risk:** Images will have an outsized impact on UX
- **Risk:** Email A/B test will not give us 100% accurate segmentation/attribution data but based on historical open rates and CTR, product thinks it'll be "good enough" to be useful
- **Risk:** There were multiple features released in FY2024 that may negatively impact sales, but won't be visible via an audit
- **Assumption:** Marketing team reports historically that promotions drive higher anonymous user traffic, so we will prioritize unauthenticated user experience
- **Assumption:** Landing pages will be accessible in staging environment for testing before they go-live to users
- **Assumption:** Landing pages will be using production imagery and CMS data expected to go-live

Demo: Sample Report & Template

Methodology

Testing will be performed in staging preview environment (<https://www-stage.globomantics.com>).

Over FY2024, we have released numerous improvements and changes to the site. To isolate the changes made in response to the audit, we will use a feature flag (AUDIT_FY2024_Q4) to opt-in traffic to changes.

For any changes that are prioritized and released for this audit, we will measure the before vs. after KPI attributed via an email A/B test:

- On Oct 18, email will go out to 50% of recipients with non-optimized "Halloween Gift Guide" LP, and 50% of recipients will get the optimized version
- On Oct 22, email will go out with segments reversed, to help account for differences in open rate

Tools

- CMS (holiday content) – <https://cms.globomantics.com>
- ConvertKit (email automation)
- LaunchDarkly (feature flags)
- Google Lighthouse (measurement tool) – v3.25.6
- React Dev Tools v4.27.7
- WebPageTest (mobile device simulation)

Timeline

Shelly and Farrah to refine/confirm with partners.

- **Aug 1-Sep 5:** Prep, define strategy, identify & take care of prerequisites
- **Sep 10:** Audit with staging/production-ready CMS data
- **Sep 15-Oct 16:** Implement prioritized code changes before freeze using feature flag
- **Oct 18:** Marketing promo A test
- **Oct 22:** Marketing promo B test
- **Oct 22-28:** Look at the data, report on expected results
- **Oct 26:** GO/NO-GO decision (leave feature flag on/off)
- **Oct 28:** CODE FREEZE – P1 fixes only (with VP approval)

Assumptions & Risks

- **Risk:** Images will have an outsized impact on UX
- **Risk:** Email A/B test will not give us 100% accurate segmentation/attribution data but based on historical open rates and CTR, product thinks it'll be "good enough" to be useful
- **Risk:** There were multiple features released in FY2024 that may negatively impact sales, but won't be visible via an audit
- **Assumption:** Marketing team reports historically that promotions drive higher anonymous user traffic, so we will prioritize unauthenticated user experience
- **Assumption:** Landing pages will be accessible in staging environment for testing before they go-live to users
- **Assumption:** Landing pages will be using production imagery and CMS data expected to go-live



Demo: Sample Report & Template

Sample Audit Report

Report Date: Sep 13, 2024

EXECUTIVE SUMMARY

We identified a few key improvements to the FilmStrip component and photo storage hosting that could potentially improve page load by 20%. Based on historical Q3 LP to A2C KPI data (see *Appendix 1: FY2024 Q3 Baseline Dataset*), we believe that these improvements could translate to an **incremental sales lift of 3.2% for Q4**. Other suggested improvements could further add **+0.5-0.85% lift in sales** if we have time before the holiday code freeze.

For sales forecasting and analysis model, see *Appendix 2: Simulated Sales Regression Analysis*.

Issues to Address

▲ HIGHLY RECOMMENDED

Insight	Comments / Recommendations	Expected Impact
▲ FilmStrip component uses 67% of page load budget.	<p>The film strip is a key component that is heavily used in holiday landing page slots.</p> <p>It is used to display a carousel that the user can page through and quickly add items to their cart.</p> <p>We noticed that there is no "lazy loading" behavior, so some carousels with 5+ items are loading on mobile even though they are not visible, affecting initial load times.</p>	Lazy loading carousel items that are off-screen should decrease Film Strip's page budget usage by at least 50%.
▲ Image storage logic in CMS is not resizing images for target device.	<p>We identified that images uploaded through the CMS tool are JPEG images and are not being optimized for web compression (WebP), nor are taking advantage of "srcset" (HTML5 picture) optimizations for mobile screen sizes.</p> <p>The holiday LPs display many images, and this impacts Web Vitals FCP/LCP metrics.</p>	Providing alternative WebP images when uploaded through the CMS would lead to significant bandwidth savings (87%) on supported platforms (iOS Safari, Chrome for Android), translating to decreased device render time (FCP/LCP metrics in Web Vitals).



Demo: Sample Report & Template

Test Results and Runs

Flow 1: Email to landing page to carousel A2C

Steps to Test

- PREP ENV
 - Verify staging preview is started (<https://www-stage.globomantics.com>)
 - Close all apps
 - Zoom
 - Slack
 - Spotify
 - Other Chrome instances
 - Check taskmgr for running apps using CPU
 - Launch Chrome profile "Auditing"
 - Clear all site data from browser
 - Close all tabs except one in main browser profile
 - Open email client
- Login to CMS (<https://cms.globomantics.com/login>)
 - If access denied, check AD group (DIGITAL_WEB_CMS_CONTRIB)
 - Open quick search and find holiday promo slots ("FY2024_HOLIDAY_" prefix)
 - For landing page, choose the appropriate slot (e.g. "FY2024_HOLIDAY_HALLOWEEN")
 - Click "Send test campaign" and check box "Send to me only"
 - Trigger test promo email from CMS
 - Open email in client
 - Click into secondary Chrome browser profile (test profile)
 - IF UNAUTHENTICATED FLOW: Clear all history/data
 - Click gift guide promo image
 - Should open in test profile browser
 - Verify utm params are set
 - utm_source=<check slot variable in CMS> (e.g. "cms_manual_test")
 - utm_campaign=<check slot variable in CMS> (e.g. "ck_campaign_20240905_01")
 - utm_content=<based on image you clicked>
 - utm_medium=email
- FOR PROFILING A2C:
 - Open browser dev tools (F12)
 - IF MOBILE: Set device view (iPhone 12 Pro)
 - IF TABLET: Set device view (iPad Pro – Landscape)
 - Start React dev tools
 - Go into Profiler tab and click "Reload and profile" for hard reload profiling
 - Scroll down to "Treats That Say I Love You, Boo!"
 - Click arrow to scroll right
 - On black cat sticking tongue out t-shirt
 - Click size Medium in dropdown
 - Click orange color
 - Click "Add to cart"
 - Stop profiler when message confirmation toast appears
- FOR INITIAL PAGE LOAD:
 - Open Lighthouse tools in Chrome
 - Only check Performance
 - Start report
 - Save captured HTML report to SHARED DRIVE\Audits\2024\<QUARTER>\<filename>
 - Paste link to report in test results below

Test Results

Code Commit: [672cf7e](#)

Route: /l/halloween-gift-guide?<utm_params>

Test Date	Test Variables	Run	Results / Summary	Notes	Lighthouse Reports	React Profile Sessions
2024-09-06	Unauthenticated Mobile (iPhone 12 Pro)	1	LH Score 79 LCP: 765ms FilmStrip: 512ms render 25 commits		Link	Link
		2	LH Score 78 LCP: 748ms FilmStrip: 565ms render 25 commits		Link	Link
		3	LH Score 67 LCP: 816ms FilmStrip: 530ms render 25 commits	Outlier; system update was started.	Link	Link
		4	LH Score 79 LCP: 744ms FilmStrip: 522ms render 25 commits		Link	Link
2024-09-06	Authenticated Mobile (iPhone 12 Pro)	1			Link	Link
		2			Link	Link
		3			Link	Link

Test Notes

FilmStrip render time

In all the tests, noticed high render time for FilmStrip (67% of LCP). When investigated, browser script profiling showed high execution time on `reducers/products.ts` (`getFormattedImagesForCarousel()`) ([code link](#)). Called for every image loaded (~50 on page). Noticed all children cards render on mobile even when off-screen. Network tab shows all images loading.



Live Outro



**Do you need all this for
making one-off
performance
improvements?**



Live Outro



Understanding and Auditing Web Vitals



Live Intro



What Are the Core Web Vitals?

LCP

Largest Contentful Paint

CLS

Cumulative Layout Shift

INP

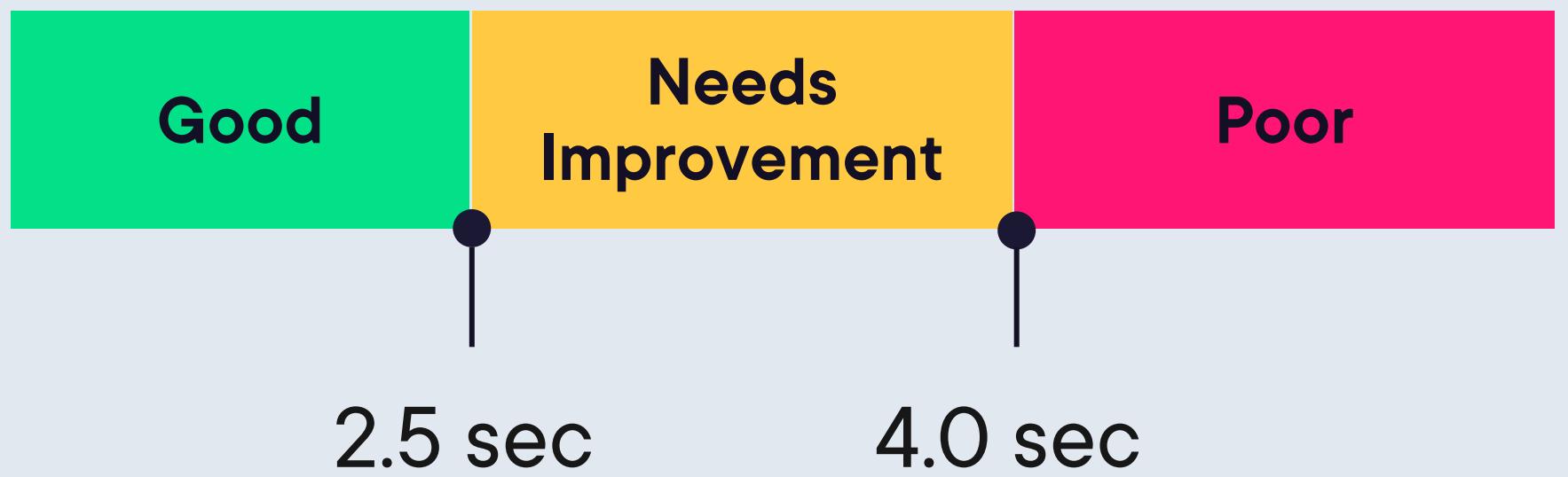
Interaction to Next Paint



Largest Contentful Paint (LCP)

Measures *loading* performance.

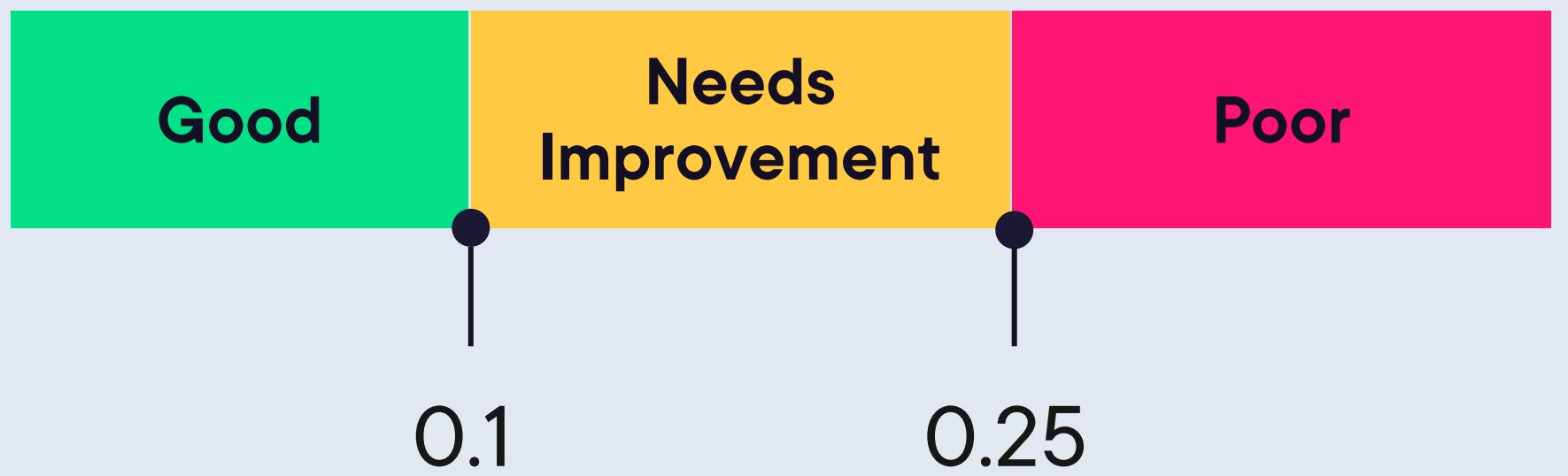
How long does it take to show the largest piece of content to the user?



Cumulative Layout Shift (CLS)

Measures *visual stability*.

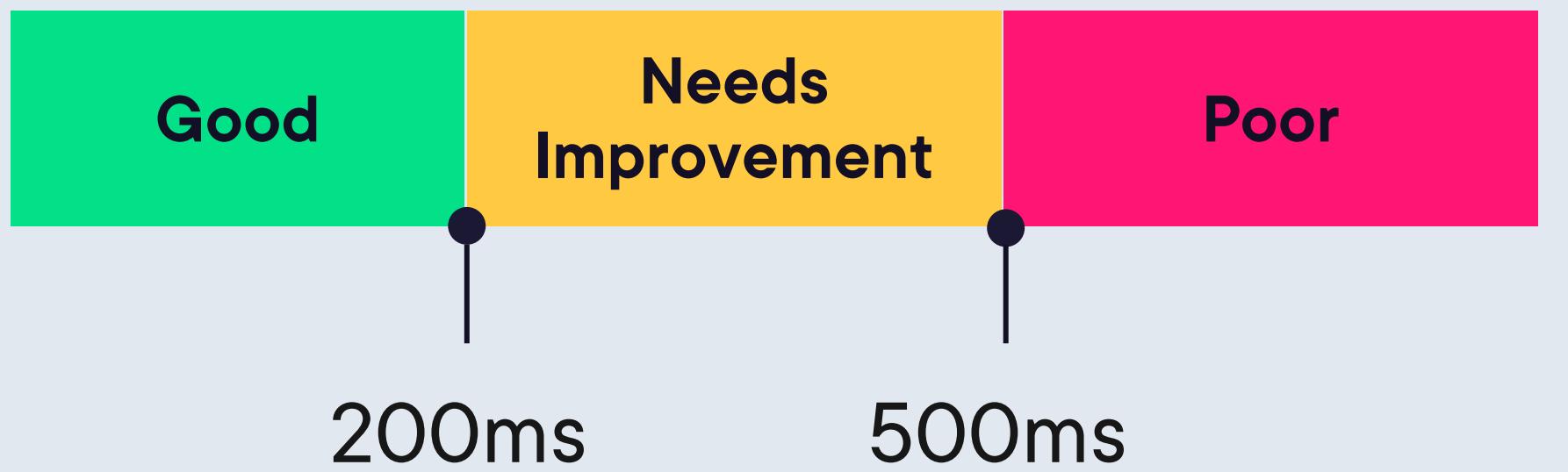
How "janky" does the page look?



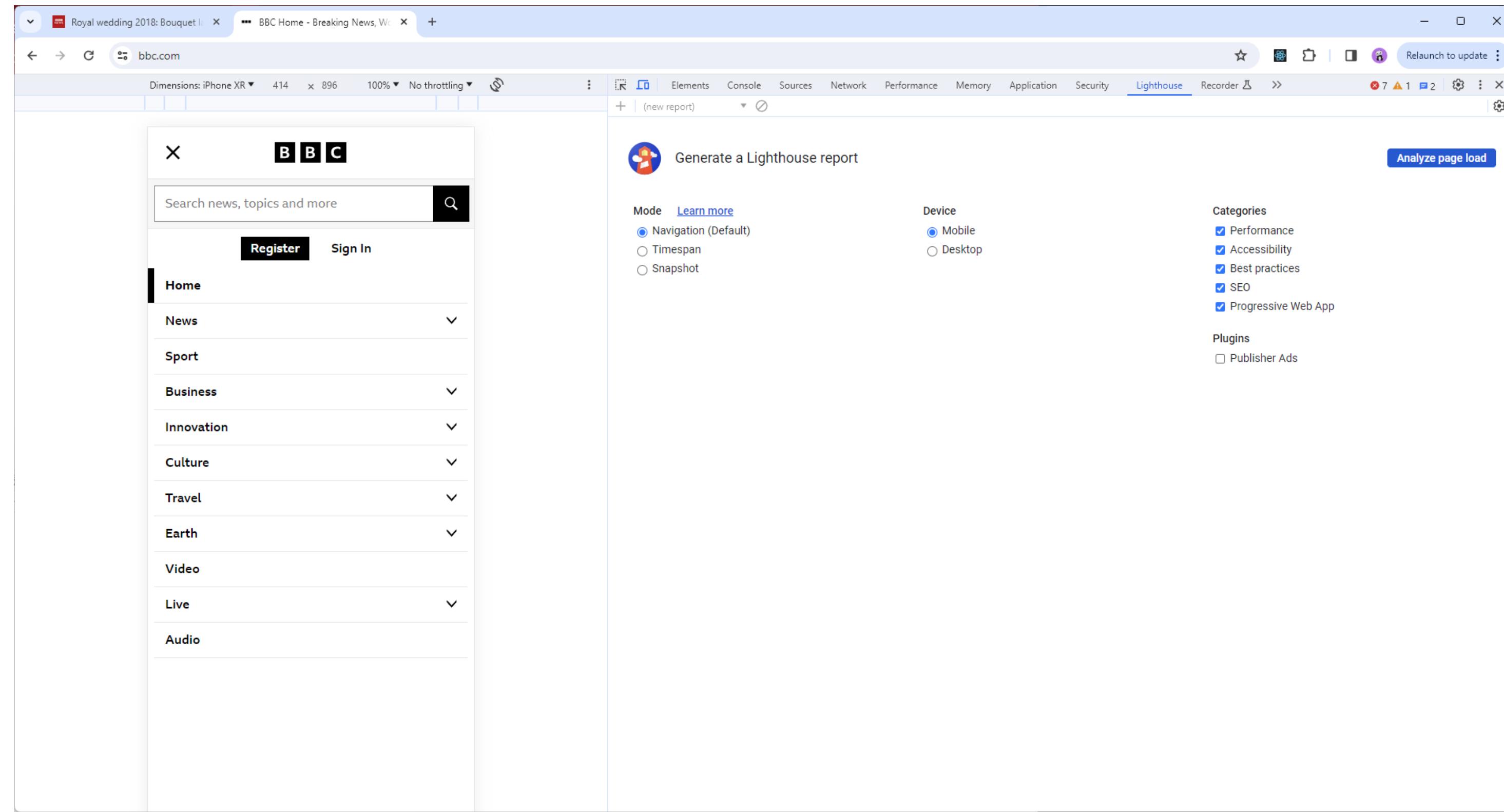
Interaction to Next Paint (INP)

Measures responsiveness.

How long does the user have to wait until they can reliably interact with the page?



Lighthouse in the Browser



Lighthouse in the Browser (Cont.)

The screenshot shows the Chrome DevTools interface with the Lighthouse tab selected. The main panel displays the BBC homepage with a performance score of 48. Below the score, there's a note about issues: "There were issues affecting this run of Lighthouse: Chrome extensions negatively affected this page's load performance. Try auditing the page in incognito mode or from a Chrome profile without extensions." The left sidebar lists audit metrics with their values:

- First Contentful Paint: 1.9 s
- Largest Contentful Paint: 12.7 s
- Total Blocking Time: 1,110 ms
- Speed Index: 4.3 s
- Cumulative Layout Shift: 0.007

The right panel provides a detailed breakdown of the Performance metric, showing a score of 48 with a visual representation of the score and two screenshots of the BBC homepage illustrating layout shifts.



Lighthouse in the Browser (Cont.)

▲ Reduce unused JavaScript 1.80s ▲

Reduce unused JavaScript and defer loading scripts until they are required to decrease bytes consumed by network activity. [Learn how to reduce unused JavaScript.](#) [LCP](#)

NEXT: Use Webpack Bundle Analyzer to detect unused JavaScript code. [Learn more](#)

Show 3rd-party resources (2)

URL	Transfer Size	Potential Savings
covatic.io	276.1 KiB	135.4 KiB
...v1/latest.js (browser.covatic.io)	276.1 KiB	135.4 KiB
bbc.com [1st Party]	139.8 KiB	59.8 KiB
...chunks/307-201a7fa2b7dfa663.js (www.bbc.com)	68.0 KiB	33.9 KiB
...pages/_app-787bdca8ae0c7a8a.js (www.bbc.com)	71.9 KiB	25.9 KiB
bbc.co.uk	39.4 KiB	35.3 KiB
...bump-4/bump-4.js (emp.bbc.co.uk)	39.4 KiB	35.3 KiB

▲ Enable text compression 1.35s ▲

Text-based resources should be served with compression (gzip, deflate or brotli) to minimize total network bytes. [Learn more about text compression.](#) [FCP](#) [LCP](#)

NEXT: Enable compression on your Next.js server. [Learn more](#).

URL	Transfer Size	Potential Savings
-----	---------------	-------------------



Lighthouse Standalone

GoogleChrome / lighthouse Public

Code Issues 581 Pull requests 58 Discussions Actions Security Insights

main 475 branches 132 tags Go to file Code

EvilKarter docs: update deprecated link to shared flags (#15722)	✓ 28ba6e0 9 hours ago	5,977 commits
.github tests(ci): stop using xvfb (#15707)	2 weeks ago	
assets misc(assets): update logo (#13919)	2 years ago	
build misc(build): set line limit to 1000 (#15733)	yesterday	
cli tests(dbw): fix server latency flake (#15729)	yesterday	
clients tests(ci): stop using xvfb (#15707)	2 weeks ago	
core deps: upgrade puppeteer to 21.7.0 (#15724)	9 hours ago	
docs docs: update deprecated link to shared flags (#15722)	9 hours ago	
flow-report report(performance): use metric savings for metric filter (#15540)	2 days ago	
lighthouse-logger misc: adopt minimal license headers (#15456)	4 months ago	
proto report: add timespan/snapshot mode labels to footer (#15589)	2 months ago	

About
Automated auditing, performance metrics, and best practices for the web.

[developer.chrome.com/docs/lighthouse/...](https://developer.chrome.com/docs/lighthouse/)

web pwa chrome-devtools
best-practices audit performance-metrics
developer-tools performance-analysis

Readme
Apache-2.0 license
Code of conduct
Activity
27.5k stars
592 watching
9.6k forks
Report repository



Lighthouse in CI

GoogleChrome / lighthouse-ci Public

Code Issues 176 Pull requests 7 Discussions Actions Security Insights

main ▾ 18 branches 55 tags Go to file Code ▾

Connor Clark	chore: bump lhci references to 0.13	✓ 083d639 3 weeks ago	⌚ 973 commits
.github	chore: bump to node 18 (#989)	3 weeks ago	
docs	chore: bump lhci references to 0.13	3 weeks ago	
packages	feat: upgrade to lighthouse 11.4.0 (#991)	3 weeks ago	
scripts	chore(deps): upgrade to storybook 7.6.4, preact 10.19.3, esbuild 0.19...	3 weeks ago	
types	feat(cli): add static dir file discovery depth flag (#940)	3 weeks ago	
.browserslistrc	refactor: split package into cli and utils	5 years ago	
.eslintrc.js	chore(deps): upgrade to storybook 7.6.4, preact 10.19.3, esbuild 0.19...	3 weeks ago	
.eslintrc.tests.js	chore(deps): update lint deps and use implicit configuration (#751)	3 years ago	
.eslintrc.ui.js	chore(deps): update lint deps and use implicit configuration (#751)	3 years ago	
.gitignore	refactor: use esbuild instead of parcel, upgrade to node 14 (#738)	3 years ago	
.lighthouserc.js	feat: upgrade to lighthouse 8.0	3 years ago	
.prettierrc	chore: update devDependencies	4 years ago	

About
Automate running Lighthouse for every commit, viewing the changes, and preventing regressions

webperf lighthouse

Readme Apache-2.0 license Activity 6.1k stars 77 watching 680 forks Report repository

Releases 55
v0.13.0 Latest 3 weeks ago + 54 releases



WebPageTest Example



PageSpeed Insights

Report from Jan 5, 2024, 9:33:07 PM

<https://bbc.com/>

Analyze

 Mobile  Desktop

Showing results for URL: <https://www.bbc.com/>
[Run with original URL](#)

 Discover what your real users are experiencing

[This URL](#) [Origin](#)



Core Web Vitals Assessment: Failed ⓘ

[Expand view](#)

● [Largest Contentful Paint \(LCP\)](#)

2.5 s

● [First Input Delay \(FID\)](#)

22 ms

■ [Cumulative Layout Shift \(CLS\)](#)

0.14

OTHER NOTABLE METRICS

● [First Contentful Paint \(FCP\)](#)

0.9 s

● [Interaction to Next Paint \(INP\)](#) ⓘ

175 ms

● [Time to First Byte \(TTFB\)](#) ⤵

0.3 s

 Latest 28-day collection period

 Various mobile devices

 Many samples ([Chrome UX Report](#))

 Full visit durations

 Various network connections

 All Chrome versions



Measuring Web Vitals in Create React App

index.js

```
reportWebVitals(console.log)

reportWebVitals(metric => {
  /* do something */
}) ;
```



Measuring Web Vitals in Next.js

_app.js

```
export function reportWebVitals(metric) {  
  console.log(metric);  
}
```



Measuring Web Vitals Using Custom Code

Use the `web-vitals` or `web-vitals-reporter` package

index.js

```
import { onCLS, onINP, onLCP } from 'web-vitals';
import { createApiReporter, getDeviceInfo } from 'web-vitals-reporter';

const report = createApiReporter('/analytics', { initial: getDeviceInfo() });

onCLS(report);
onINP(report);
onLCP(report);
```



Learn More About Web Vitals

<https://web.dev/explore/learn-core-web-vitals>

<https://github.com/GoogleChrome/web-vitals>

<https://github.com/treosh/web-vitals-reporter>



Live outro



Measuring App Rendering Performance



Live Video Intro

B-Roll footage



Dive Into the React Dev Tools

React 18 Debugging Playbook

Kamran Ayub



BBC's Open Source Web App -- Simorgh

```
bash
git clone https://github.com/bbc/simorgh
nvm use
npm install yarn -global
yarn
yarn dev
```

The screenshot shows the GitHub repository page for 'simorgh' (Public). At the top, there are buttons for 'Watch 39', 'Fork 188', and 'Star 1.2k'. Below the header, there are tabs for 'Code' (selected), 'File', 'Issue', and 'Pull request'. The main area displays a timeline of commits from AndrewAnderson01, showing various updates to the repository. On the right side, there is an 'About' section with a brief description: 'The BBC's Open Source Web Application. Contributions welcome! Used on some of our biggest websites, e.g.' followed by links to 'www.bbc.com/thai' and several tags: react, express, bbc, news, article, reactjs, and isomorphic-javascript. Below that is a sidebar with links to 'Readme', 'View license', 'Code of conduct', 'Security policy', 'Activity', '1.2k stars', '39 watching', '188 forks', and 'Report repository'. At the bottom, there is a 'Releases' section with a link to 'Release 4.1301.0' (Latest) and '+ 1,656 releases'.

simorgh Public

Watch 39 Fork 188 Star 1.2k

Code

latest 149 Branches 1,661 Tags Go to file Add file

AndrewAnderson01 Merge pull request #11211 from bbc/WSTEAM1-886-I... f906fbc · 18 hours ago 52,949 Commits

.github Add Github Actions workflow permissions (#11160) 2 months ago

.husky Less husky-ing 2 years ago

.nvm NextJS integration (#10305) last year

.storybook remove rambda changes 5 months ago

.vscode Ensure runtime version is node 16 5 months ago

.yarn WSTEAM1-498 Live experience translations (#11210) last week

3rdPartyCypress feat: add documentation to storybook 8 months ago

AdHocCypress feat: add documentation to storybook 8 months ago

codebuild renamed node module install script to decouple from yarn 2 years ago

cypress remove cypress idx persian/afghanistan test last week

data code cleanup last week

docs code cleanup last week

envConfig WSTEAM1-369: Add public prefix (#10828) 7 months ago

About

The BBC's Open Source Web Application. Contributions welcome! Used on some of our biggest websites, e.g.

www.bbc.com/thai

react express bbc news article

reactjs isomorphic-javascript

Readme

View license

Code of conduct

Security policy

Activity

1.2k stars

39 watching

188 forks

Report repository

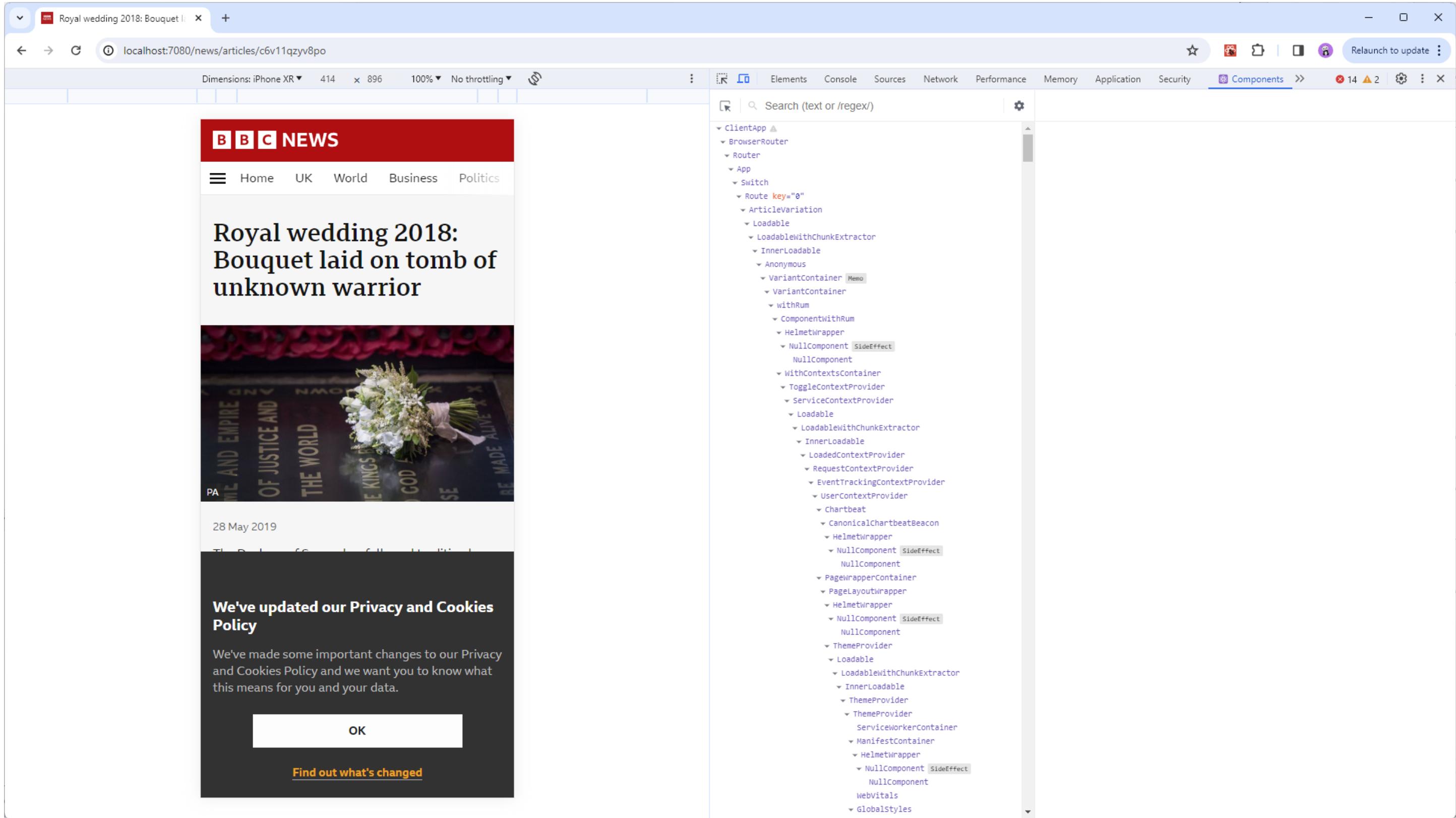
Releases 1,657

Release 4.1301.0 Latest 18 hours ago

+ 1,656 releases



Starting a Profiling Session



The screenshot shows a browser window with the BBC News article titled "Royal wedding 2018: Bouquet laid on tomb of unknown warrior". The browser's developer tools are open, specifically the Components tab, which displays a hierarchical tree of React components. The tree starts with "ClientApp" at the top, followed by "BrowserRouter", "Router", "App", "Switch", and so on, down to "GlobalStyles" at the bottom. The "Components" tab has a status bar indicating 14 tracked components and 2 tracked side effects. The BBC News article itself features a large image of a bouquet of flowers and text about the royal wedding.

Royal wedding 2018: Bouquet laid on tomb of unknown warrior

28 May 2019

We've updated our Privacy and Cookies Policy

OK

Find out what's changed

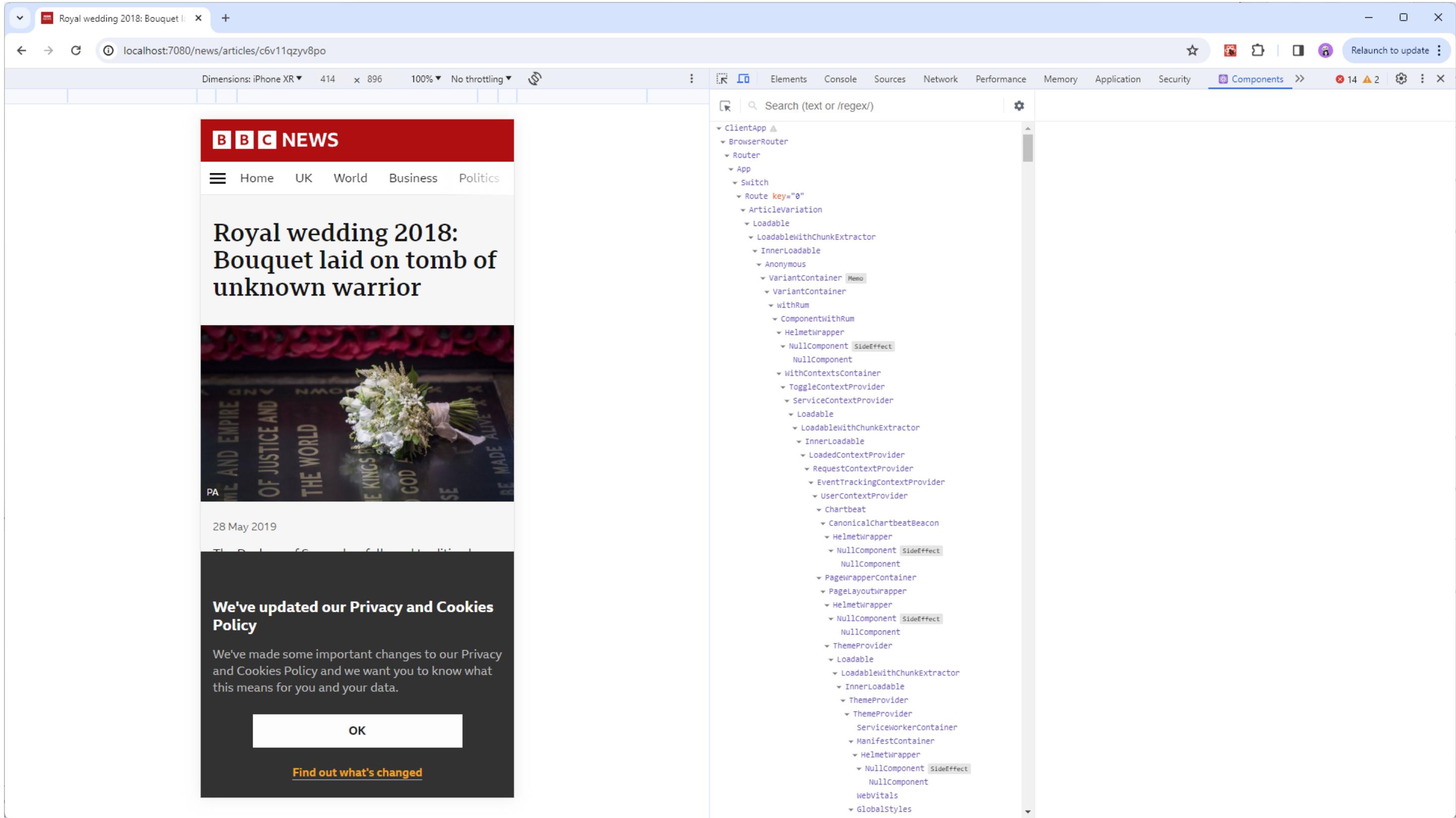
Pluralsight

Not signed in

Sync and personalize Chrome across your devices

Turn on sync...

Starting a Profiling Session



The screenshot shows a browser window with the BBC News article titled "Royal wedding 2018: Bouquet laid on tomb of unknown warrior". The browser's developer tools are open, specifically the Components tab, which displays a hierarchical tree of React components. The tree starts with "ClientApp" at the top, followed by "BrowserRouter", "Router", "App", "Switch", and so on, down to "GlobalStyles" at the bottom. The Components tab also includes a search bar and various configuration options. To the right of the browser window, the Pluralsight logo and a "Not signed in" message are visible, along with a "Sync and personalize Chrome across your devices" button.

Royal wedding 2018: Bouquet laid on tomb of unknown warrior

28 May 2019

We've updated our Privacy and Cookies Policy

OK

Find out what's changed

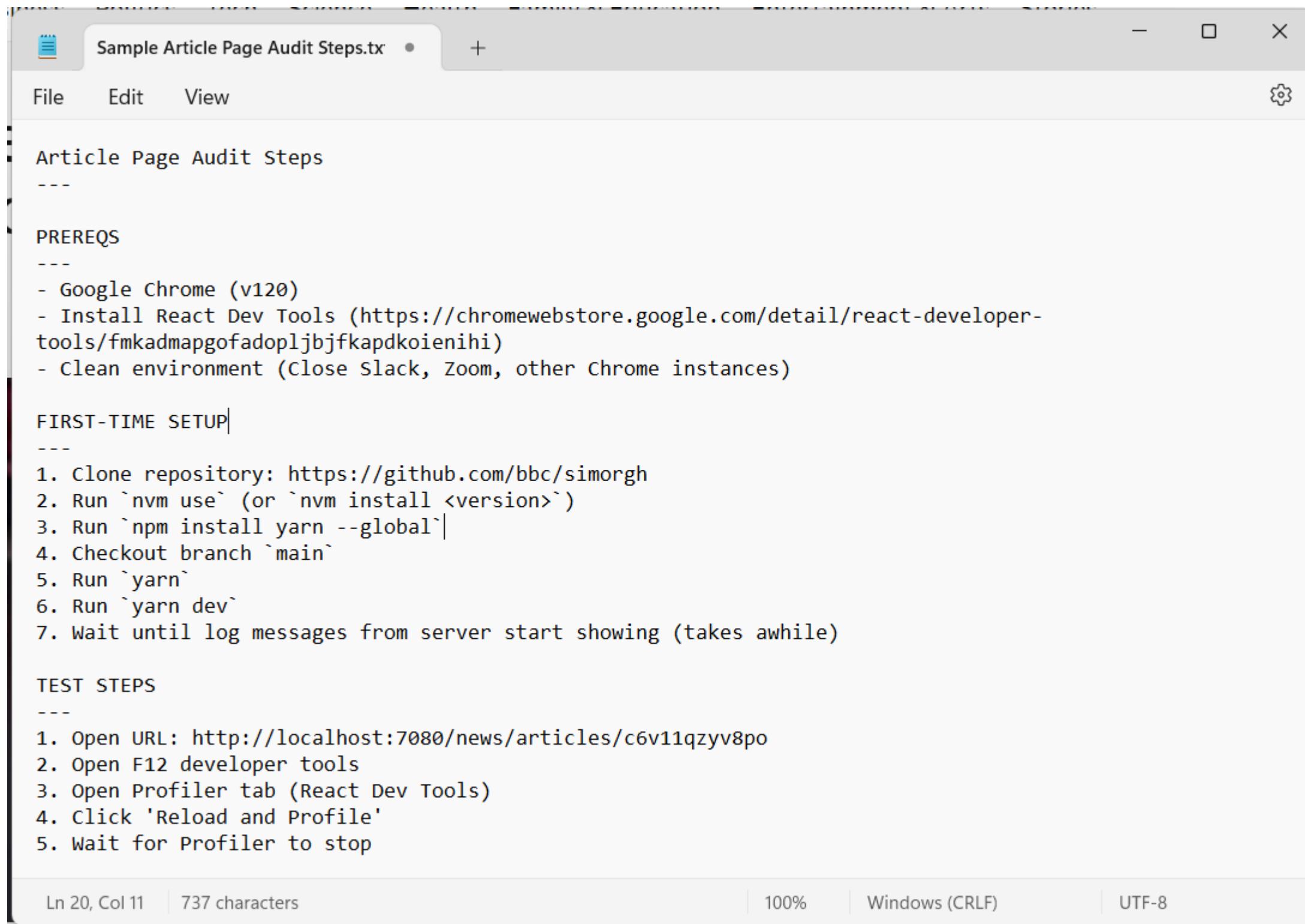
Pluralsight

Not signed in

Sync and personalize Chrome across your devices

Turn on sync...

Starting a Profiling Session



The screenshot shows a code editor window titled "Sample Article Page Audit Steps.txt". The file contains the following content:

```
Article Page Audit Steps
---

PREREQS
---
- Google Chrome (v120)
- Install React Dev Tools (https://chromewebstore.google.com/detail/react-developer-tools/fmkadmapgofadoplbjbjfkapdkoienihi)
- Clean environment (Close Slack, Zoom, other Chrome instances)

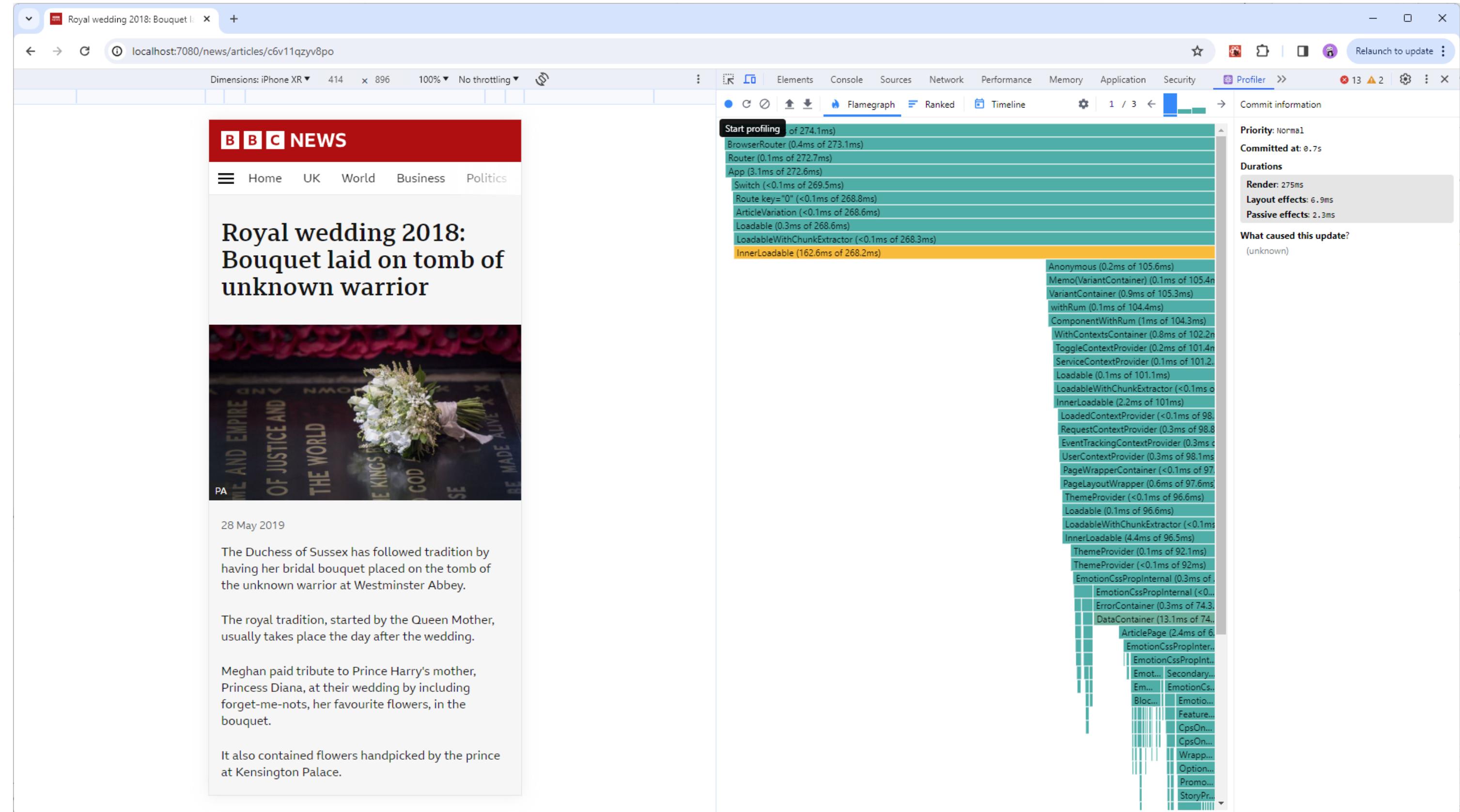
FIRST-TIME SETUP
---
1. Clone repository: https://github.com/bbc/simorgh
2. Run `nvm use` (or `nvm install <version>`)
3. Run `npm install yarn --global`
4. Checkout branch `main`
5. Run `yarn`
6. Run `yarn dev`
7. Wait until log messages from server start showing (takes awhile)

TEST STEPS
---
1. Open URL: http://localhost:7080/news/articles/c6v11qzyv8po
2. Open F12 developer tools
3. Open Profiler tab (React Dev Tools)
4. Click 'Reload and Profile'
5. Wait for Profiler to stop
```

At the bottom of the editor, the status bar shows "Ln 20, Col 11 | 737 characters | 100% | Windows (CRLF) | UTF-8".



Interpreting the Flamegraph



Looking for Whitespace

The screenshot shows a BBC News article titled "Royal wedding 2018: Bouquet laid on tomb of unknown warrior" on the left, and the Chrome DevTools Performance tab on the right.

Article Preview:

Royal wedding 2018: Bouquet laid on tomb of unknown warrior

28 May 2019

The Duchess of Sussex has followed tradition by having her bridal bouquet placed on the tomb of the unknown warrior at Westminster Abbey.

The royal tradition, started by the Queen Mother, usually takes place the day after the wedding.

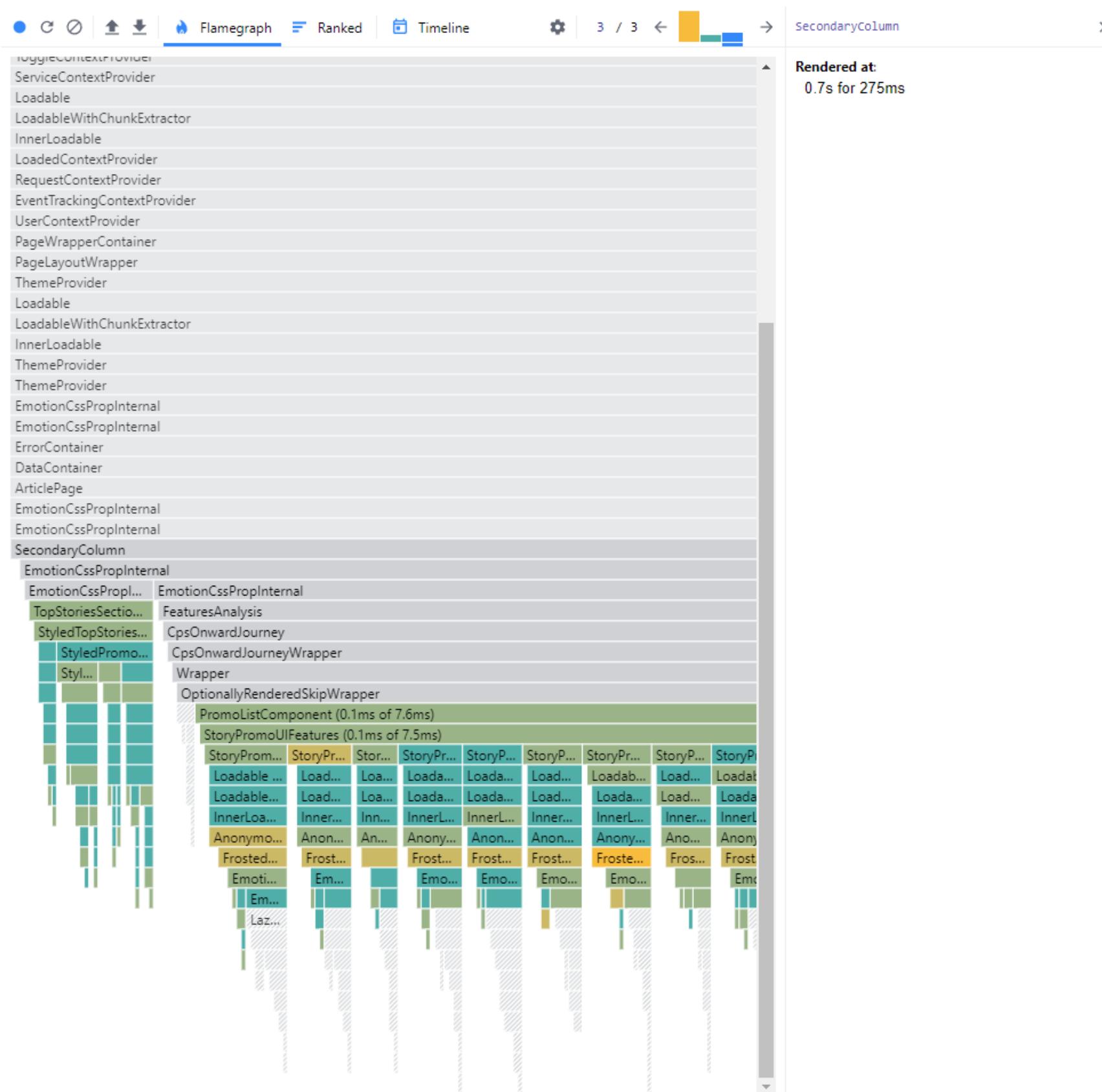
Meghan paid tribute to Prince Harry's mother, Princess Diana, at their wedding by including forget-me-nots, her favourite flowers, in the bouquet.

It also contained flowers handpicked by the prince at Kensington Palace.

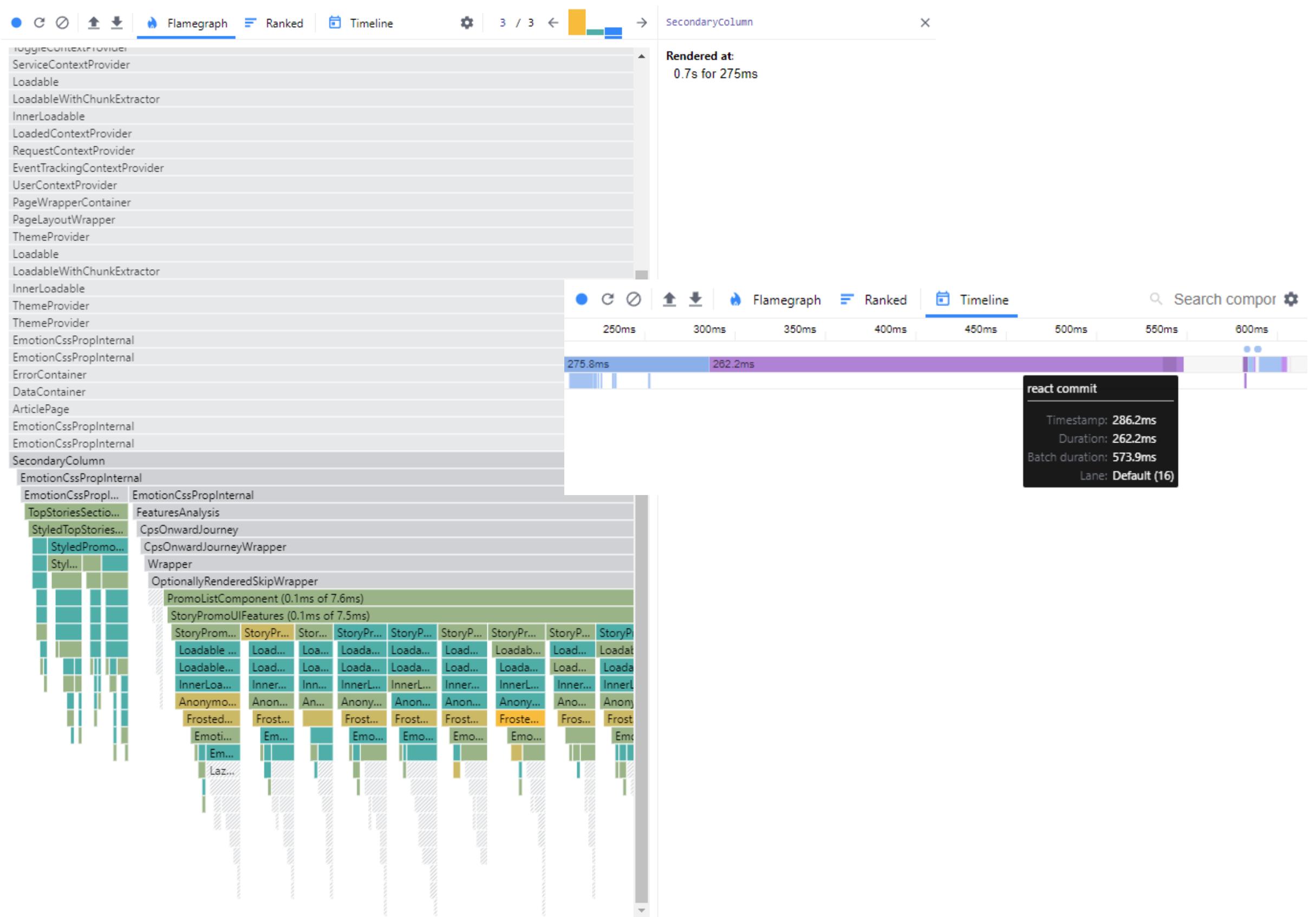
Performance Tab:

The Performance tab is set to the "Flamegraph" view. A large yellow bar at the top indicates a total duration of 274.1ms. The main flamegraph shows various components and their execution times, such as BrowserRouter (0.4ms), Router (0.1ms), App (3.1ms), Switch (<0.1ms), Route key="0" (<0.1ms), ArticleVariation (<0.1ms), Loadable (0.3ms), LoadableWithChunkExtractor (<0.1ms), and InnerLoadable (162.6ms). The "Durations" panel on the right provides summary statistics: Render: 275ms, Layout effects: 6.9ms, and Passive effects: 2.3ms. The "What caused this update?" section notes "(unknown)".

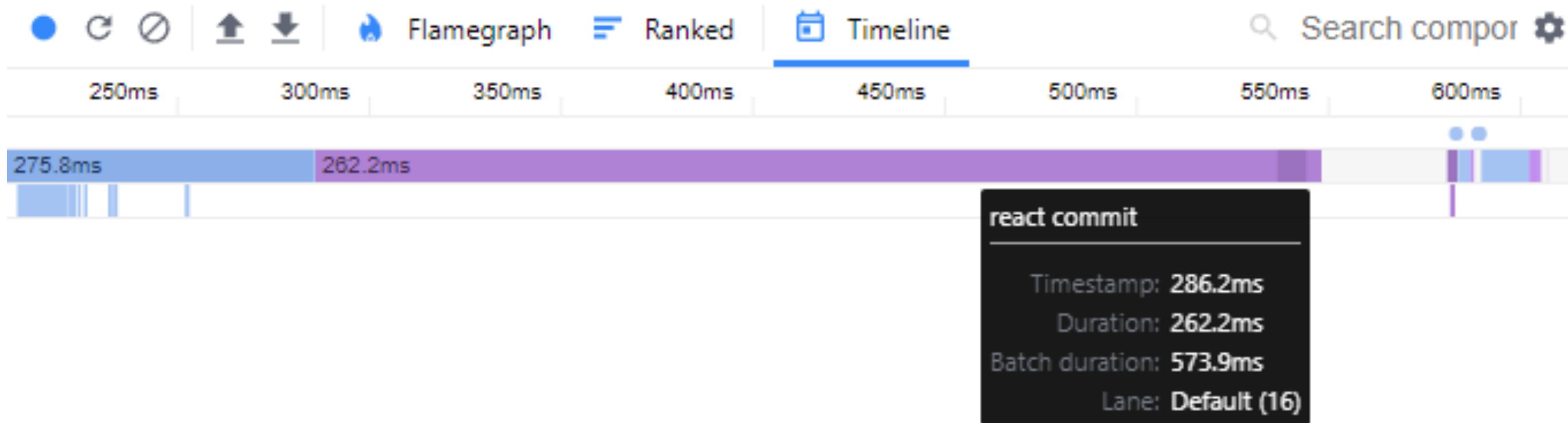
Navigating React Commits



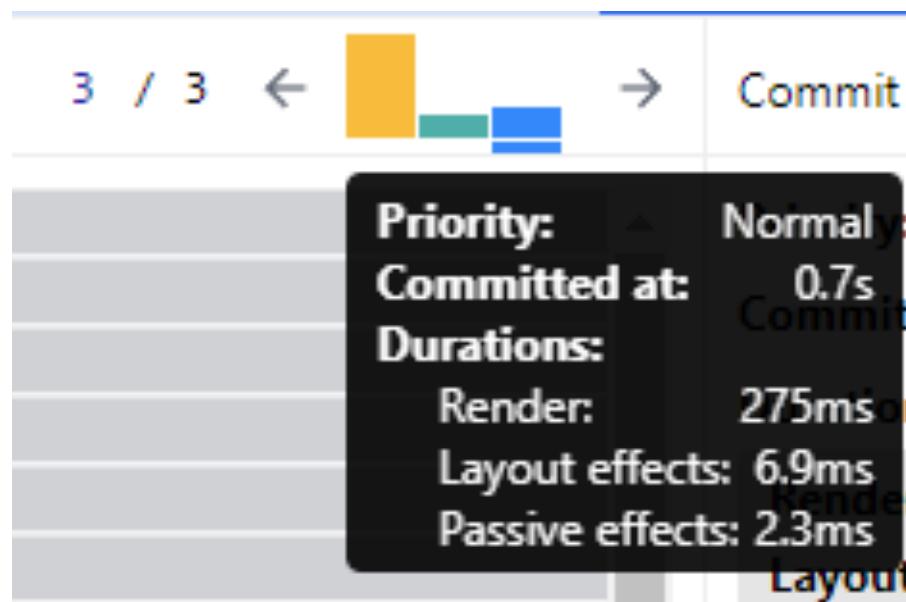
Navigating React Commits



Understanding Commit Timing



Record Top-Level Metrics



Initial mount time (first commit)

Update time (subsequent commits)

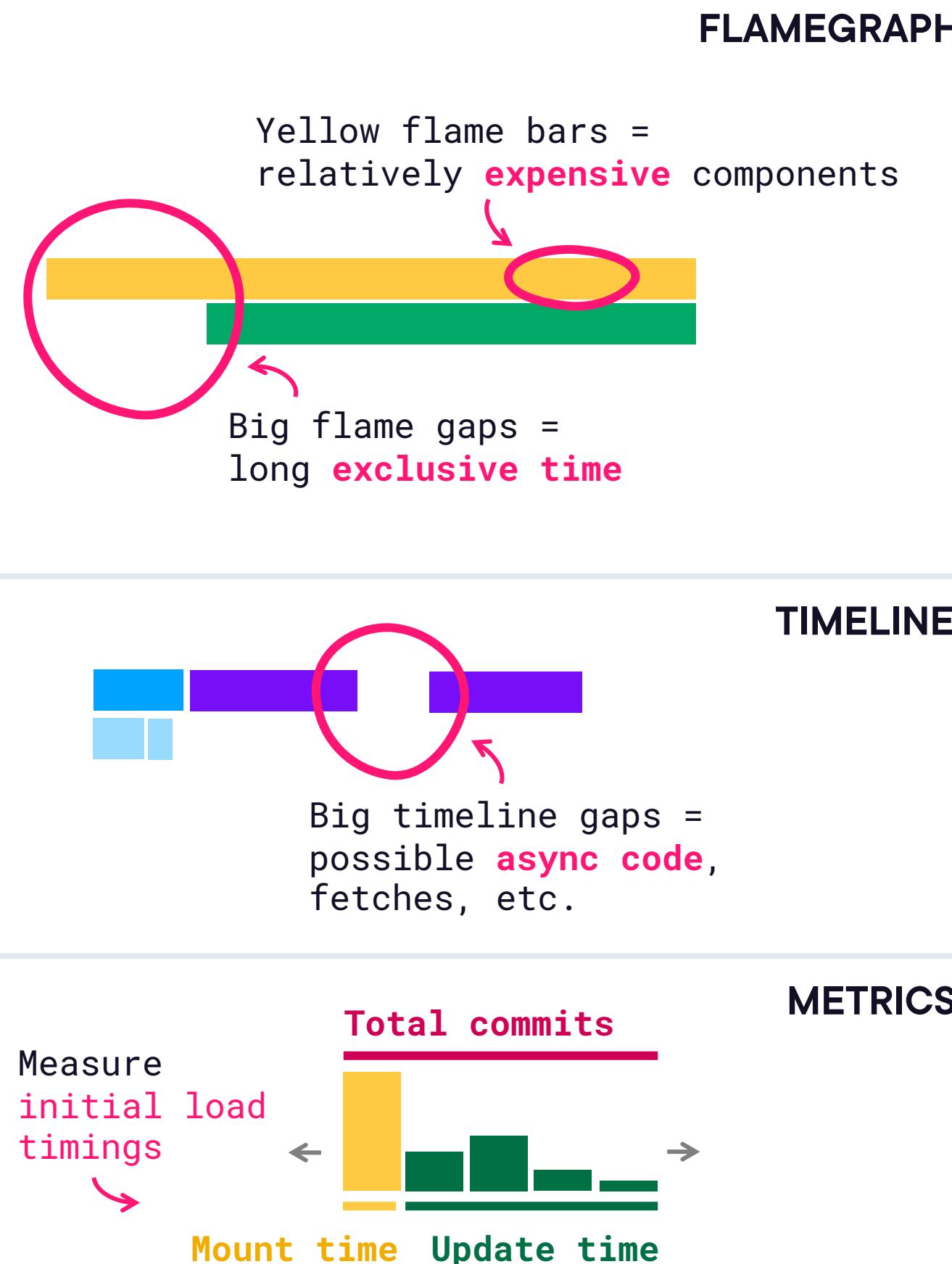
Number of commits



Live: Rinse and Repeat



React Auditing Cheat Sheet



Steps

1. Use a clean browser profile with the React Dev Tools installed
2. Open a scratchpad and your test checklist/SOP
3. Close any intensive apps that might affect measurements
4. Profile the flow, running through your steps
 - 🔴 to Profile
 - ⟳ to Reload and Profile
5. Analyze the Flamegraph and Timeline for gaps
 - 📝 Record findings, but avoid diagnosing
6. Record top-level metrics:
 1. Mount time¹
 2. Update time¹
 3. Total commits
7. Save profile in shared storage to refer to later

¹ for initial app load, mount time is the first commit and update time is subsequent commits. Otherwise, it is measured per component.



Measuring App Execution Performance



B-Roll

Live Video Intro



BBC's Open Source Web App -- Simorgh

```
bash
git clone https://github.com/bbc/simorgh
nvm use
npm install yarn -global
yarn
yarn dev
```

The screenshot shows the GitHub repository page for 'simorgh' (Public). At the top, there are buttons for 'Watch 39', 'Fork 188', and 'Star 1.2k'. Below the header, there are tabs for 'Code' (selected), 'File', 'Issue', and 'Pull request'. The main area displays a timeline of commits from AndrewAnderson01, showing various updates to the repository. On the right side, there is an 'About' section with a brief description: 'The BBC's Open Source Web Application. Contributions welcome! Used on some of our biggest websites, e.g.' followed by links to 'www.bbc.com/thai' and several tags: react, express, bbc, news, article, reactjs, and isomorphic-javascript. Below that is a sidebar with links to 'Readme', 'View license', 'Code of conduct', 'Security policy', 'Activity', '1.2k stars', '39 watching', '188 forks', and 'Report repository'. At the bottom, there is a 'Releases' section with a link to 'Release 4.1301.0' (Latest) and '+ 1,656 releases'.

simorgh Public

Watch 39 Fork 188 Star 1.2k

Code

latest 149 Branches 1,661 Tags Go to file Add file

AndrewAnderson01 Merge pull request #11211 from bbc/WSTEAM1-886-I... f906fbc · 18 hours ago 52,949 Commits

.github Add Github Actions workflow permissions (#11160) 2 months ago

.husky Less husky-ing 2 years ago

.nvm NextJS integration (#10305) last year

.storybook remove rambda changes 5 months ago

.vscode Ensure runtime version is node 16 5 months ago

.yarn WSTEAM1-498 Live experience translations (#11210) last week

3rdPartyCypress feat: add documentation to storybook 8 months ago

AdHocCypress feat: add documentation to storybook 8 months ago

codebuild renamed node module install script to decouple from yarn 2 years ago

cypress remove cypress idx persian/afghanistan test last week

data code cleanup last week

docs code cleanup last week

envConfig WSTEAM1-369: Add public prefix (#10828) 7 months ago

About

The BBC's Open Source Web Application. Contributions welcome! Used on some of our biggest websites, e.g.

www.bbc.com/thai

react express bbc news article

reactjs isomorphic-javascript

Readme

View license

Code of conduct

Security policy

Activity

1.2k stars

39 watching

188 forks

Report repository

Releases 1,657

Release 4.1301.0 Latest 18 hours ago

+ 1,656 releases

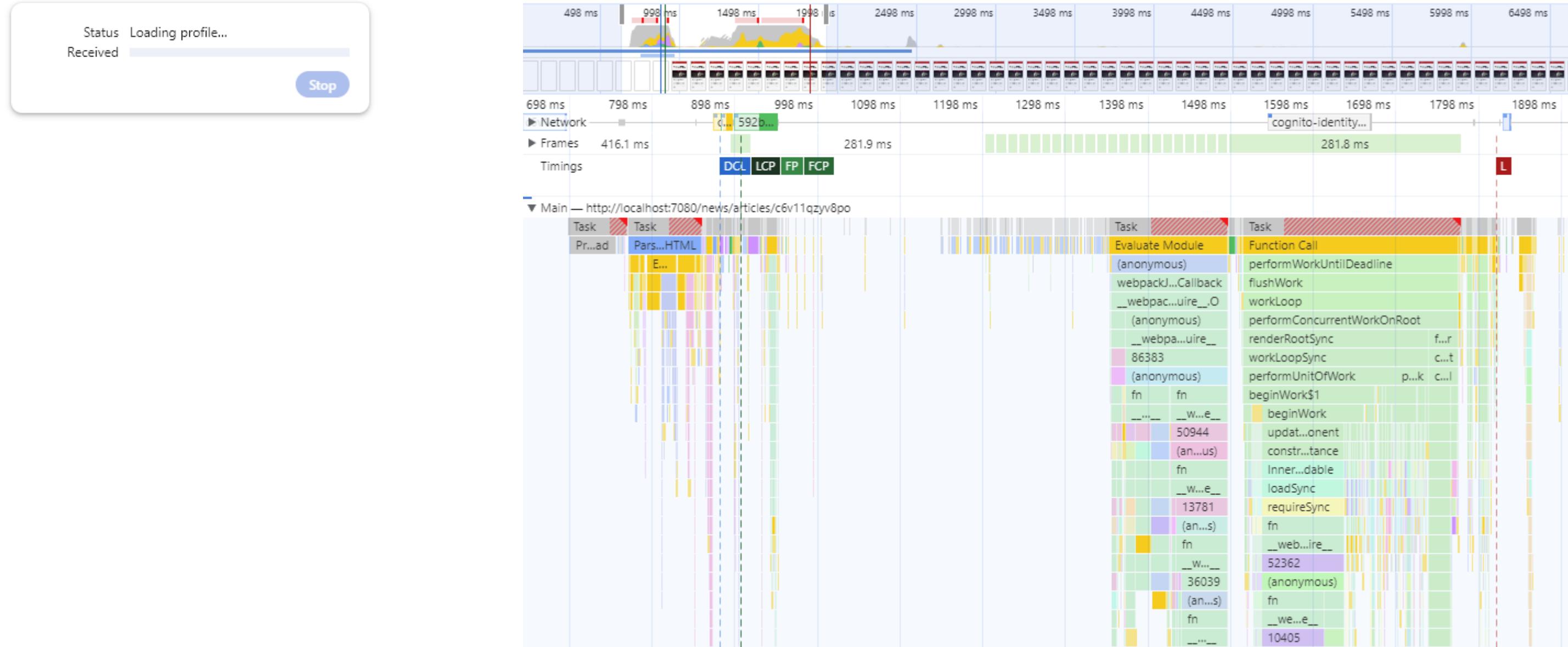


Accessing the Browser's Profiler

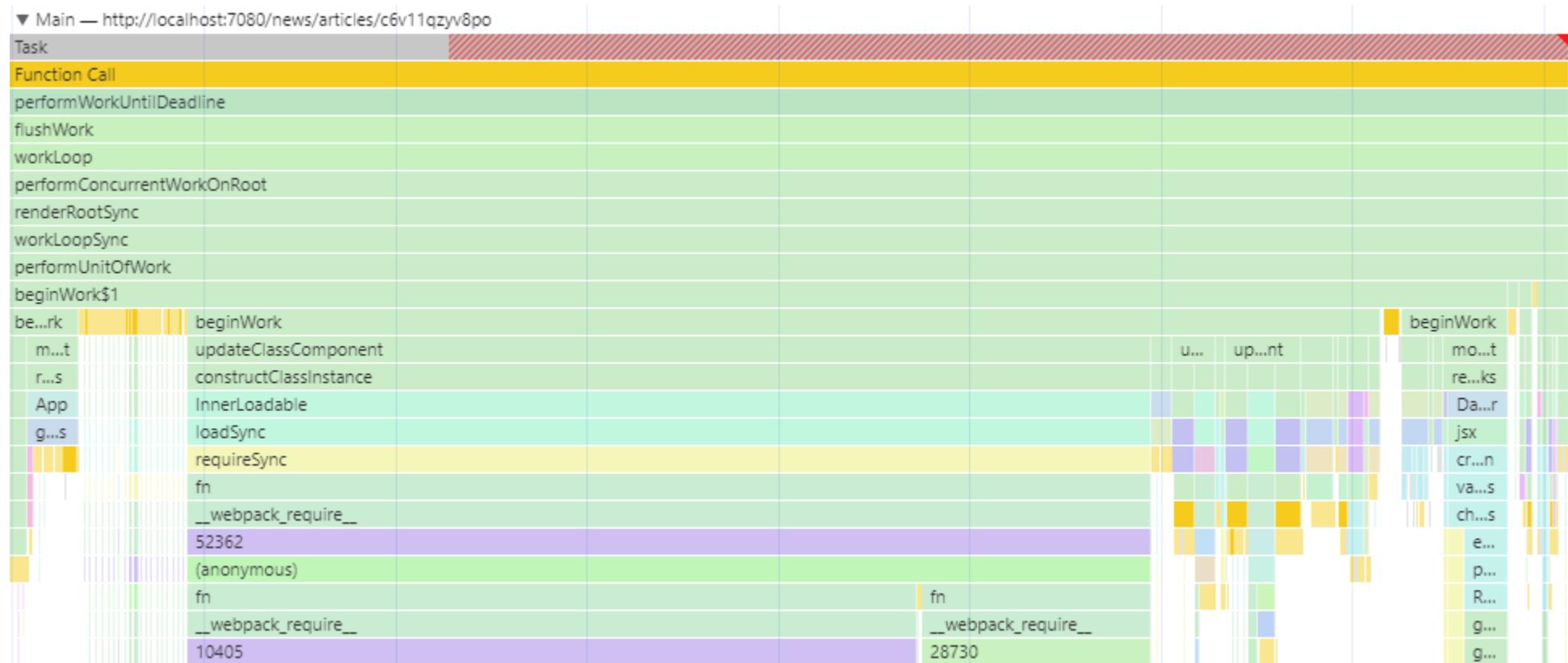
The screenshot shows a browser window with the BBC News website loaded. The URL in the address bar is `localhost:7080/news/articles/c6v11qzyv8po`. The page content is about the Royal wedding 2018 bouquet laid on the tomb of the unknown warrior. The BBC News logo and navigation menu are visible at the top. The main headline is "Royal wedding 2018: Bouquet laid on tomb of unknown warrior". Below the headline is a photograph of a white bouquet resting on a dark, inscribed plaque. The plaque has text including "PALE AND EMPIRE", "OF JUSTICE AND", "THE WORLD", "THE KING'S", "BY GOD", and "BE MADE ALIVE". The date "28 May 2019" is mentioned. The text below describes the tradition followed by the Duchess of Sussex. The developer tools are open on the right side of the browser, specifically the "Performance" tab. The tab bar also includes "Elements", "Console", "Sources", "Network", "Memory", "Application", "Security", and "Lighthouse". The "Performance" tab has a sub-menu with options like "Screenshots" and "Memory". There are also checkboxes for "Disable JavaScript samples", "CPU: No throttling", "Hardware concurrency 12", "Enable advanced paint instrumentation (slow)", and "Network: No throttling". Instructions at the bottom of the performance panel advise users to click the record button (a circle icon) or `Ctrl + E` to start recording, and to click the reload button (a circular arrow icon) or `Ctrl + Shift + E` to record the page load. It also mentions that after recording, users can select an area of interest in the overview by dragging, then zoom and pan the timeline with the mousewheel or `WASD` keys.



Recording a Performance Profile



Analyzing Performance in a React App

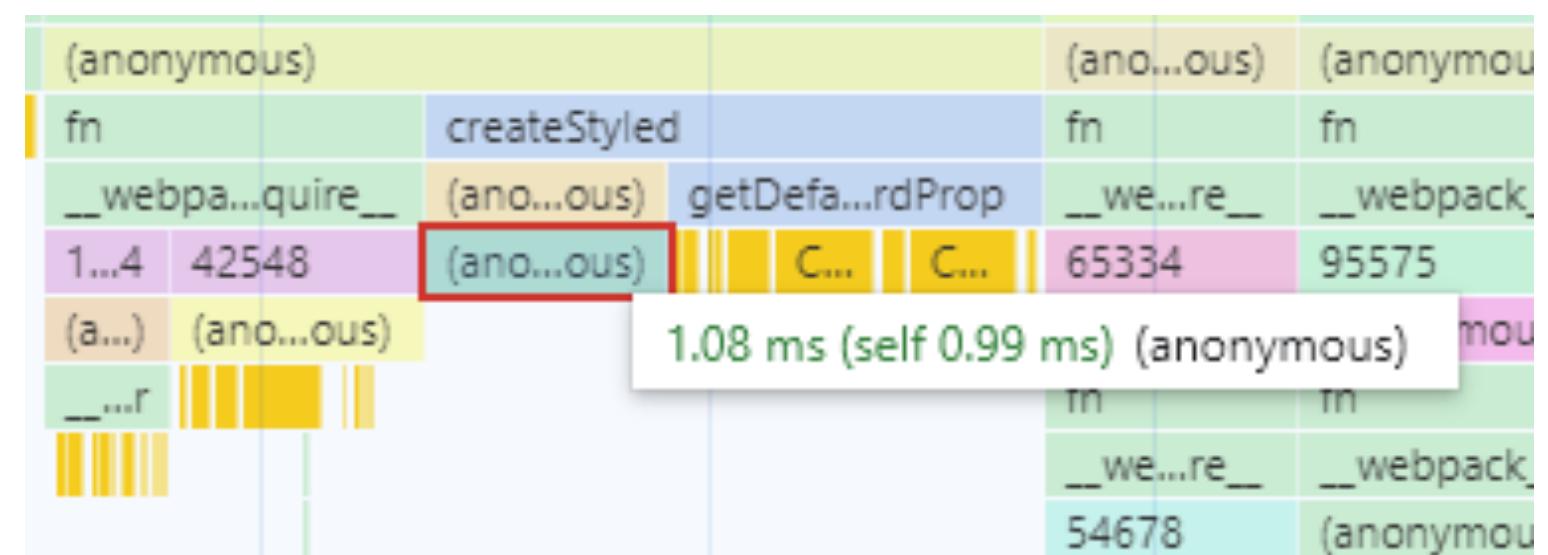


(anonymous)

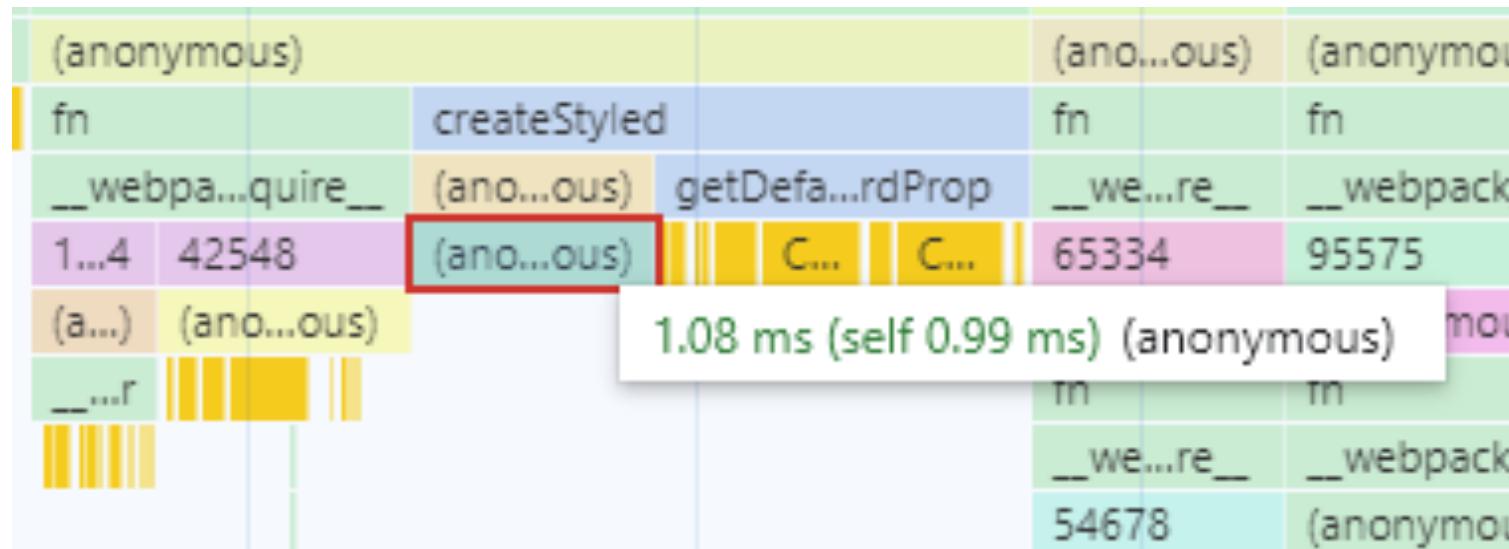
Total Time 1.08 ms

Self Time 0.99 ms

Function (anonymous) @ [emotion-is-prop-valid.esm.js:5:41](#)



Analyzing Performance in a React App

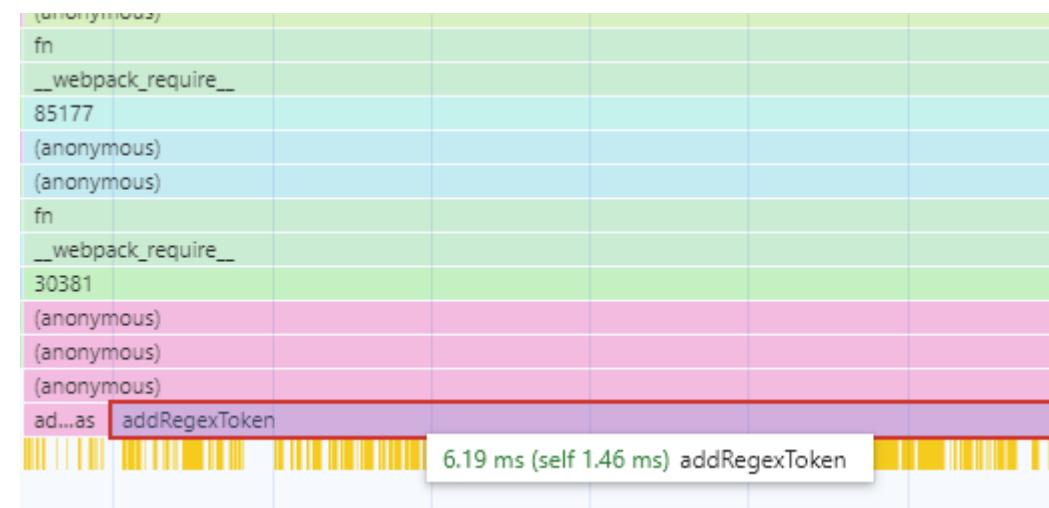
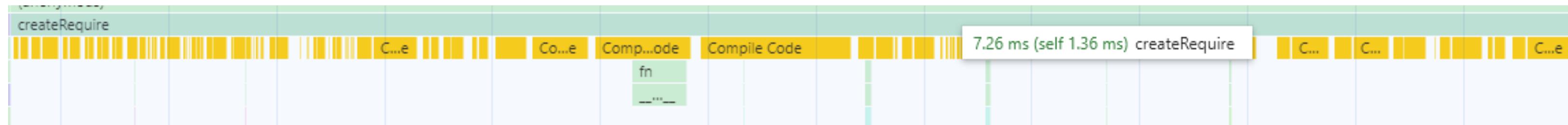


Screenshot of a code editor showing the 'emotion-is-prop-valid.esm.js' file. The code uses memoization to check if a prop is valid based on its character code. A performance annotation at line 6 indicates a duration of 1.7 ms.

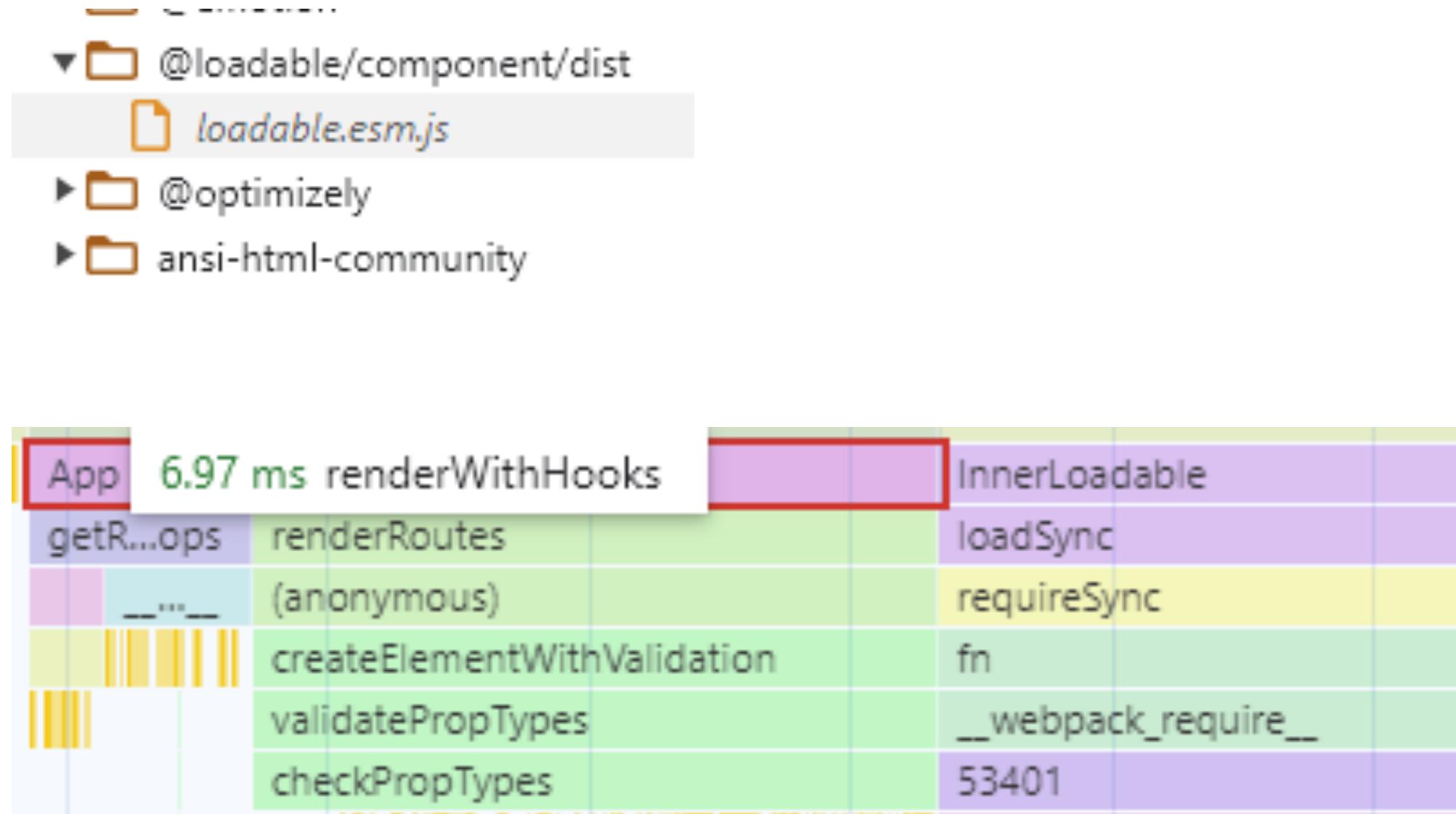
```
1 import memoize from '@emotion/memoize';
2
3 var reactPropsRegex = /^(children|dangerouslySetInnerHTML|key|ref|au
4
5 var isValidProp = /* #__PURE__ */memoize(function (prop) {
6   1.7 ms
7     return reactPropsRegex.test(prop) || prop.charCodeAt(0) === 111
8     /* o */
9     && prop.charCodeAt(1) === 110
10    /* n */
11    && prop.charCodeAt(2) < 91;
12  }
13  /* Z+1 */
14 );
15
16 export {isValidProp as default};
```



Analyzing Performance in a React App



Analyzing Performance in a React App



Console What's New Issues Search X

Aa .* file:src Loadable

index.tsx — webpack://simorgh/src/app/context/ServiceContext/index.tsx

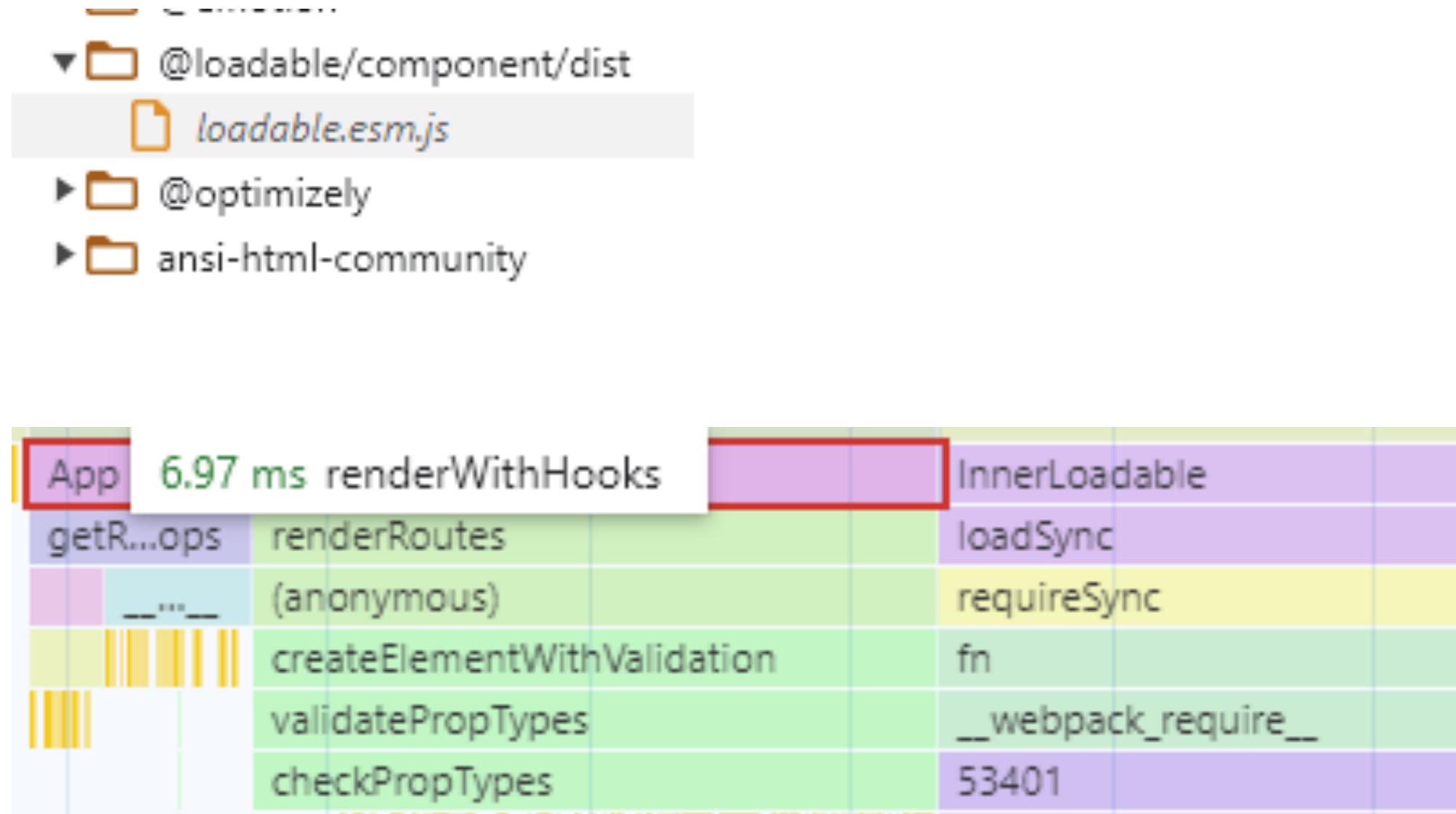
loadableConfig.ts — webpack://simorgh/src/app/lib/config/services/loadableConfig.ts

index.js — webpack://simorgh/src/app/pages/index.js

```
1 import loadable from '@loadable/component';
1 import loadable from '@loadable/component';
3 ...port const ArticlePage = loadable(() => import('./ArticlePage'));
4 ...const MediaArticlePage = loadable(() => import('./MediaArticlePage'));
5 export const ErrorPage = loadable(() => import('./ErrorPage'));
6 export const FrontPage = loadable(() => import('./FrontPage'));
7 export const HomePage = loadable(() => import('./HomePage'));
8 ...t const MediaAssetPage = loadable(() => import('./MediaAssetPage'));
9 ...ort const MostReadPage = loadable(() => import('./MostReadPage'));
10 ... const MostWatchedPage = loadable(() => import('./MostWatchedPage'));
11 ...const PhotoGalleryPage = loadable(() => import('./PhotoGalleryPage'));
12 ...rt const LiveRadioPage = loadable(() => import('./LiveRadioPage'));
13 ...onst OnDemandAudioPage = loadable(() => import('./OnDemandAudioPage'));
14 ...t const OnDemandTvPage = loadable(() => import('./OnDemandTvPage'));
15 export const TopicPage = loadable(() => import('./TopicPage'));
```



Analyzing Performance in a React App



Console What's New Issues Search X

Aa .* file:src Loadable

▶ index.tsx — webpack://simorgh/src/app/context/ServiceContext/index.tsx

▶ loadableConfig.ts — webpack://simorgh/src/app/lib/config/services/loadableConfig.ts

▼ index.js — webpack://simorgh/src/app/pages/index.js

```
1 import loadable from '@loadable/component';
1 import loadable from '@loadable/component';
3 ...port const ArticlePage = loadable(() => import('./ArticlePage'));
4 ...const MediaArticlePage = loadable(() => import('./MediaArticlePage'));
5 export const ErrorPage = loadable(() => import('./ErrorPage'));
6 export const FrontPage = loadable(() => import('./FrontPage'));
7 export const HomePage = loadable(() => import('./HomePage'));
8 ...t const MediaAssetPage = loadable(() => import('./MediaAssetPage'));
9 ...ort const MostReadPage = loadable(() => import('./MostReadPage'));
10 ... const MostWatchedPage = loadable(() => import('./MostWatchedPage'));
11 ...const PhotoGalleryPage = loadable(() => import('./PhotoGalleryPage'));
12 ...rt const LiveRadioPage = loadable(() => import('./LiveRadioPage'));
13 ...onst OnDemandAudioPage = loadable(() => import('./OnDemandAudioPage'));
14 ...t const OnDemandTvPage = loadable(() => import('./OnDemandTvPage'));
15 export const TopicPage = loadable(() => import('./TopicPage'));
```



Live: Outro



Script Profiling Cheat Sheet



Total time spent on line

27 | function test() {
28 | const prime = calculatePrime()
29 | }

SOURCE VIEW

Steps

1. Use a clean browser profile and open the Performance dev tool
2. Open a scratchpad and your test checklist/SOP
3. Close any intensive apps that might affect measurements
4. Profile the flow, running through your steps
 - 🔴 to Profile
 - ⟳ to Reload and Profile
5. Analyze the flamegraph
 - 📝 Record findings, but avoid diagnosing
6. Save profile in shared storage to refer to later



Measuring App Memory Performance



Live Intro: Memory profiling



BBC's Open Source Web App – Simorgh

```
bash
git clone https://github.com/bbc/simorgh
nvm use
npm install yarn -global
yarn
yarn dev
```

The screenshot shows the GitHub repository page for 'simorgh' (Public). At the top, there are buttons for 'Watch 39', 'Fork 188', and 'Star 1.2k'. Below the header, there are tabs for 'Code' (selected), 'File', 'Issue', and 'Pull request'. The main area displays a timeline of commits from AndrewAnderson01, showing various updates and contributions. On the right side, there is an 'About' section with a brief description: 'The BBC's Open Source Web Application. Contributions welcome! Used on some of our biggest websites, e.g.' followed by links to 'www.bbc.com/thai' and several tags: react, express, bbc, news, article, reactjs, and isomorphic-javascript. Below that is a sidebar with links to 'Readme', 'View license', 'Code of conduct', 'Security policy', 'Activity', '1.2k stars', '39 watching', '188 forks', and 'Report repository'. At the bottom, there is a 'Releases' section with a link to 'Release 4.1301.0' (Latest) and '+ 1,656 releases'.

simorgh Public

Watch 39 Fork 188 Star 1.2k

Code

latest 149 Branches 1,661 Tags Go to file Add file

AndrewAnderson01 Merge pull request #11211 from bbc/WSTEAMA-886-I... f906fbc · 18 hours ago 52,949 Commits

.github Add Github Actions workflow permissions (#11160) 2 months ago

.husky Less husky-ing 2 years ago

.nvm NextJS integration (#10305) last year

.storybook remove rambda changes 5 months ago

.vscode Ensure runtime version is node 16 5 months ago

.yarn WSTEAM1-498 Live experience translations (#11210) last week

3rdPartyCypress feat: add documentation to storybook 8 months ago

AdHocCypress feat: add documentation to storybook 8 months ago

codebuild renamed node module install script to decouple from yarn 2 years ago

cypress remove cypress idx persian/afghanistan test last week

data code cleanup last week

docs code cleanup last week

envConfig WSTEAM1-369: Add public prefix (#10828) 7 months ago

About

The BBC's Open Source Web Application. Contributions welcome! Used on some of our biggest websites, e.g.

www.bbc.com/thai

react express bbc news article

reactjs isomorphic-javascript

Readme

View license

Code of conduct

Security policy

Activity

1.2k stars

39 watching

188 forks

Report repository

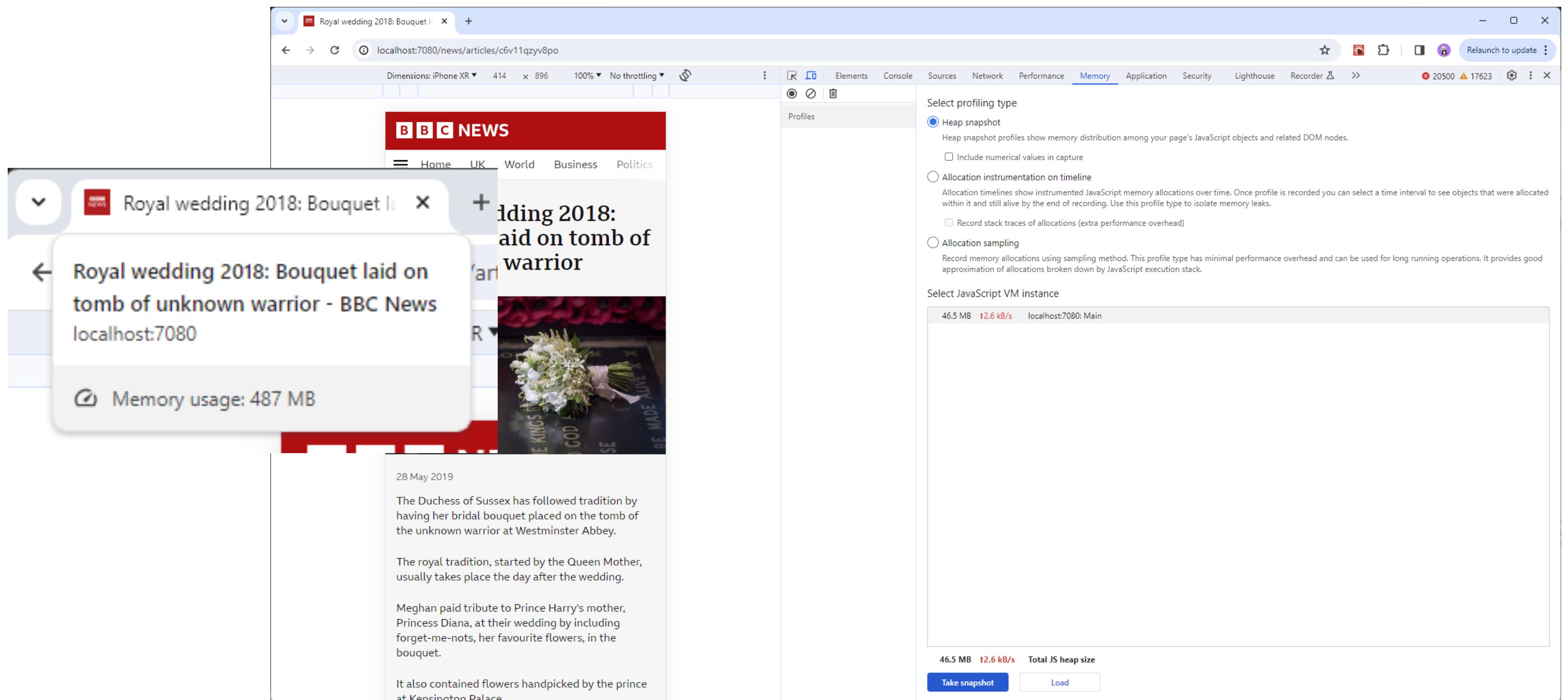
Releases 1,657

Release 4.1301.0 Latest 18 hours ago

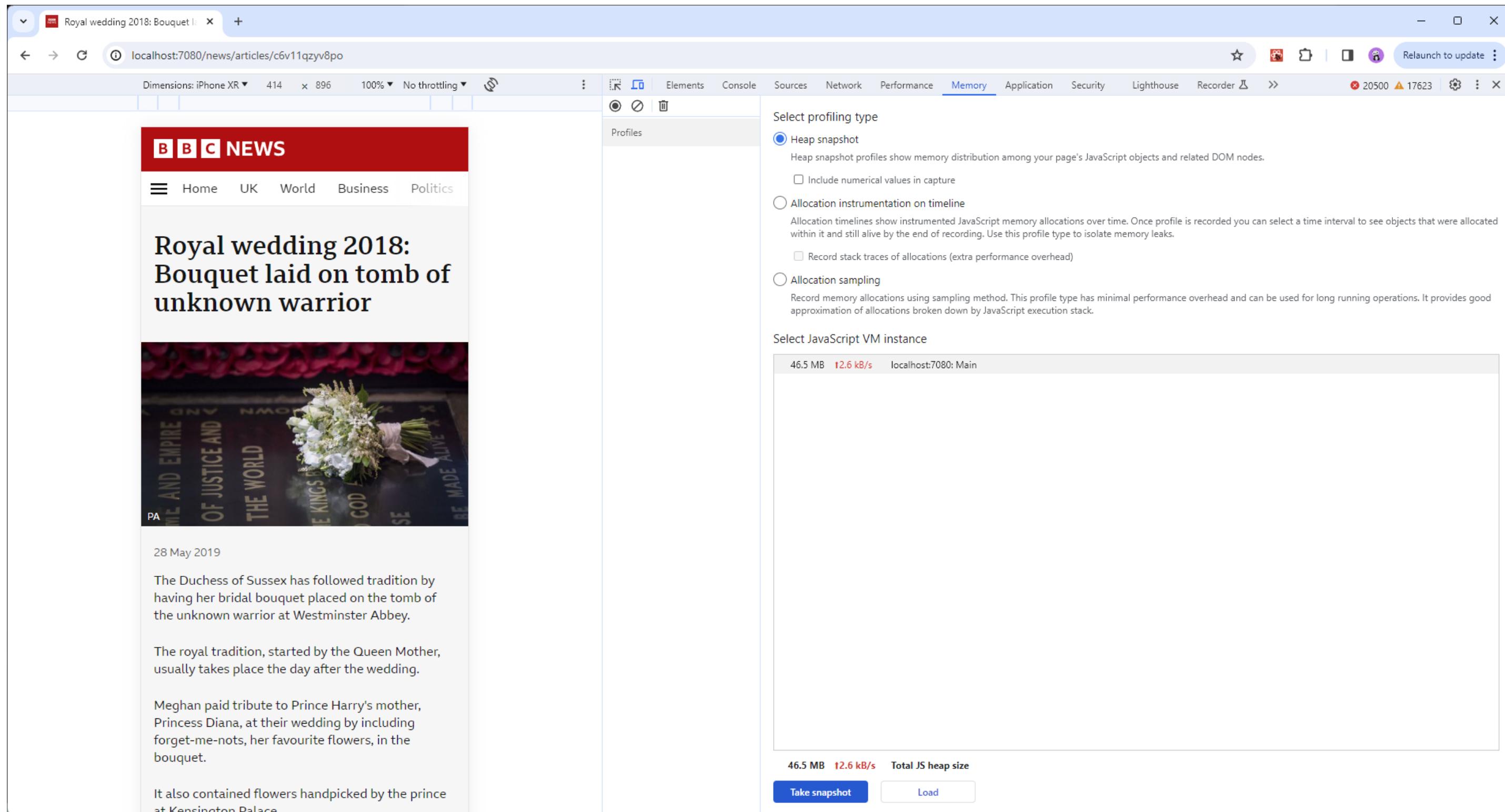
+ 1,656 releases



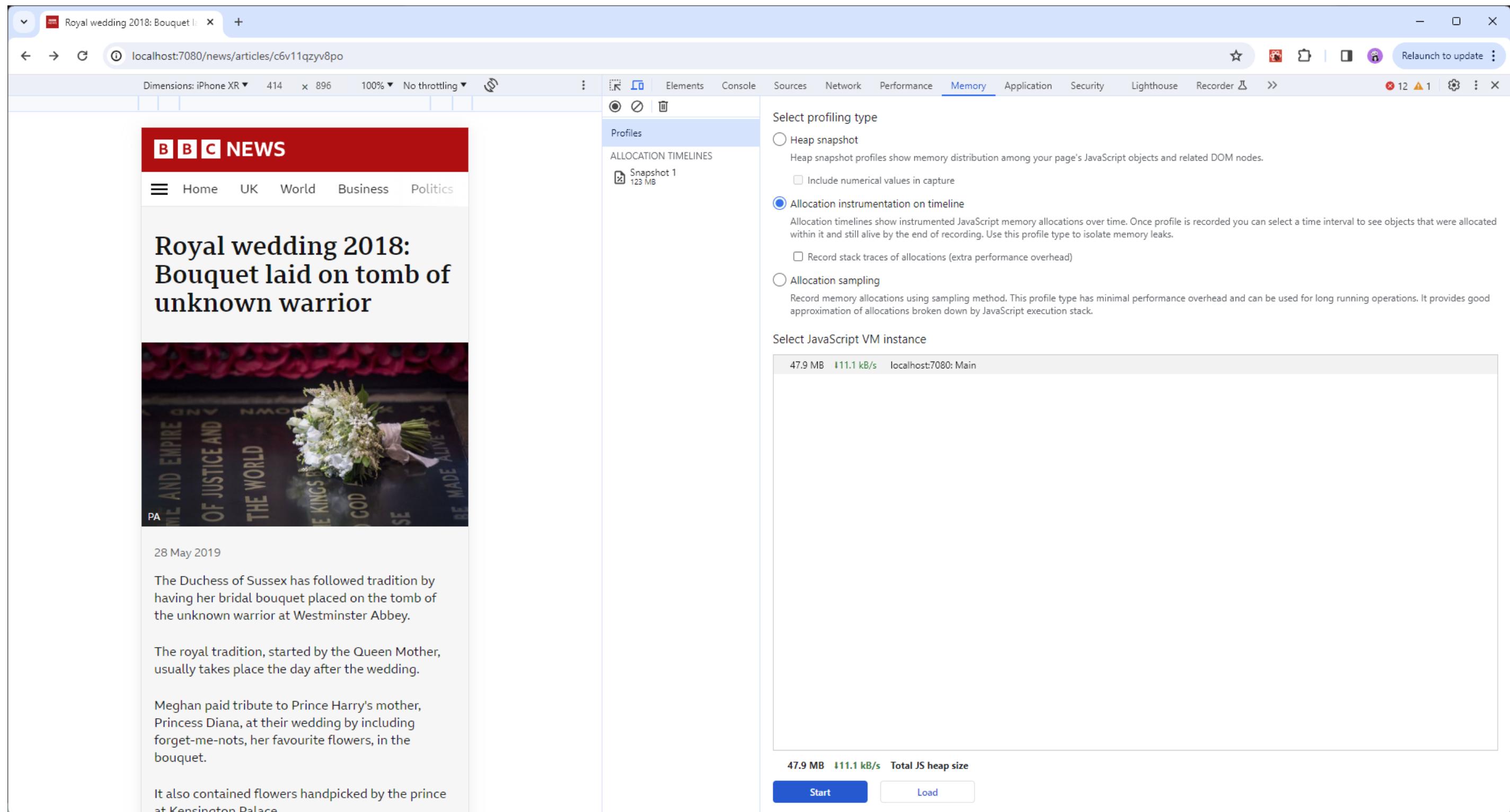
Accessing the Browser's Profiler



Accessing the Browser's Profiler



Recording a Memory Profile



Analyzing Memory Usage in a React App

The screenshot shows the Chrome DevTools Memory tab analyzing a BBC News article. The left pane displays the article content with several text blocks containing bold, italic, and both styles. The right pane shows two memory snapshots (Snapshot 1 at 123 MB and Snapshot 2 at 131 MB) and an allocation timeline from 5.00 s to 55.00 s. The constructor table lists memory usage by object, and the retainers table shows which objects keep others alive.

This is a headline block - it contains *italics*, words en Francais, and an italic word in *Cymraeg*

This is a subheadline block - it is very similar to the headline block and can also contain *italics*, words en Francais, and an italic word in *Cymraeg*

This is a text block.

This is a second paragraph inside the same text block.

This third paragraph has fragments which are **bold**, *italic*, and **both** at the same time.

This fourth paragraph contains a hyperlink to [BBC Weather](#).

This fifth paragraph contains a hyperlink with words in **bold** and *italic* and **both**.

Object	Distance	Shallow Size	Retained Size
5 in @13647271	9	72 0 %	1 722 236 1 %
script in FiberNode @13287093	8	44 0 %	1 932 0 %
shared in FiberNode() @13305579 □	7	32 0 %	7 324 0 %
constructor in Object @13305581 □	6	28 0 %	152 0 %
__proto__ in FiberNode @13652701 □	5	132 0 %	408 0 %
__reactFiber\$c3qxs1hekbt in HTMLButtonElement @13577611 □	4	212 0 %	736 0 %
[1] in InternalNode @137126 □	3	0 0 %	0 0 %
[44] in HTMLDocument @13106791 □	2	2 900 0 %	13 560 0 %

Analyzing Memory Usage in a React App

The screenshot shows the Chrome DevTools Memory tab analyzing a BBC News article. The left pane displays the article content with several text blocks containing bold, italic, and both styles. The right pane shows two memory snapshots (Snapshot 1 at 123 MB and Snapshot 2 at 131 MB) and an allocation timeline from 5.00 s to 55.00 s. The constructor table lists memory usage by object, and the retainers table shows which objects keep others alive.

This is a headline block - it contains *italics*, words en Francais, and an italic word in *Cymraeg*

This is a subheadline block - it is very similar to the headline block and can also contain *italics*, words en Francais, and an italic word in *Cymraeg*

This is a text block.

This is a second paragraph inside the same text block.

This third paragraph has fragments which are **bold**, *italic*, and **both** at the same time.

This fourth paragraph contains a hyperlink to [BBC Weather](#).

This fifth paragraph contains a hyperlink with words in **bold** and *italic* and **both**.

Object	Distance	Shallow Size	Retained Size
5 in @13647271	9	72 0 %	1 722 236 1 %
script in FiberNode @13287093	8	44 0 %	1 932 0 %
shared in FiberNode() @13305579 □	7	32 0 %	7 324 0 %
constructor in Object @13305581 □	6	28 0 %	152 0 %
__proto__ in FiberNode @13652701 □	5	132 0 %	408 0 %
__reactFiber\$c3qxs1hekbt in HTMLButtonElement @13577611 □	4	212 0 %	736 0 %
[1] in InternalNode @137126 □	3	0 0 %	0 0 %
[44] in HTMLDocument @13106791 □	2	2 900 0 %	13 560 0 %



Analyzing Memory Usage in a React App

The screenshot shows the Chrome DevTools Memory tab analyzing a BBC News article. The left pane displays the article content with several text blocks containing bold, italic, and both styles. The right pane shows two memory snapshots (Snapshot 1 at 123 MB and Snapshot 2 at 131 MB) and an allocation timeline from 5.00 s to 55.00 s. The constructor table lists memory usage by object, and the retainers table shows which objects keep others alive.

This is a headline block - it contains *italics*, words en Francais, and an italic word in *Cymraeg*

This is a subheadline block - it is very similar to the headline block and can also contain *italics*, words en Francais, and an italic word in *Cymraeg*

This is a text block.

This is a second paragraph inside the same text block.

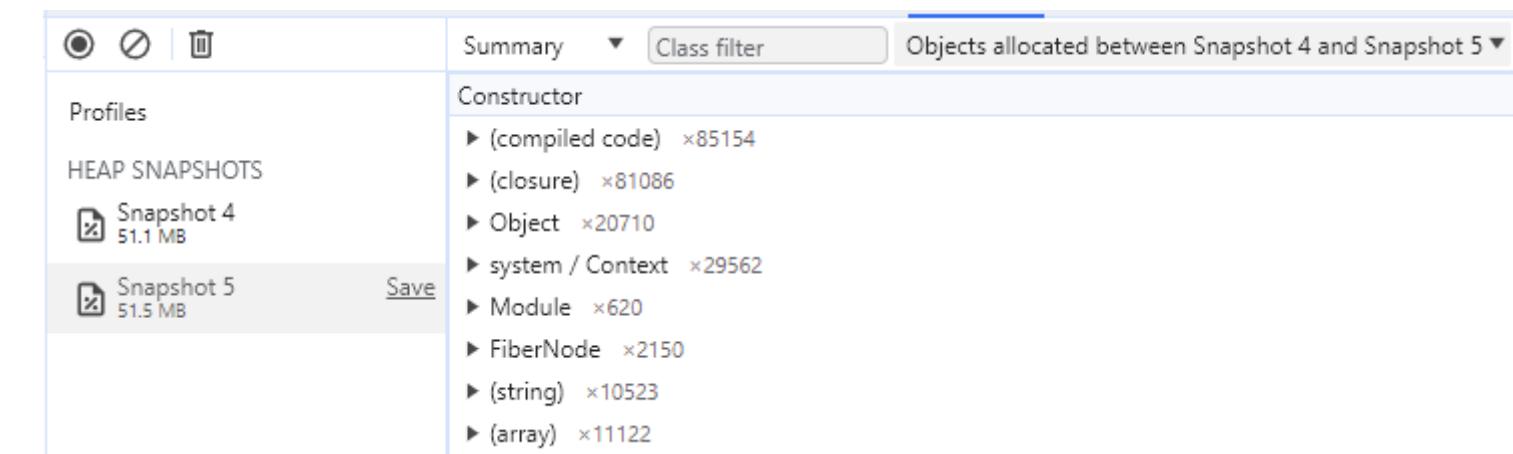
This third paragraph has fragments which are **bold**, *italic*, and **both** at the same time.

This fourth paragraph contains a hyperlink to [BBC Weather](#).

This fifth paragraph contains a hyperlink with words in **bold** and *italic* and **both**.

Object	Distance	Shallow Size	Retained Size
5 in @13647271	9	72 0 %	1 722 236 1 %
script in FiberNode @13287093	8	44 0 %	1 932 0 %
shared in FiberNode() @13305579 □	7	32 0 %	7 324 0 %
constructor in Object @13305581 □	6	28 0 %	152 0 %
__proto__ in FiberNode @13652701 □	5	132 0 %	408 0 %
__reactFiber\$c3qxs1hekbt in HTMLButtonElement @13577611 □	4	212 0 %	736 0 %
[1] in InternalNode @137126 □	3	0 0 %	0 0 %
[44] in HTMLDocument @13106791 □	2	2 900 0 %	13 560 0 %

Analyzing Memory Usage in a React App



Loading and Saving Profiles

The screenshot shows the Chrome DevTools Memory tab open, displaying two memory snapshots: Snapshot 1 (123 MB) and Snapshot 2 (133 MB). The allocations timeline shows memory usage over time, with a significant jump from Snapshot 1 to Snapshot 2 occurring around 20.00 seconds. The detailed view shows various memory blocks, their addresses, sizes, and retention details. The left side of the screen displays a BBC News article page with several text blocks containing bold, italic, and both styles.

This is a headline block - it contains *italics*, words en Francais, and an italic word in *Cymraeg*

9 October 2020

This is a subheadline block - it is very similar to the headline block and can also contain *italics*, words en Francais, and an italic word in *Cymraeg*

This is a text block.

This is a second paragraph inside the same text block.

This third paragraph has fragments which are **bold**, *italic*, and **both** at the same time.

This fourth paragraph contains a hyperlink to [BBC Weather](#).

This fifth paragraph contains a hyperlink with words in **bold** and *italic* and **both**.

Object	Distance	Shallow Size	Retained Size
5 in @13647271	9	72 0 %	1 722 236 1 %
script in FiberNode @13287093	8	44 0 %	1 932 0 %
shared in FiberNode() @13305579 □	7	32 0 %	7 324 0 %
constructor in Object @13305581 □	6	28 0 %	152 0 %
__proto__ in FiberNode @13652701 □	5	132 0 %	408 0 %
__reactFiber\$c3qxs1hekbt in HTMLButtonElement @13577611 □	4	212 0 %	736 0 %
[1] in InternalNode @137126 □	3	0 0 %	0 0 %
[44] in HTMLDocument @13106791 □	2	2 900 0 %	13 560 0 %



Live: Outro



Memory Profiling Cheat Sheet

CONSTRUCTORS

IF Retained Size is **much bigger** than Shallow Size THEN

Inspect tree

RETAINERS
Objects that are preventing memory from being released

ParentFunction

RetainingFunction1

RetainingFunction2

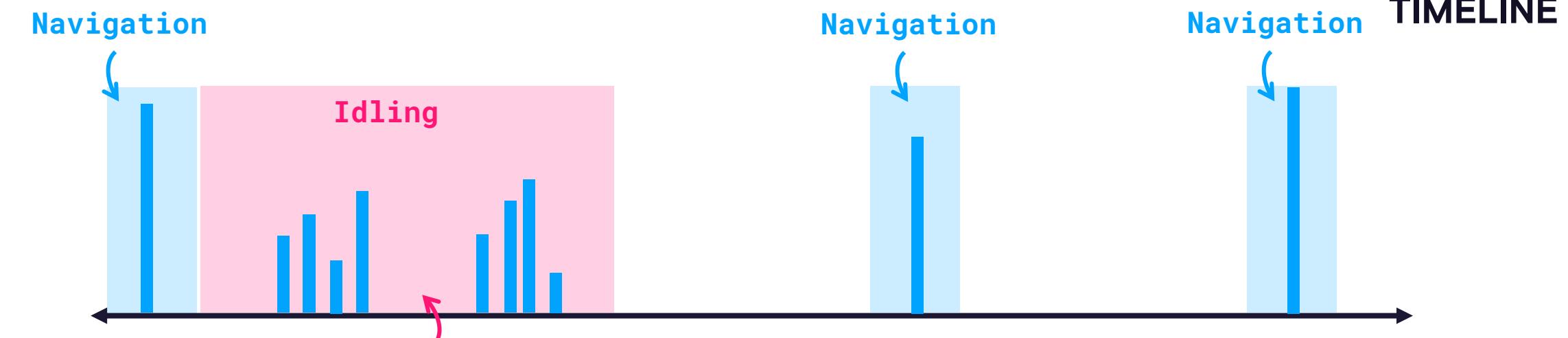
FurtherRetainingFunctionN

👀 **Look suspicious?**

SNAPSHOTS



Compare allocations since A



Steps

1. Use a clean browser profile and open Memory dev tool
2. Open a scratchpad and your test checklist/SOP
3. Close any intensive apps that might affect measurements
4. Profile the flow, running through your steps
 - Heap Snapshot to analyze memory at a point-in-time
 - Profile Instrumentation to analyze memory usage over time
5. Analyze the Constructors and Retainers tree
 - Record findings, but avoid diagnosing
6. Save profile in shared storage to refer to later



Measuring Render Time with <Profiler />



```
<Profiler onRender={handleOnRender}>  
  <SearchPage />  
</Profiler>
```



How to Use the React <Profiler> Component

Import the <Profiler /> component and include in the app

index.jsx

```
import { render, Profiler } from 'react';

render(
  <Profiler id="App" onRender={handleOnRender}>
    <App />
  </Profiler>
);
```



How to Use the React <Profiler> Component

Wrap the <Profiler /> around parts of your app you want to measure

SearchPage.jsx

```
import { Profiler } from 'react';

export const SearchPage = () => (
  <Layout title="Search">
    <SearchBar />
    <Profiler id="SearchFacets" onRender={handleOnRender}>
      <SearchFacets />
    </Profiler>
    <SearchResults />
  </Layout>
);
```



How to Use the React <Profiler> Component

You can even nest profilers to instrument different parts of the app

SearchPage.jsx

```
import { Profiler } from 'react';

export const SearchPage = () => (
  <Layout title="Search">
    <Profiler id="Search" onRender={handleOnRender}>
      <SearchBar />
      <Profiler id="SearchFacets" onRender={handleOnRender}>
        <SearchFacets />
      </Profiler>
      <SearchResults />
    </Profiler>
  </Layout>
);
```



Profiler onRender callback

<https://react.dev/reference/react/Profiler>

```
function onRender(  
  id: string,  
  phase: "mount" | "update" | "nested-update",  
  actualDuration: number,  
  baseDuration: number,  
  startTime: number,  
  commitTime: number)  
{  
  // Aggregate or log render timings...  
}
```



BBC's Open Source Web App -- Simorgh

```
bash
git clone https://github.com/bbc/simorgh
nvm use
npm install yarn -global
yarn
yarn dev
```

The screenshot shows the GitHub repository page for 'simorgh' (Public). At the top, there are buttons for 'Watch 39', 'Fork 188', and 'Star 1.2k'. Below the header, there are tabs for 'Code' (selected), 'File', 'Issue', and 'Pull request'. The main area displays a timeline of commits from AndrewAnderson01, showing various updates to the repository. On the right side, there is an 'About' section with a brief description: 'The BBC's Open Source Web Application. Contributions welcome! Used on some of our biggest websites, e.g.' followed by links to 'www.bbc.com/thai' and several tags: react, express, bbc, news, article, reactjs, and isomorphic-javascript. Below that is a sidebar with links to 'Readme', 'View license', 'Code of conduct', 'Security policy', 'Activity', '1.2k stars', '39 watching', '188 forks', and 'Report repository'. At the bottom, there is a 'Releases' section with a link to 'Release 4.1301.0' (Latest) and '+ 1,656 releases'.

simorgh Public

Watch 39 Fork 188 Star 1.2k

Code

latest 149 Branches 1,661 Tags Go to file Add file

AndrewAnderson01 Merge pull request #11211 from bbc/WSTEAM1-886-I... f906fbc · 18 hours ago 52,949 Commits

.github Add Github Actions workflow permissions (#11160) 2 months ago

.husky Less husky-ing 2 years ago

.nvm NextJS integration (#10305) last year

.storybook remove rambda changes 5 months ago

.vscode Ensure runtime version is node 16 5 months ago

.yarn WSTEAM1-498 Live experience translations (#11210) last week

3rdPartyCypress feat: add documentation to storybook 8 months ago

AdHocCypress feat: add documentation to storybook 8 months ago

codebuild renamed node module install script to decouple from yarn 2 years ago

cypress remove cypress idx persian/afghanistan test last week

data code cleanup last week

docs code cleanup last week

envConfig WSTEAM1-369: Add public prefix (#10828) 7 months ago

About

The BBC's Open Source Web Application. Contributions welcome! Used on some of our biggest websites, e.g.

www.bbc.com/thai

react express bbc news article

reactjs isomorphic-javascript

Readme

View license

Code of conduct

Security policy

Activity

1.2k stars

39 watching

188 forks

Report repository

Releases 1,657

Release 4.1301.0 Latest 18 hours ago

+ 1,656 releases



Implementing <Profiler> in React App

```
index.jsx  X
src > app > legacy > containers > App > index.jsx
  1  /* eslint-disable */
  2  /*
  3   * © Jordan Tart https://github.com/jtart
  4   * https://github.com/jtart/react-universal-app
  5   */
  6 import React from 'react';
  7 import { StaticRouter, BrowserRouter } from 'react-router-dom';
  8 import App from './App';
  9
 10 export class ClientApp extends React.Component {
 11   // Having an error boundary here means that if hydration fails, users are left with the server-rendered
 12   // DOM
 13   // Without this, DOM would be removed if hydration fails, leaving users with a blank white page
 14   componentDidCatch() {}
 15
 16   render() {
 17     return (
 18       <BrowserRouter {...this.props}>
 19         <React.Profiler id="App" onRender={this.handleOnRender}>
 20           <App initData={this.props.data} />
 21         </React.Profiler>
 22       </BrowserRouter>
 23     );
 24   }
 25
 26   handleOnRender(id, phase, actualDuration, baseDuration, startTime, commitTime) {
 27     console.log({ id, phase, actualDuration, baseDuration, startTime, commitTime })
 28   }
 29
 30   export const ServerApp = props => (
 31     <StaticRouter {...props}>
 32       <App initData={props.data} bbcOrigin={props.bbcOrigin} />
 33     </StaticRouter>
 34   );
 35 }
```



Implementing <Profiler> in React App (Cont.)

```
▶ {id: 'App', phase: 'mount', actualDuration: 272.6000003814697, baseDuration: 261.9000005722046, startTime: 3495.6000003814697, ...}  
▶ {id: 'App', phase: 'update', actualDuration: 3.1999998092651367, baseDuration: 263.3000011444092, startTime: 4042.7000002861023, ...}  
▶ {id: 'App', phase: 'update', actualDuration: 11.40000057220459, baseDuration: 249.99999952316284, startTime: 4048.4000000953674, ...}
```



Implementing <Profiler> in React App (Cont.)

```
return (
  <Profiler id="ArticlePage" onRender={handleOnRender}>
    <div css={styles.pageWrapper}>
      <ATIAalytics atiData={atiAnalytics} />
      <ChartbeatAnalytics
        sectionName={pageData ?. relatedContent ?. section ?. name}
        title={getPromoHeadline(pageData)}
      />
      <ComscoreAnalytics />
      <NielsenAnalytics />
    </div>
  </Profiler>
)
```



Implementing <Profiler> in React App (Cont.)

```
▶ {id: 'ArticlePage', phase: 'mount', actualDuration: 58.30000019073486, baseDuration: 52.50000190734863, startTime: 4048.5999999046326, ...}
▶ {id: 'App', phase: 'mount', actualDuration: 293.59999990463257, baseDuration: 283.80000257492065, startTime: 3815.3999996185303, ...}
▶ {id: 'ArticlePage', phase: 'update', actualDuration: 0.19999980926513672, baseDuration: 51.80000162124634, startTime: 4387, ...}
▶ {id: 'App', phase: 'update', actualDuration: 1.9000000953674316, baseDuration: 283.50000286102295, startTime: 4385.099999904633, ...}
▶ {id: 'ArticlePage', phase: 'update', actualDuration: 8.599999904632568, baseDuration: 34.40000009536743, startTime: 4391.799999713898, ...}
▶ {id: 'App', phase: 'update', actualDuration: 8.700000286102295, baseDuration: 266.10000133514404, startTime: 4391.699999809265, ...}
```



Enabling Production Profiling Support

Adds overhead, use sparingly or in staging environments

Create React App

```
npm run build -- --profile
```

Next.js

```
next build --profile
```





Configure Production Profiling

React 18 Debugging Playbook

Kamran Ayub





```
const analyticsData = [];  
  
function handleOnRender(/* args */) {  
  analyticsData.push({  
    /* analytics object */  
  });  
  
}  
}
```

◀ **Create an event batch**

◀ **Queue messages with timing data**



```
const analyticsData = [];  
  
function handleOnRender(/* args */) {  
  analyticsData.push({  
    /* analytics object */  
  });  
  
}  
  
document.addEventListener(  
  "visibilitychange", function logData() {  
    if (document.visibilityState === "hidden") {  
    }  
  }) ;
```

◀ **Create an event batch**

◀ **Queue messages with timing data**

◀ **When the user navigates away from the page**



```
const analyticsData = [];

function handleOnRender(/* args */) {
  analyticsData.push({
    /* analytics object */
  });
}

document.addEventListener(
  "visibilitychange", function logData() {
  if (document.visibilityState === "hidden") {

    const queued = navigator.sendBeacon(
      "/log", analyticsData);

    if (queued) {
      analyticsData.length = 0;
    }
  }
});
```

◀ **Create an event batch**

◀ **Queue messages with timing data**

◀ **When the user navigates away from the page**

◀ **Queue up event batch**

◀ **Flush the queue (reset the array)**



Recommended Analytics and Tracking Libraries

<https://www.npmjs.com/package/react-tracking>

<https://github.com/treosh/web-vitals-reporter>



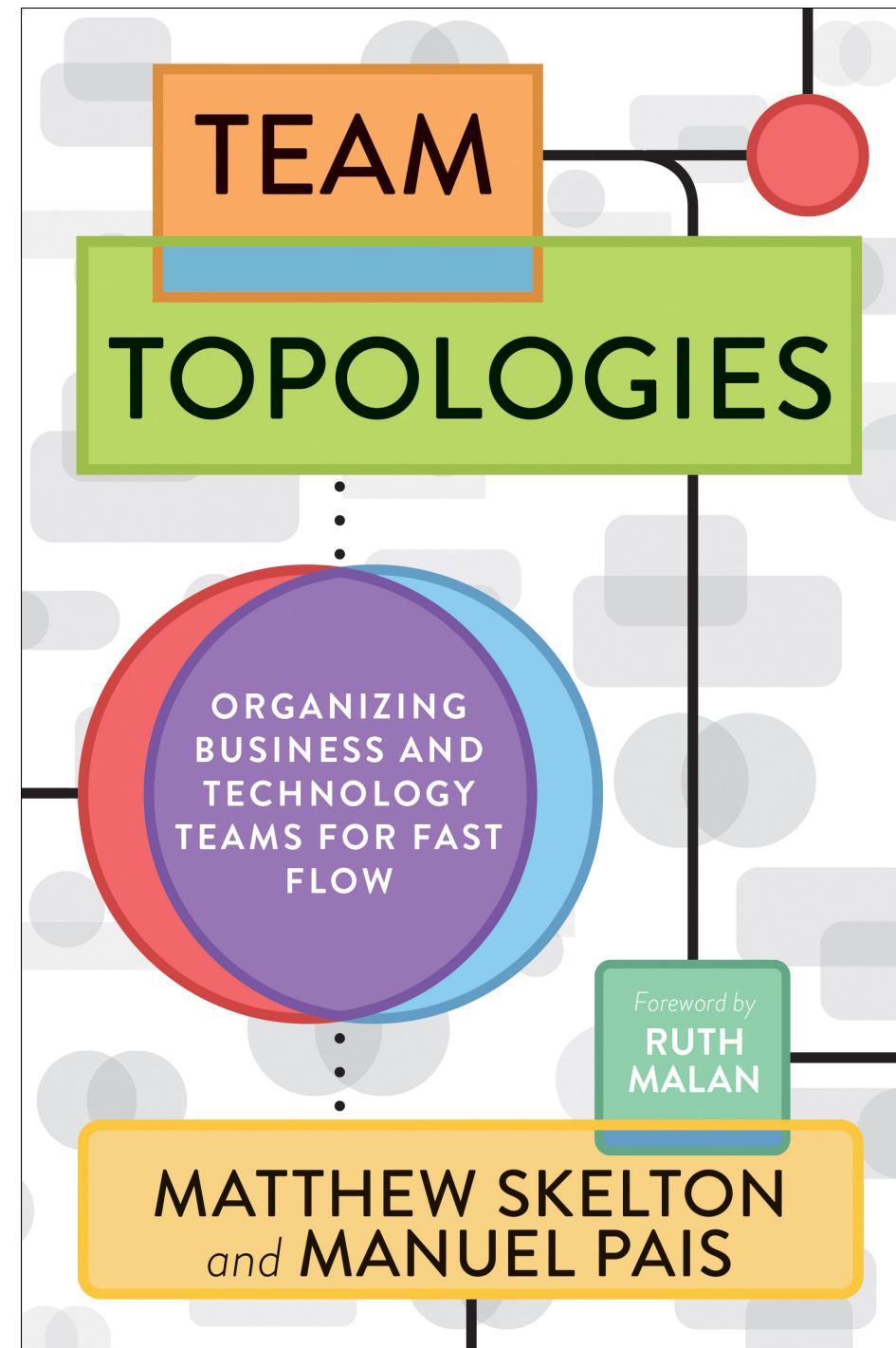


Monitoring Performance Over Time

Live Video Intro

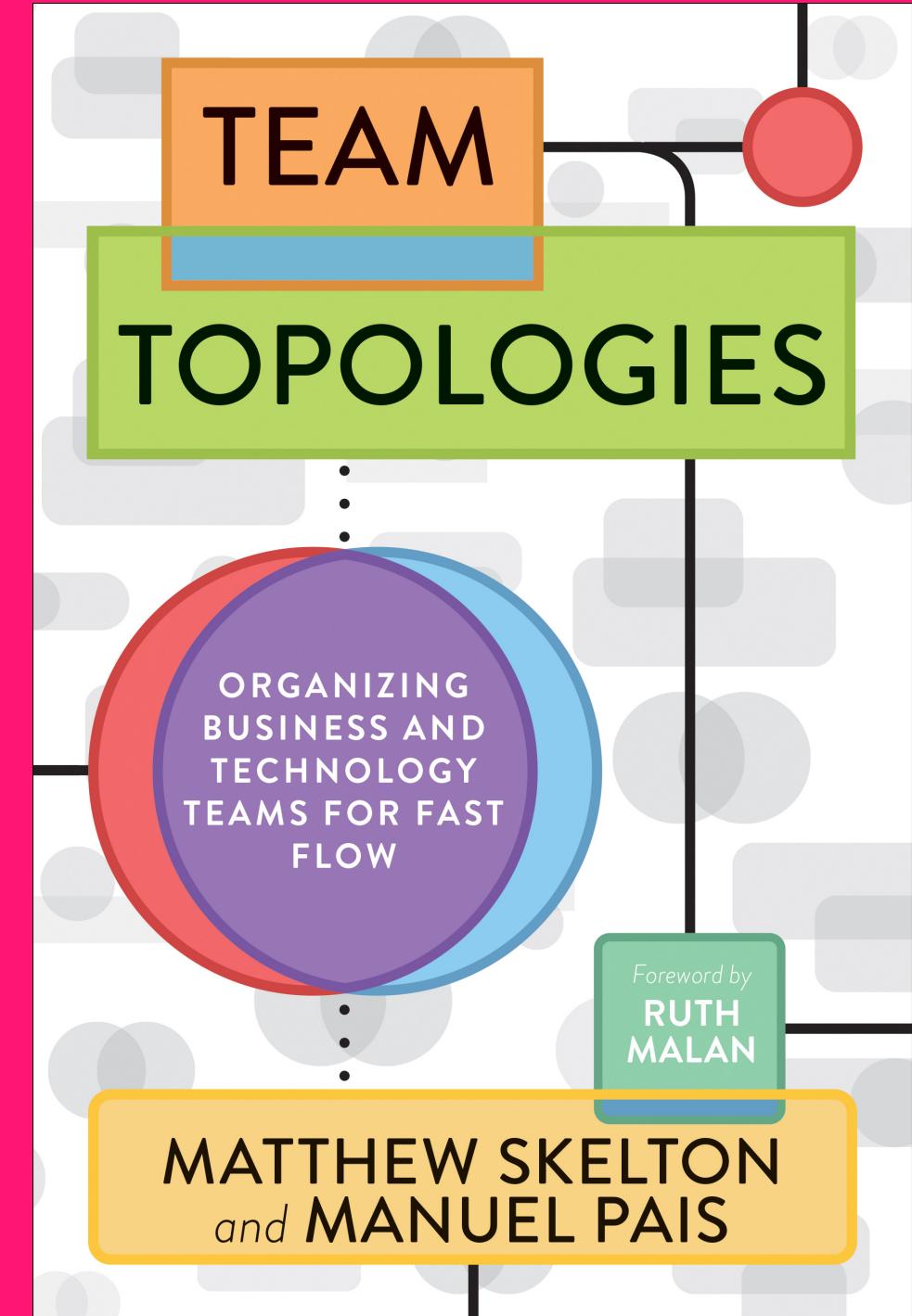


Live Video Intro (Cont.)

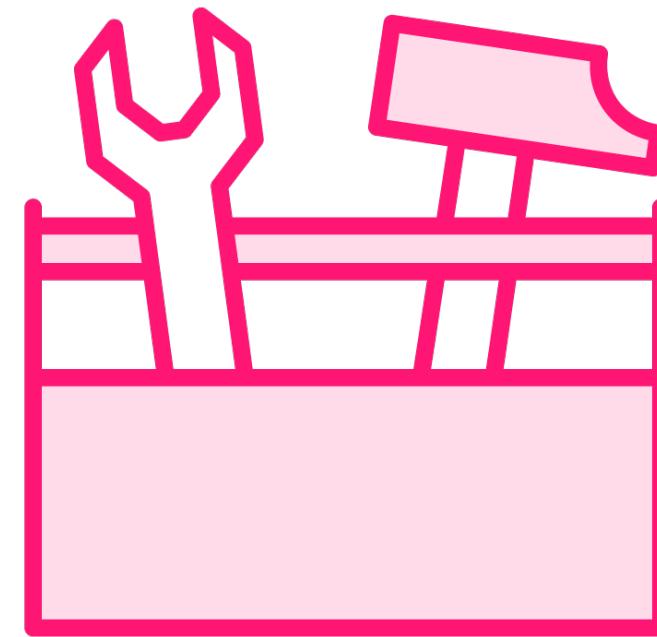


Enabling Team

“[Enabling teams] cross-cut to the stream-aligned teams and have the required bandwidth to research, try out options, and make informed suggestions on adequate tooling, practices, frameworks, and any of the ecosystem choices around the application stack.”



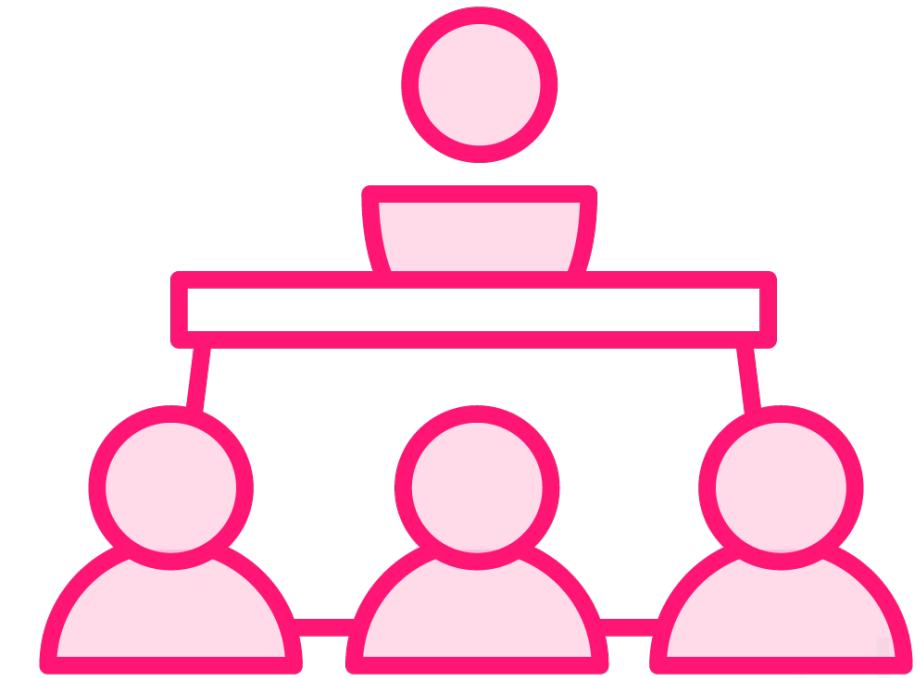
Establishing a Performance Enablement Team



Choose, manage, and integrate tools



Design and conduct performance audits



Teach developers about performance



Live interstitial



Need to convince your boss?

Objective

To establish a specialized Web Performance Enablement Team dedicated to enhancing the efficiency, speed, and overall performance of our web applications. This team will be instrumental in ensuring our web properties deliver optimal user experiences, aligning with our strategic goals of customer satisfaction and technological excellence.

Business Impact of Web Performance Optimization

The establishment of a Web Performance Enablement Team is not just a technical necessity but a strategic business decision. Evidence from various case studies underlines the critical importance of web performance in different sectors.

E-commerce Conversion Rates:

- **Amazon, 2009:** An Amazon engineer revealed that every 1-second increase in latency led to a 1% reduction in conversions.
- This statistic, though from 2009, highlights the direct correlation between web performance and e-commerce success.

Media and Content Views:

- **The Financial Times:** Found that an additional second in load time resulted in 4.6% fewer article views, and this almost doubled to 7.9% fewer views with a 3-second delay.
- **BBC:** Discovered that each additional second in load time led to a significant loss in page views, critical for a site with over a billion page views per day.

User Engagement and Retention:

- A study involving an experiment with The Telegraph showed a 4-second delay resulted in 11% fewer visitors, and a delay of up to 20 seconds reduced page views by 44%.
- These findings demonstrate the impact of performance on user retention and engagement.

Comparative Performance Data:

- **MXS, 2022:** A case study by GatsbyJS involving MXS, which built over 100 auto dealership websites, showed significant performance differences between competitors in the same region. This case study was based on actual live production traffic data, emphasizing the real-world implications of web performance.

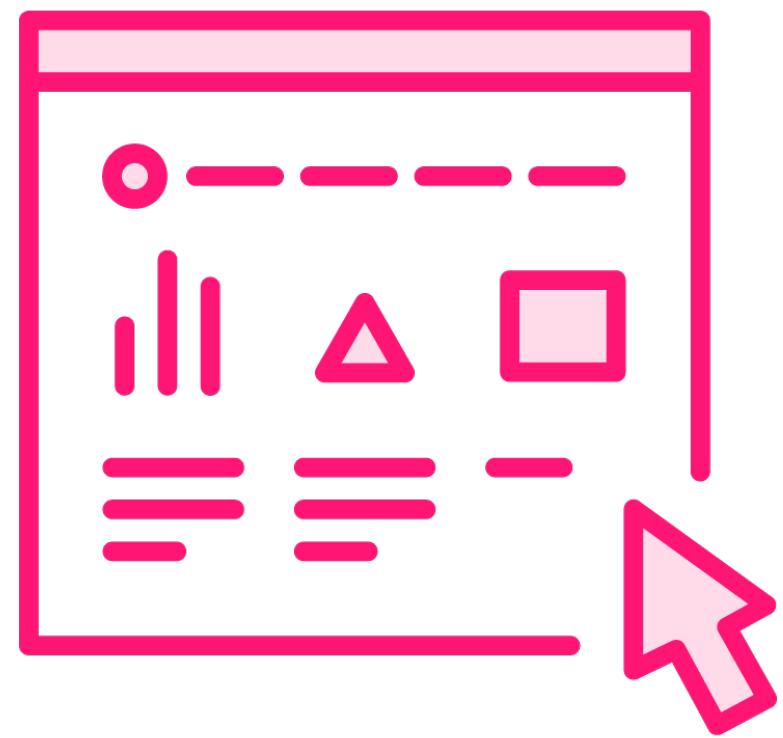
Justification for a Web Performance Enablement Team:

- **Direct Impact on Revenue:** As evidenced by the Amazon case, web performance directly affects sales and revenue, especially in e-commerce.
- **Content Engagement:** Media sites like The Financial Times and BBC have shown that even small delays can significantly reduce content engagement.

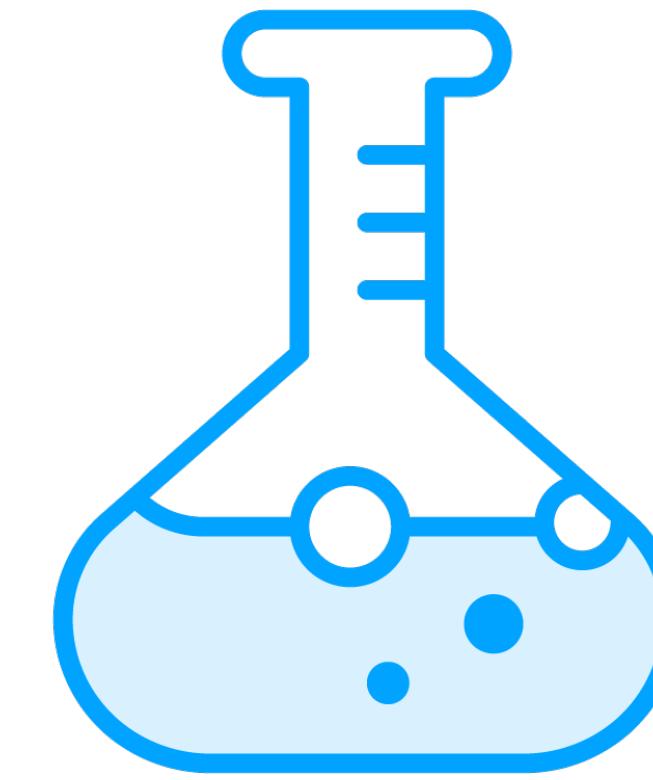
Check out the Sample Executive Briefing for a Web Performance Enablement Team in the Exercise Files of the course



Performance Monitoring Tools



Real-User Monitoring (RUM)

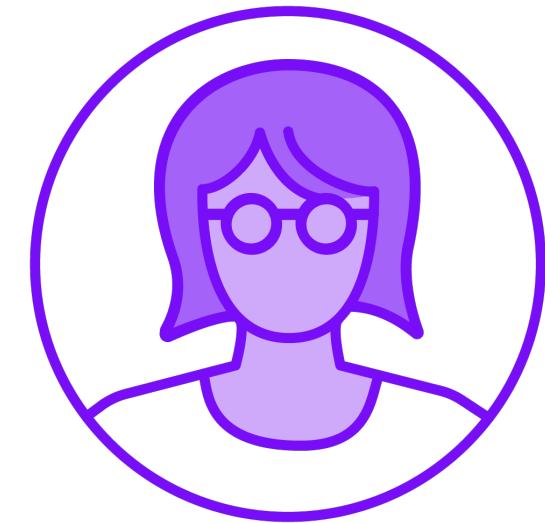


Synthetic Monitoring



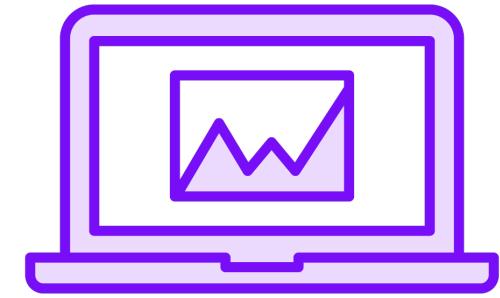
Understanding Real-User Monitoring (RUM)

Capture and analyze user interactions in real-time



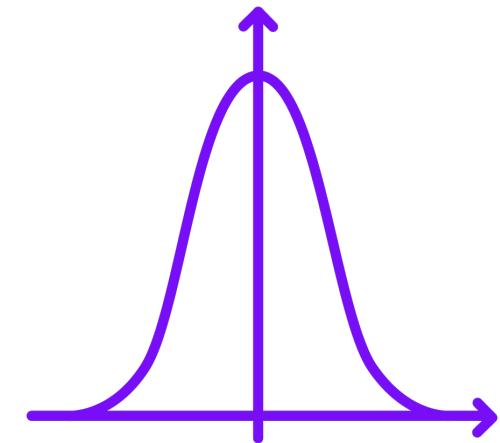
Reflects actual UX

Get insights into how users actually experience your app



Captures metrics

Such as Web Vitals, network conditions, and request metrics



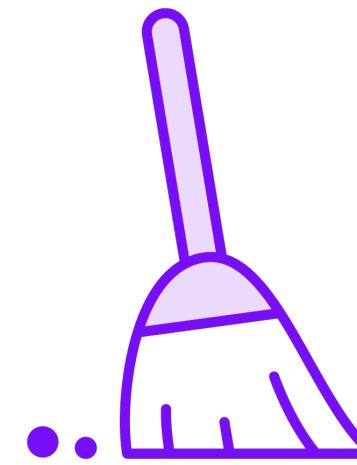
Results will vary

Depending on device, location, and network conditions



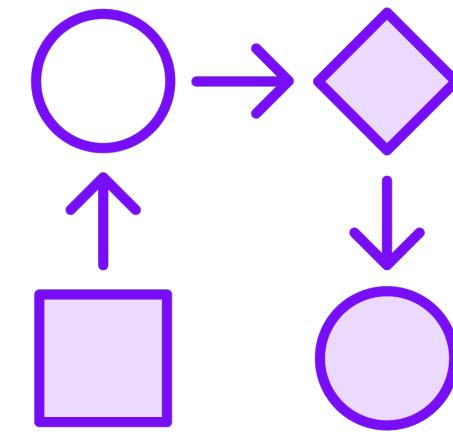
Understanding Synthetic Monitoring

Capture and analyze apps in a simulated lab environment



Controlled environment

Ensures consistency
between benchmarks



Automation-friendly

Usually provides ways
to script flows



Regression analysis

Understand when code
changes introduce
regressions



Combining RUM and Synthetic Monitoring

Real-User Monitoring

Identify issues not caught during development

Prioritize issues that affect real users

Used for dynamic *observability* (the "why")

&

Synthetic Monitoring

Identify issues before they reach end-users

Catch regressions before they are reintroduced

Used for static *monitoring* (the "what")



Anecdote: An Elusive Bug

The screenshot shows a web browser window with the Target website loaded. The page features a navigation bar at the top with links for 'Categories', 'Deals', 'New & Featured', 'Pickup & Delivery', a search bar containing 'What can we help you find?', a microphone icon, a magnifying glass icon, a 'Sign in' link, and a shopping cart icon.

A prominent orange error message box is centered on the page. It contains a yellow warning icon followed by the text 'Please review these errors'. Below this, a smaller message reads 'Something went wrong with the request, please try again.'

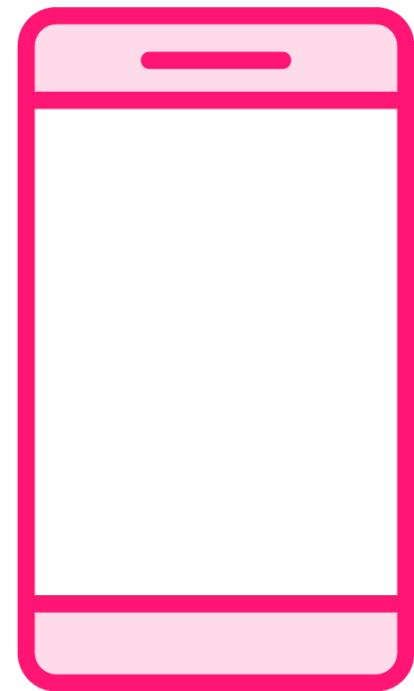


Anecdote: An Elusive Bug

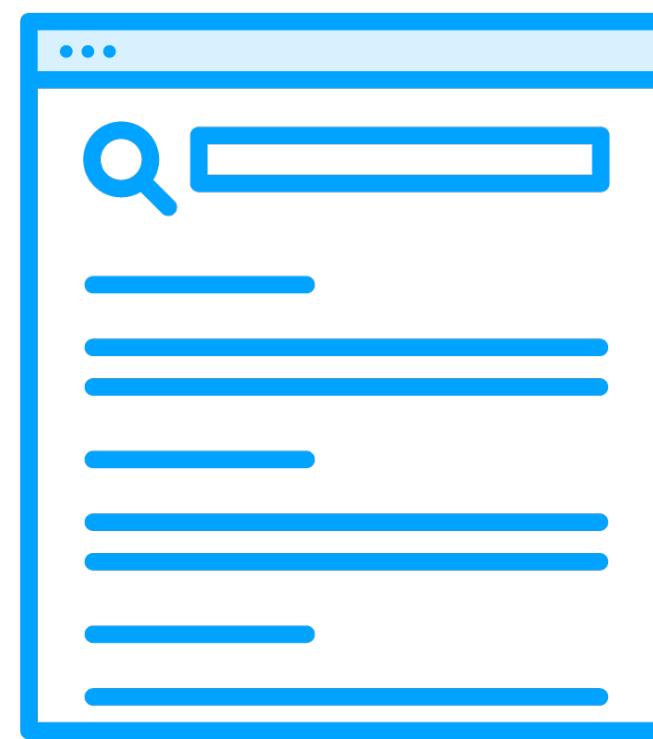
Failed to execute 'removeChild' on 'Node': The node to be removed is not a child of this node.



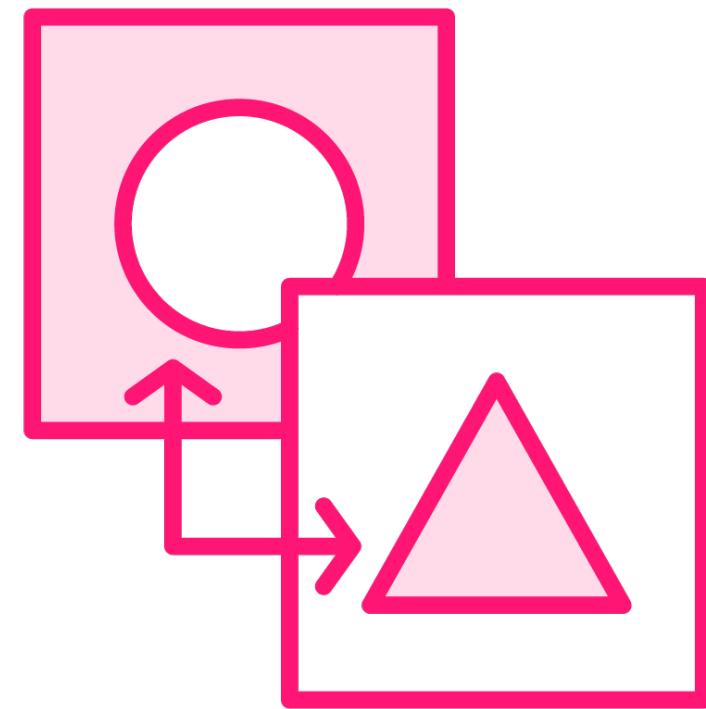
Anecdote: An Elusive Bug



Chrome on Android



**Happening often on
search page**



**"es-US" reported from
navigator.language**



Anecdote: An Elusive Bug

Make React resilient to DOM mutations from Google Translate #11538

 **Closed**

fritz-c opened this issue on Nov 12, 2017 · 68 comments



Live interstitial



Performance Monitoring Tools and Services

Real-Time

Product Analytics

Google Analytics,
Hotjar, Glassbox, etc.



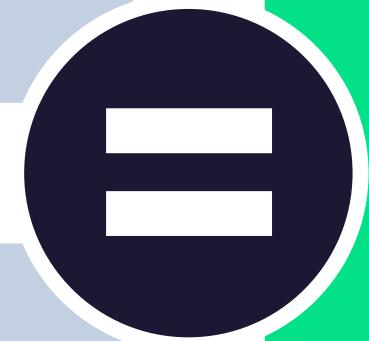
APM Tools

New Relic, LogRocket,
Application Insights, etc.

Synthetic

Browser Automation

Selenium, Cypress,
Playwright, etc.



Monitoring Services

Pingdom, WebPageTest,
BrowserStack, etc.

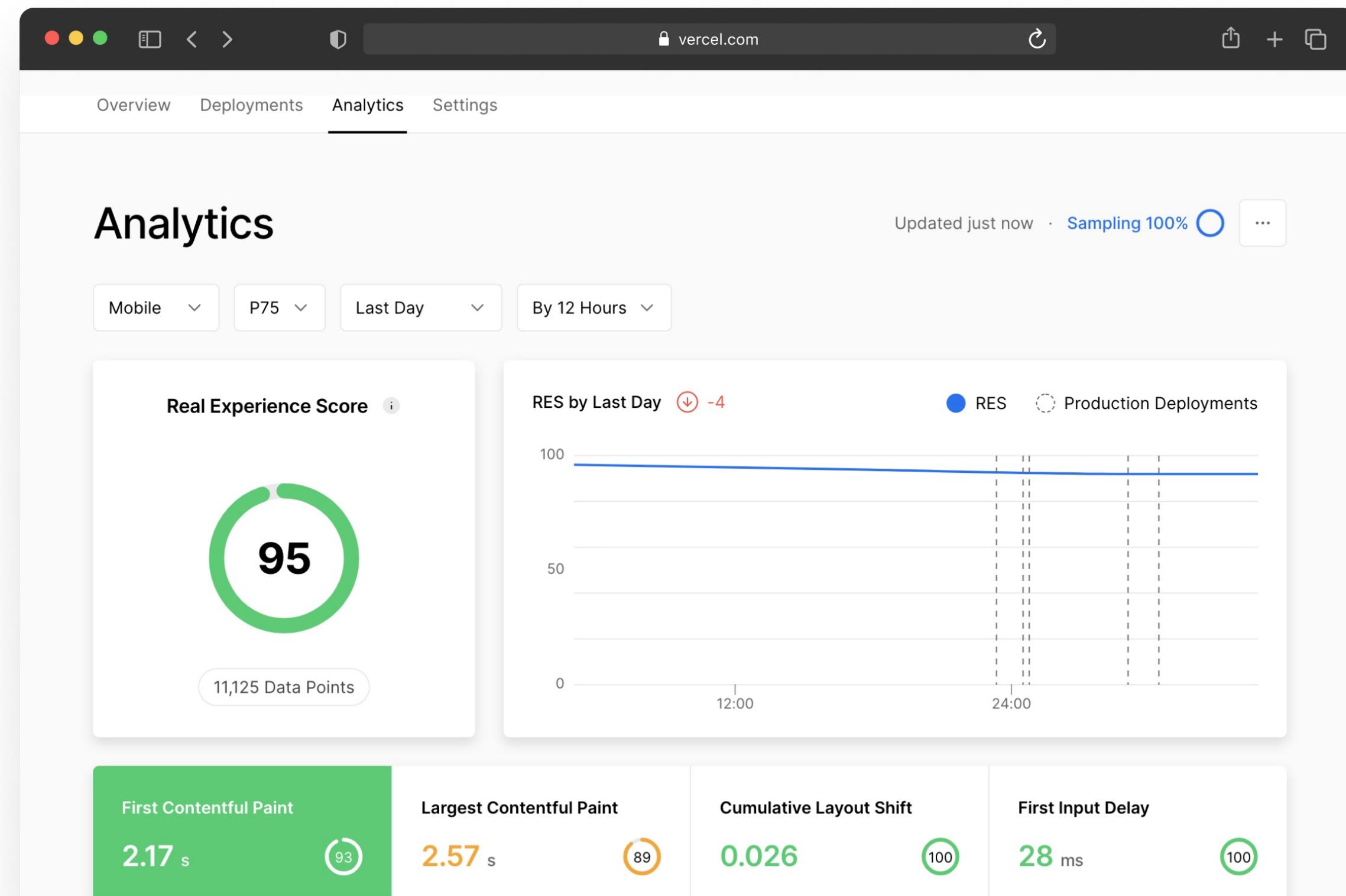
Combined

Digital Experience Monitoring (DEM)

Dynatrace, Splunk,
AppDynamics, Datadog



Next.js Speed Insights (Vercel-only)



Summary

Establish a performance auditing process with a clear strategy and objective

Create ownership and build a performance culture through an enablement team

Use the React Dev Tools along with native browser profiling tools to measure and collect metrics, like Web Vitals

Implement real-user and synthetic monitoring to catch issues or regressions

