

Когда лагает интерфейс

Белов Артём 

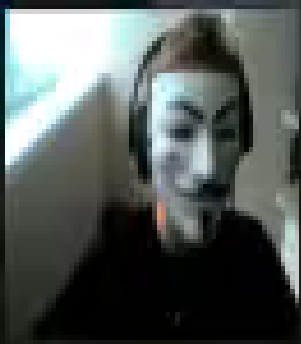


Не понятно когда лагает интерфейс

Что делать когда лагает интерфейс?

Когда лагает интерфейс, как исправить?

Как в DevTools понять когда лагает интерфейс?



Speedrun по «WebPerf»

**Когда лагает
интерфейс**

Зачем?



Кто такой и зачем нужен Артём Белов?

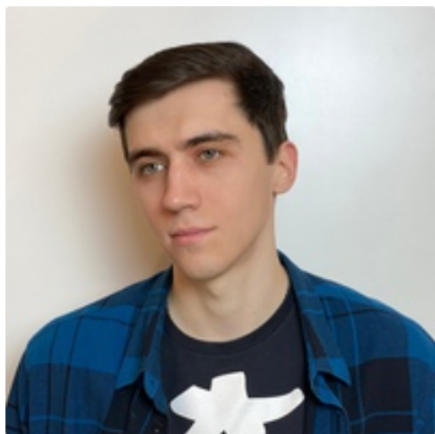


Artem Belov 

1  / 1 







Артём Белов

online

Я еду в электричке.

Родной город:

Отрадный

Место работы:

группа для тех кто еле еле просыпается сутра...выходит на улицу..идет берет у знакомой продавщици 5 бутылок пепси колы...приходит домой..залазит в контакт..ложится спать

DevTools—это сложно

- Изобилие информации
- Непонятна связь сценария UX и DevTools
- Требуется понимание работы браузера



Кто прочитал, тот освоит DevTools

БОРИШ



- Рабочее окружение
 - Знания и инструментарий
- Метрики
 - Хронология, заблуждения, будущее
- Куда смотреть в DevTools...
 - ...когда лагает интерфейс



Дисклеймер

Энвайронмент

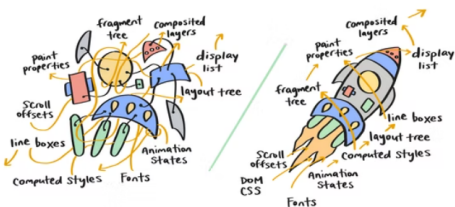


Знания

- Фундаментальные знания CS
- Понимание работы сети
- *Понимание работы браузера**



Понимание работы браузера



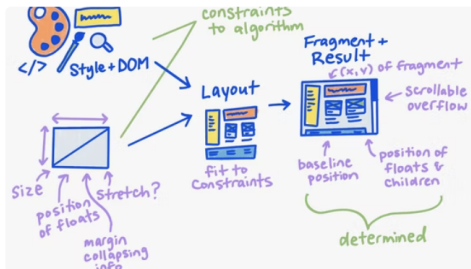
RenderingNG deep-dive: BlinkNG

Rendering



Stefan Zager, Chris Harrelson

April 19, 2022



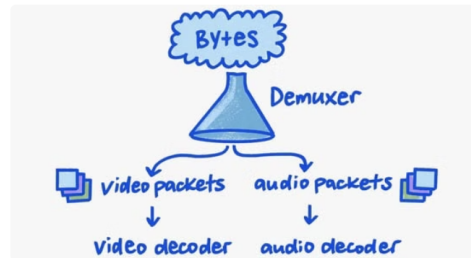
RenderingNG deep-dive: LayoutNG

Rendering



Ian Kilpatrick, Koji Ishi

October 8, 2021



Deep-dive: VideoNG

Rendering



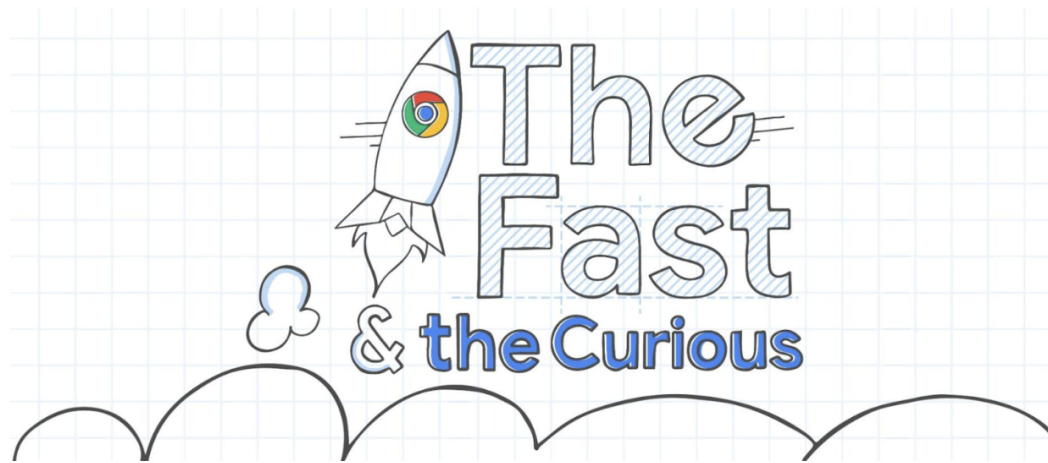
Dale Curtis

September 14, 2021



How Chrome Became the Highest Scoring Browser on Speedometer, Ever

Tuesday, March 15, 2022



Deprecations and Removals






103





Программное окружение

``dev`` -версии браузеров:

-  Chrome Dev Channel (*Canary & Beta*)
-  Firefox Developer Edition
-  Android Stodio, Xcode Simulator



Инструменты

-  Lighthouse
-  PageSpeed Insights
-  WebPageTest

8 days ago
connorjclar
k
v9.6.1
202d605
Compare

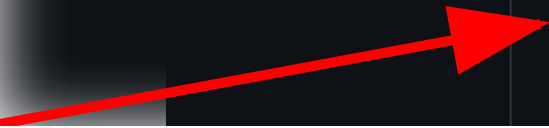
v9.6.1 Latest

Full Changelog

We expect this release to ship in the DevTools of [Chrome 103](#), and to PageSpeed Insights within 2 weeks.

Core

- fps: run at the end of timespan/snapshot (#13989)
- responsiveness: add element screenshot to INP diagnostic (#13984)
- responsiveness: add better INP fallback for old Chrome versions (#13985)



https://web.dev/

Анализировать

Мобильные устройства

Компьютер



Определяйте фактическую производительность сайта

Этот URL Источник



Оценка основных интернет-показателей: проверка пройдена

Развернуть

Largest Contentful Paint (LCP)

1,9 сек.



First Input Delay (FID)

14 мс



Cumulative Layout Shift (CLS)

0



ДРУГИЕ ПОКАЗАТЕЛИ, ЗАСЛУЖИВАЮЩИЕ ВНИМАНИЯ

First Contentful Paint (FCP)

1,4 сек.



Interaction to Next Paint (INP)

292 мс



Time to First Byte (TTFB)

0,8 сек.



Данные за последние 28 дней

Мобильные устройства

Данные нескольких выполнений (отчет об удобстве пользования браузером Chrome)

Периоды от начала до полной загрузки страницы

Сетевые подключения

Данные всех версий Chrome

First Byte **.202s**
 Start Render **.400s**
 FCP **.397s**
 Speed Index **.686s**
 LCP **1.513s**
 CLS **0**
 TBT **≥ .022s**
Total Bytes **984 KB**

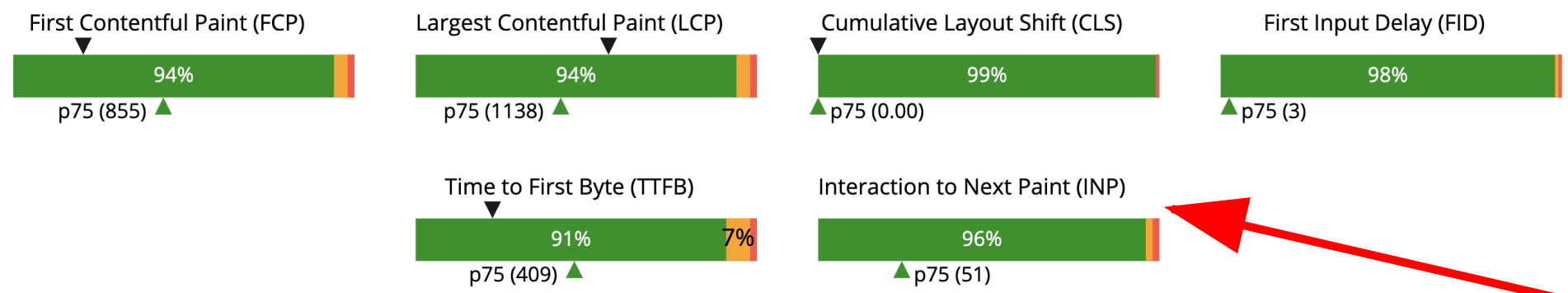
VISUAL PAGE LOADING PROCESS ([EXPLORE](#))



[Compare First Views](#)
 [Plot Full Results](#)

Real User Measurements (Collected anonymously by Chrome browser via Chrome User Experience Report.)

Key: WPT's Metrics for comparison: ▼ First View



PRODUCT NEWS

JUN. 01, 2022

Introducing Opportunities & Experiments: Taking the Guesswork out of Performance

Ещё инструменты

4. Chrome UX Report
5. Google Analytics 4
- 6.

Энвйронмент

Метрики





Web Vitals

почему



Response

Animation

Idle






Load



- Сосредоточьтесь на пользователе
- Отвечайте на ввод пользователя менее чем за 100 мс
- Выдавайте кадр менее чем за 10 мс при анимации или прокрутке
- Увеличьте время простоя основного потока браузера
- Загружайте интерактивный контент менее чем за 5 секунд



«Web Vitals» в 2022

- 🚧 Core Web Vitals (CWV)
 - FID
 - LCP
 - CLS  
- ⚠️ Other Web Vitals (OWV)
 - FCP
 - TTFB 
 - INP  



Следит за страницей



Перерабатывалась



В эксперименте

```
yarn add web-vitals@1
```

! OVV: Time to First Byte

время до первого байта от сервера

Request/Response

DURATION

Request sent



0.27 ms

Waiting (TTFB)



26.05 ms

Content Download



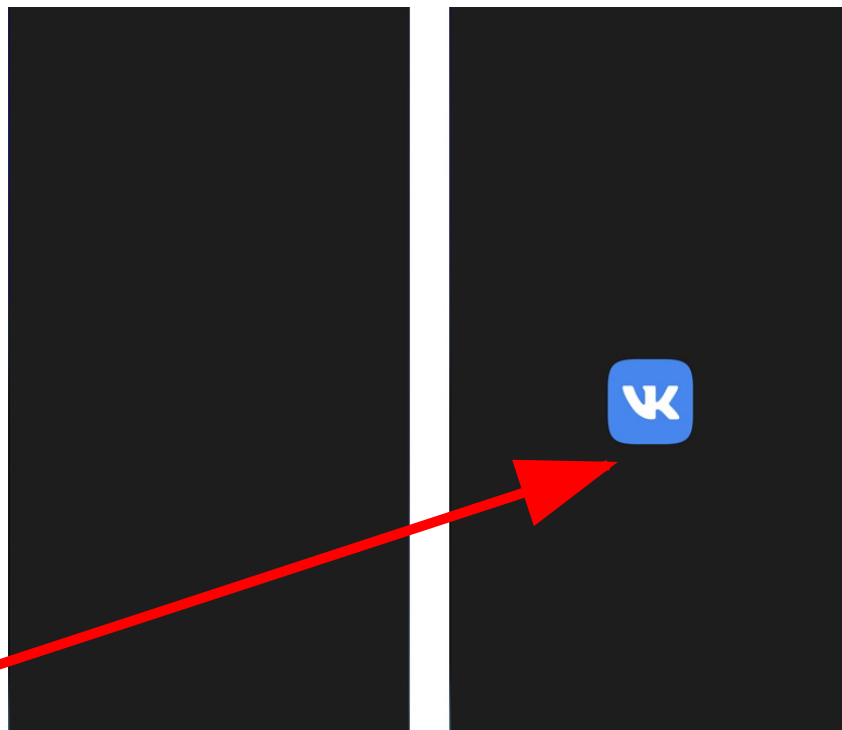
1.41 ms

[Explanation](#)

30.95 ms

! OVV: First Contentful Paint

время первой отрисовки контента



FCP

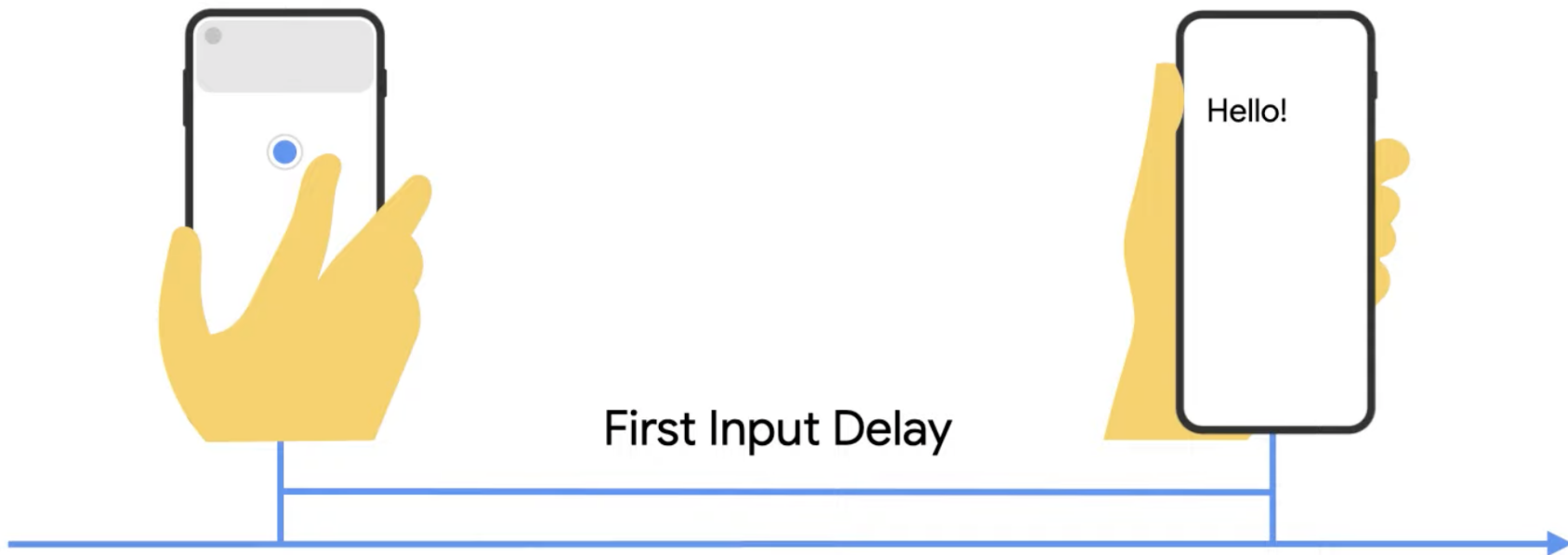


OWV: First Contentful Paint

время первой отрисовки контента

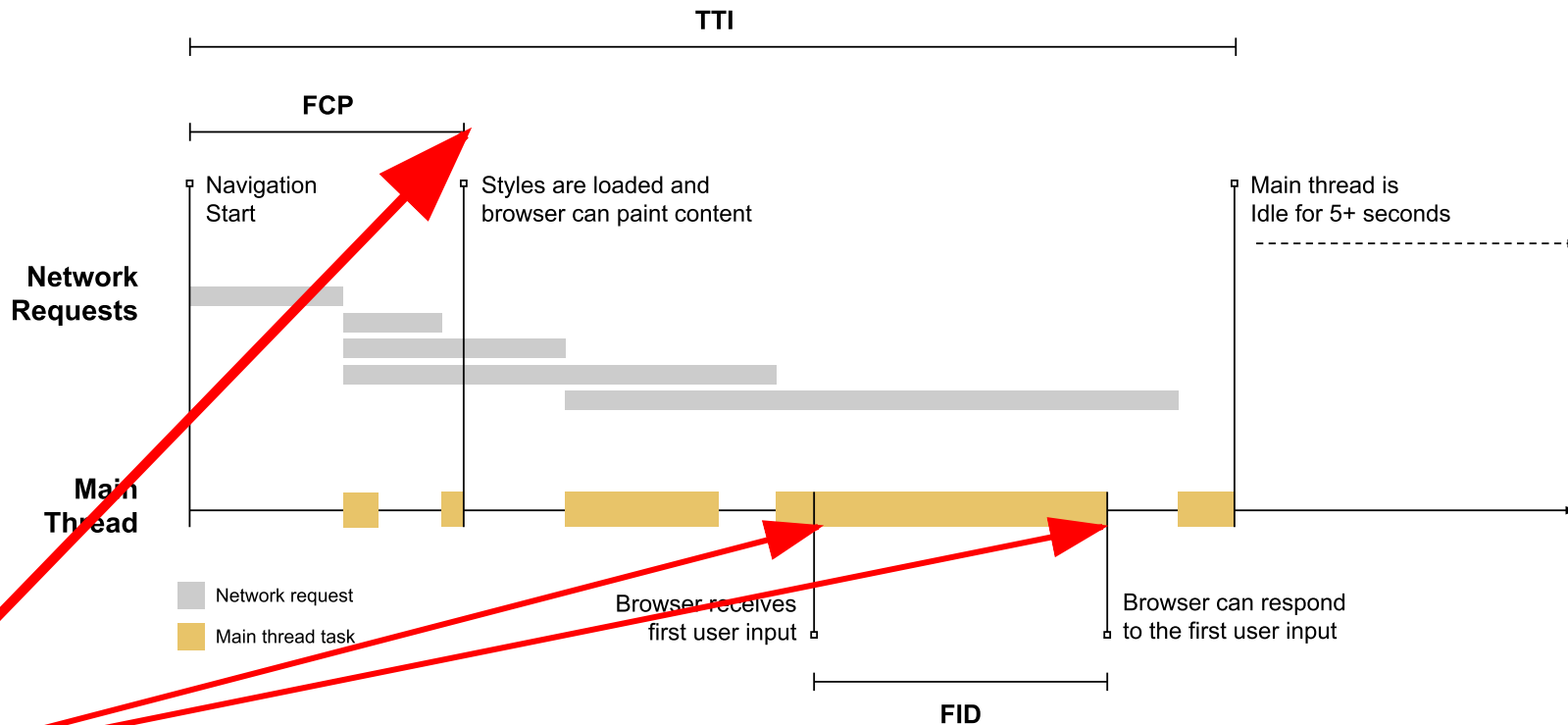
CWV: First Input Delay

интерактивность страницы



CWV: First Input Delay

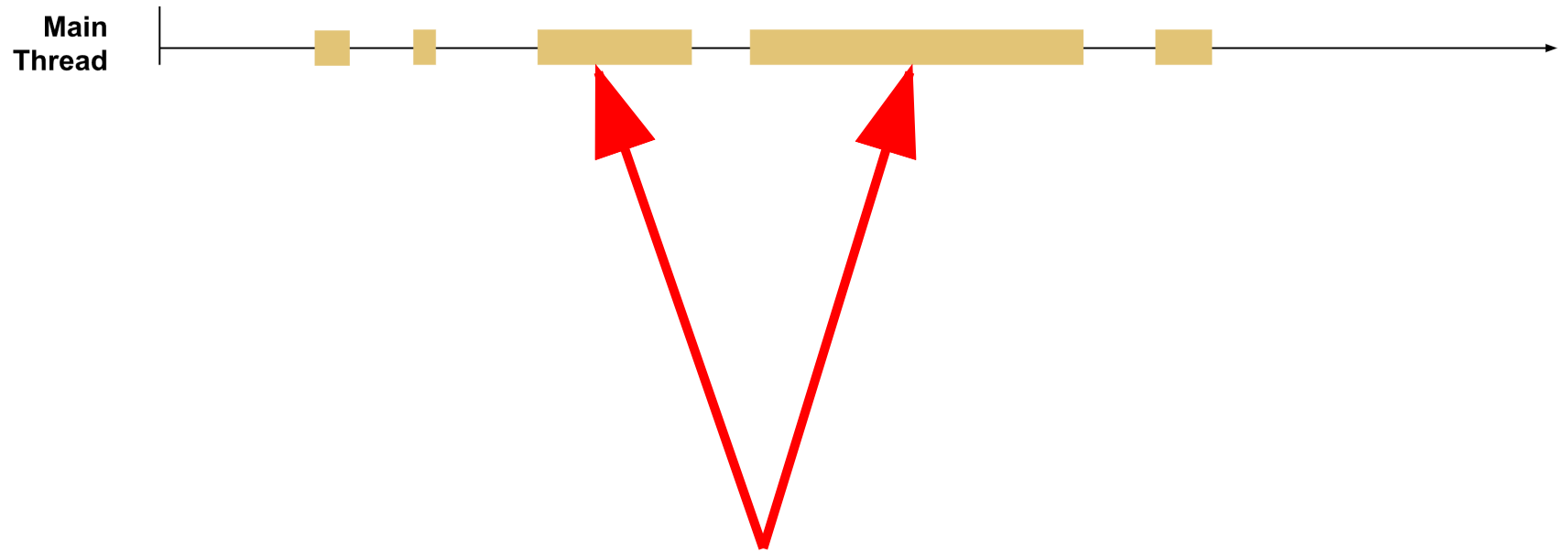
ИНТЕРАКТИВНОСТЬ СТРАНИЦЫ





ОПТИМИЗИРОВАЛ FID

И УВЕЛИЧИЛ ПОКАЗАТЕЛИ



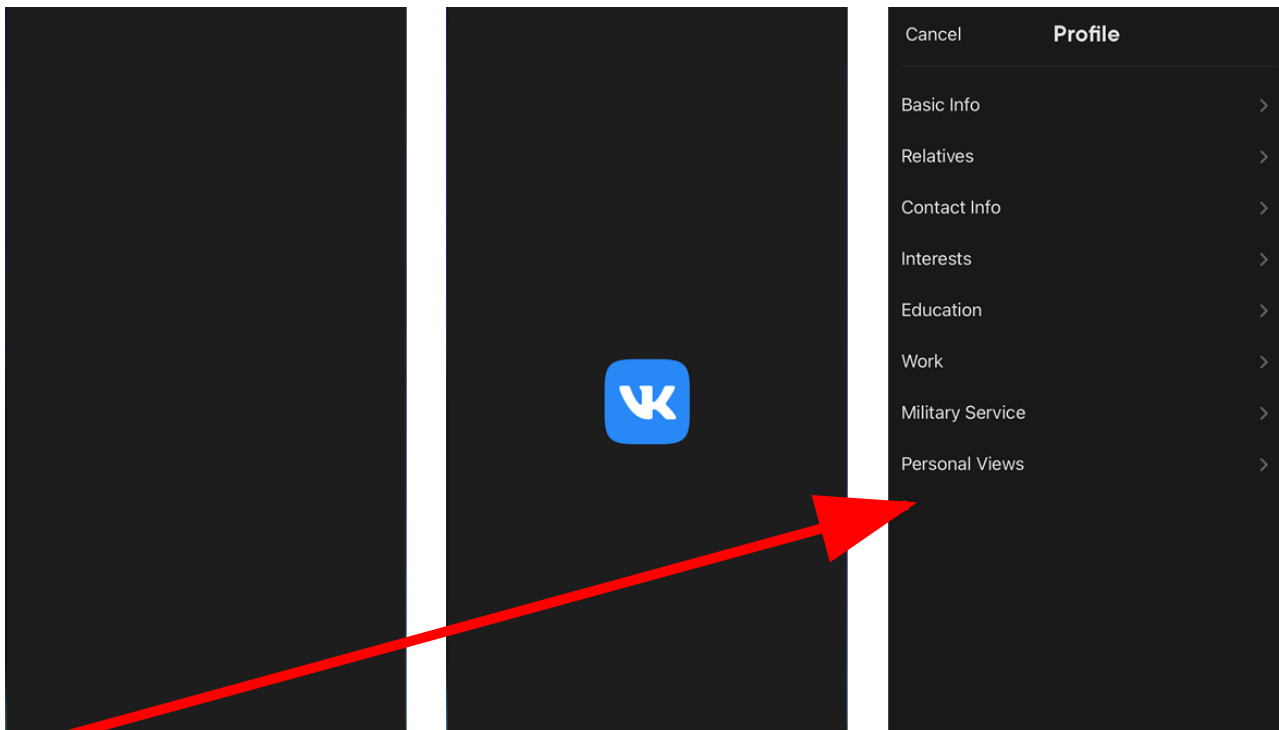
1 новое сообщение

Не могу найти причины возросшего FID.
Поможешь?



CWV: Largest Contentful Paint

воспринимаемая скорость загрузки



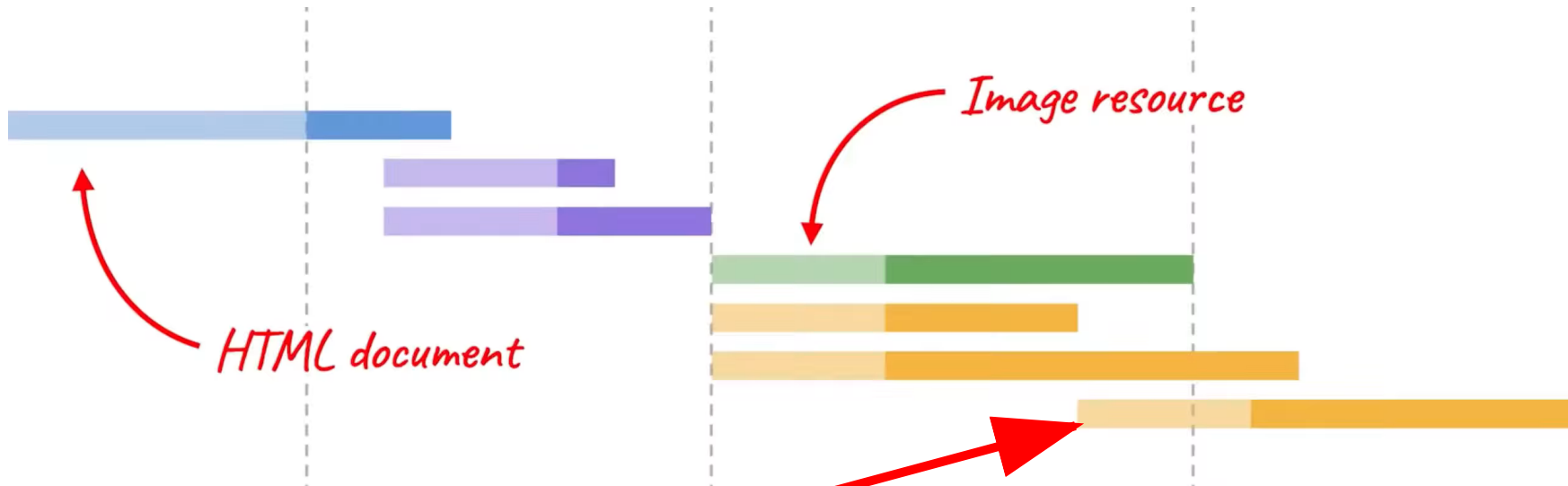
FCP

LCP

ОПТИМИЗИРОВАЛ LCP



И УВЕЛИЧИЛ ПОКАЗАТЕЛИ





CWV: Cumulative Layout Shift

визуальная стабильность





CWV: Cumulative Layout Shift

визуальная стабильность

**Success story: как мы
значительно улучшили
CWV метрику
более чем на 50%**



Ого!

```
yarn add web-vitals@2
```

[BREAKING] Update CLS to max session window 5s cap 1s gap



Cumulative Layout Shift: Session Window

визуальная стабильность








My Challenge to the Web Performance Community

October 6, 2021

I've noticed a trend in recent years—a trend that I've admittedly been part of myself—where performance-minded developers will rebuild a site and then post a screenshot of their Lighthouse score on social media to show off how fast it is.

 <https://mysuperfastsite.com/>



Performance



Accessibility



Best Practices



SEO



Progressive
Web App

▲ 0–49

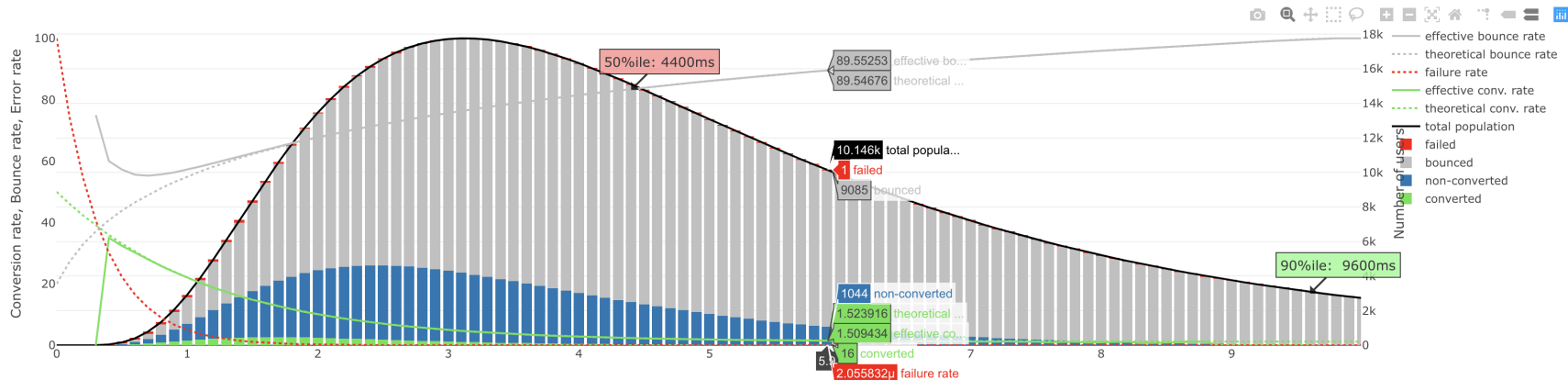
■ 50–89

● 90–100

Мои вызовы:

- Если вы говорите о производительности в продакшене, то используйте данные RUM.
- Если вы используете одно число в отчёте о производительности, то укажите перцентиль.
- Говоря о показателях RUM, уточняйте период времени.
- Если вы хвастаетесь LAB-результатами, то делайте это в контексте хронологии.

 philipwalton.com/articles/my-challenge-to-the-web-performance-community



Average Conversion Rate

1.05% (6.4%)

Converted Users

10541

Total Value

\$105,410

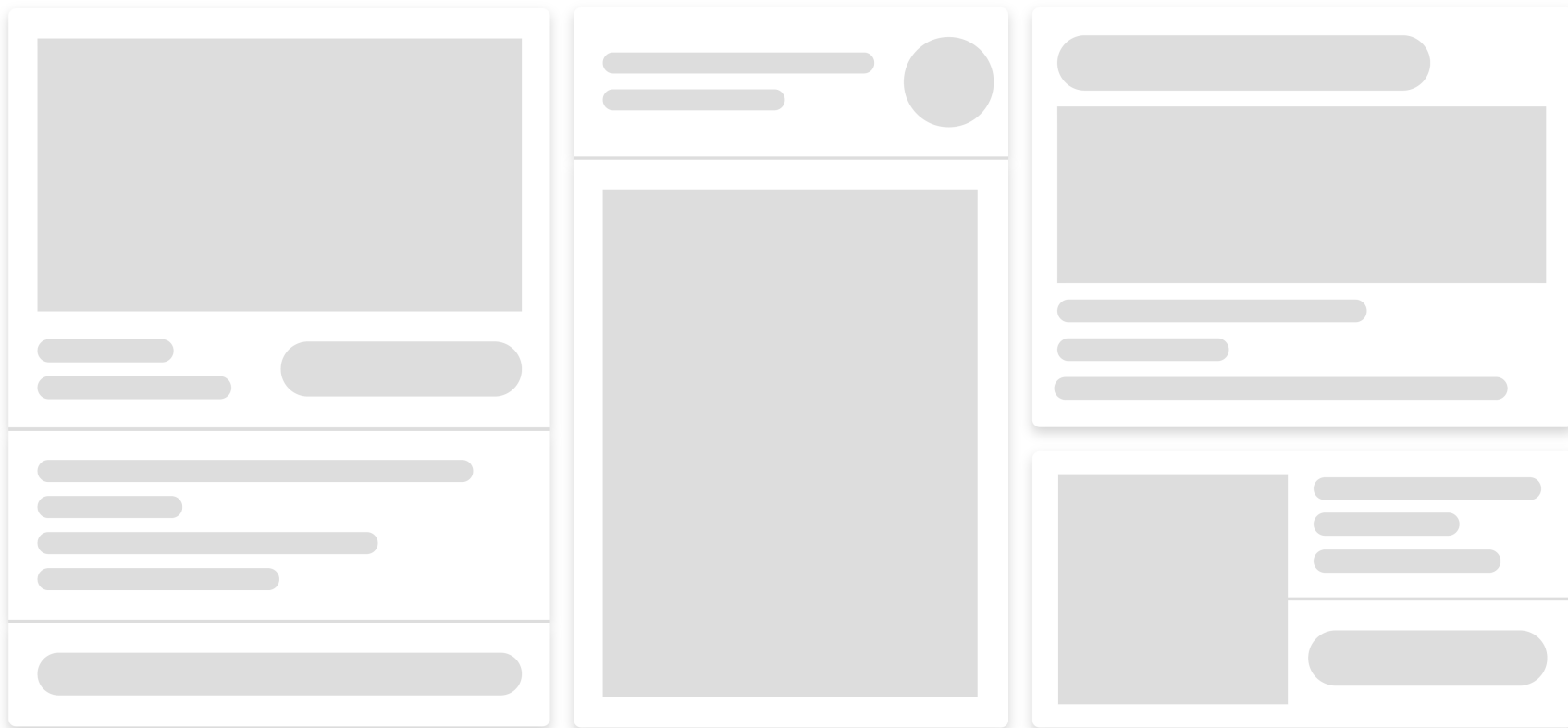
Speed Distribution	Error Rate	Bounce Rate	Conversion Rate	Chart Parameters
Base Speed (μ): <input type="text" value="1.5"/>	Max error rate: <input type="text" value="100"/> %	Min bounce rate: <input type="text" value="20"/> %	Conversion Decay: <input type="text" value="0.85"/>	Number of Users: <input type="text" value="1000000"/>
Variability (σ): <input type="text" value="0.6"/>	Error Rate Decay: <input type="text" value="3"/>	Bounce time compression: <input type="text" value="4"/>	Max Conversion: <input type="text" value="50"/> %	Average Value of a Converted User: <input type="text" value="10"/>
		Bounce rate scale: <input type="text" value="50"/> %	Conversion Poverty Line: <input type="text" value="1.2"/> %	Display Max: <input type="text" value="9.99"/> seconds
				bucket size on the histogram: <input type="text" value="0.1"/>



Οφφφ-τοπ

⚠️ OWV: First Contentful Paint: Skeletons

время первой отрисовки контента



```
yarn add web-vitals@next
```

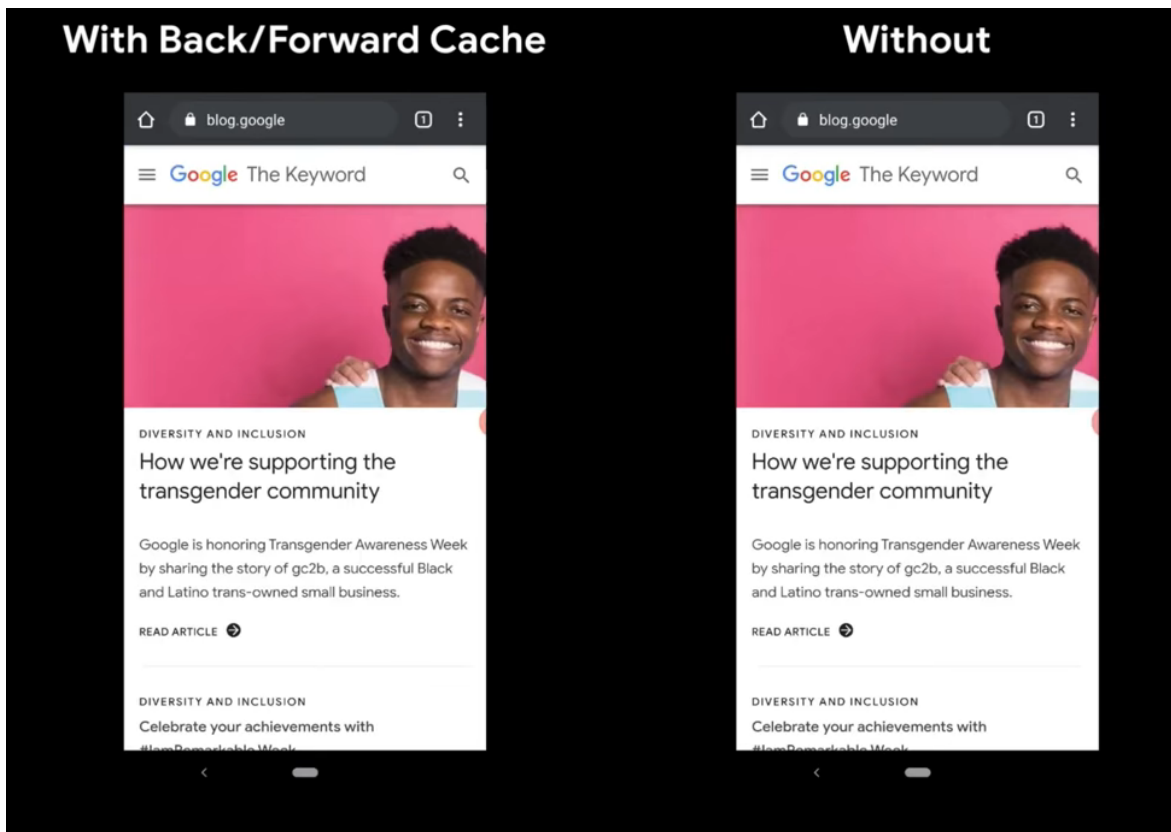
[BREAKING] Report TTFB after a bfcache restore

[FEATURE] Add support for the new INP metric



TTFB: BFCache Restore

время до первого байта от сервера



DevTools → Application

Back/forward Cache

The screenshot shows the Chrome DevTools Application panel with the 'Back/forward cache' section selected in the left sidebar. The main content area displays the following information:

- Back/forward cache**
 - Not served from back/forward cache: to trigger back/forward cache, use Chrome's back/forward buttons, or use the test button below to automatically navigate away and back.
 - URL: `https://vk.com/team`
 - [Test back/forward cache](#)
- Actionable**
 - The page has an unload handler in the main frame.
`UnloadHandlerExistsInMainFrame`
- Pending Support**
 - Pages containing plugins are not currently eligible for back/forward cache.
`ContainsPlugins`
- Not Actionable**
 - Pages whose main resource has `cache-control:no-store` cannot enter back/forward cache.
`MainResourceHasCacheControlNoStore`

[Learn more: back/forward cache eligibility](#)



DevTools → Application

Back/forward Cache

The screenshot shows the Chrome DevTools Application panel with the 'Back/forward cache' section selected in the left sidebar. The main content area displays the following information:

- Back/forward cache**
 - Not served from back/forward cache: to trigger back/forward cache, use Chrome's back/forward buttons, or use the test button below to automatically navigate away and back.
 - URL: `https://vk.com/team`
 - Frames: **8 issues found in 2 frames.** (indicated by a red arrow)
 - Test back/forward cache** button
- Actionable**
 - The page has an unload handler in the main frame. [Learn more: Never use unload handler](#)
 - UnloadHandlerExistsInMainFrame
 - 1 frame
- Pending Support**
 - Pages containing plugins are not currently eligible for back/forward cache.
 - ContainsPlugins
 - 1 frame
 - Back/forward cache is disabled due to extensions using messaging API.
 - EmbedderExtensionMessaging
 - 2 frames
- Not Actionable**
 - Back/forward cache is disabled by flags. Visit `chrome://flags/#back-forward-cache` to enable it locally on this device.



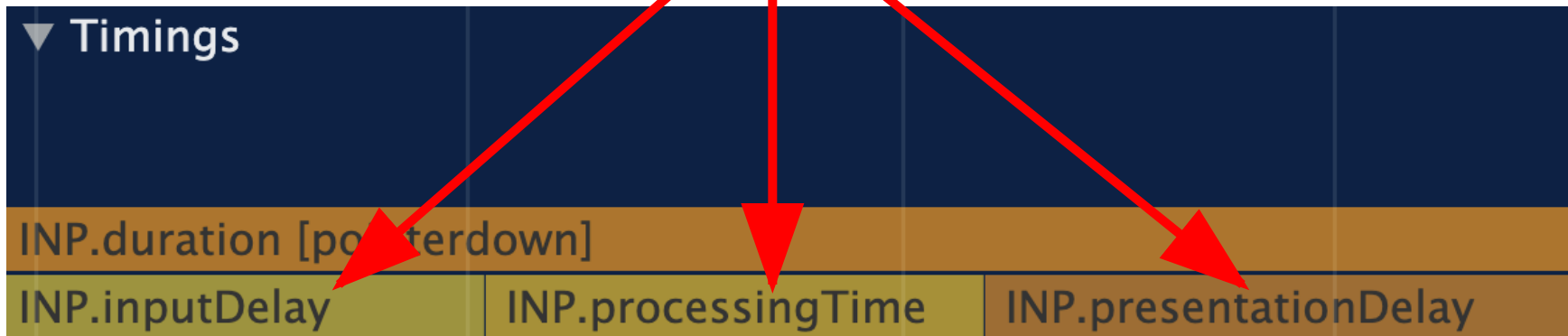


Responsiveness



Another one?

- 90% пользовательской сессии происходит после события ``load``
- 70% пользователей испытывают заметные подвисания UI хотя бы раз в неделю
- Можно рассчитывать на $\times 2$ просмотров, если у UI хорошая отзывчивость
- FID уже на 95% в «зелёной» зоне среди анализируемых сайтов



1. **Задержка ввода** – время между моментом взаимодействия со страницей и началом работы обработчиков событий
2. **Время обработки** – количество времени необходимое для выполнения кода в связанных обработчиках событий
3. **Задержка представления** – время между завершением выполнения обработчиков событий и отображением следующего кадра



OWV: Interaction to next Paint

ОТЗЫВЧИВОСТЬ СТРАНИЦЫ



OWV: Interaction to next Paint

ОТЗЫВЧИВОСТЬ СТРАНИЦЫ

Instructions

1. Use the controls below to add interaction latency.
2. Start interacting with the page (type, click, etc.)
3. Notice how the experience affects the INP value.
4. Refresh the page to start over.

Introduce periodic main thread blocking

Frequency



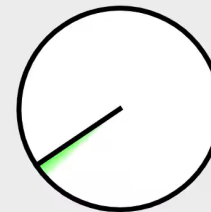
Customize event processing times

keydown blocking time (50ms)



INP value
(since page load)

416



Active









Start typing...

INP

Interaction to Next Paint





-  GoogleChrome/web-vitals #230
INP value not collected when source code is minified +3 subscribed  23 hours ago
-  GoogleChrome/web-vitals #231
Fix 228: Change on INP to also use FID duration when under durationTheshoId +1 subscribed  2 days ago
-  GoogleChrome/web-vitals #229
INP values incorrect when split over blur/focus +2 subscribed  9 days ago
-  GoogleChrome/web-vitals #228
Consider expose FID duration as first INP score, for cases when its under durationThreshold +3 subscribed  10 days ago

```

110 130      const po = observe('event', handleEntries, {
111      -      // The use of 50 here as a balance between not wanting the callback to
112      -      // run for too many already fast events (one frame or less), but also
113      -      // get enough fidelity below the recommended "good" threshold of 200.
114      -      durationThreshold: 50,
131      +      // Event Timing entries have their durations rounded to the nearest 8ms,
132      +      // so a duration of 40ms would be any event that spans 2.5 or more frames
133      +      // at 60Hz. This threshold is chosen to strike a balance between usefulness
134      +      // and performance. Running this callback for any interaction that spans
135      +      // just one or two frames is likely not worth the insight that could be
136      +      // gained.
137      +      durationThreshold: 40,
115 138      } as PerformanceObserverInit);

```




bugs

chromium ▾

New issue

Open issues ▾

🔍 Search chromium issues... ▾



Sign in

☆ Starred by 1 user

Owner: ----

CC: rviscomi@google.com

Status: Available (*Open*)

Components: [Speed](#)>[Metrics](#)

Modified: 6 days ago

Issue 1321807: Benchmark Event Timing Performance Observer at different durationThresholds

🔗 Code

Reported by mmo...@chromium.org on Mon, May 2, 2022, 10:22 PM GMT+4 (16 days ago) **Project Member**

It is not clear what the tradeoffs are for using the default (104ms) durationThreshold, vs something like 50 or the minimum of 16.

It would be nice to understand what the performance costs are, so we can make suggestions for web perf community (and, change defaults in web-vitals.js)



A meme featuring a man in a light-colored polo shirt with 'Staples' written on it, holding a lit cigarette. The background is dark with some metallic elements. The text is overlaid in white, bold, sans-serif font.

МЫ ВСТРОИЛИ WEB VITALS

ПРЯМО В БРАУЗЕР



DevTools → Show Core Web Vitals overlay


The image shows a dark-themed overlay from Chrome DevTools displaying Core Web Vitals. At the top, there is a navigation bar with icons for share, star, puzzle, flask, square, profile, and menu. The star icon is highlighted with a blue circle. Below the navigation bar, three metrics are listed, each with a green dot icon:

- Largest Contentful Paint 0.84 s
- First Input Delay 2.40 ms
- Cumulative Layout Shift 0.03

```
`import * as webVitals from 'web-vitals';`
```



ПРОЧИТАЛ README



СОБРАЛ ПЕТАБАЙТ

НЕТОЧНЫХ МЕТРИК



ПРОЧИТАЛ README

ЕЩЁ РАЗ

Метрики





Best Practices

1. Code split

2. Lazy load

...

501. `content-visibility``

502. `will-change``

...

997. `rel="preload"`

998. `async`

999. `defer`

Best Base Practices

1. Code split

2. Lazy load

...

501. `content-visibility``

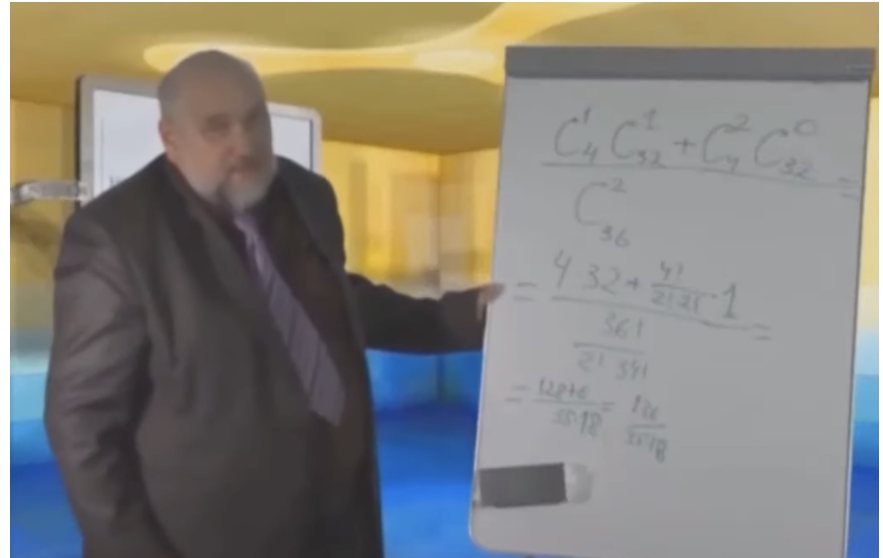
502. `will-change``

...

997. `rel="preload"`

998. `async`

999. `defer`



DevTools



ТОП-3 причины НЕ открывать DevTools

- 1.
- 2.
- 3.



🏆 ТОП-3 причины НЕ открывать DevTools

1. Аха!
2. Попался!
3.)))





The screenshot shows the Chrome DevTools interface with the Network tab selected. A context menu is open over the 'Run' button, displaying a list of commands. The 'Show Application' command is currently selected and highlighted in blue. The background interface includes the top navigation bar with 'Elements', 'Console', 'Recorder', and 'Network' tabs, and a sidebar on the left with a 'Filter' section.

Command	Action
Show Application	Panel
Show CSS Overview	Panel
Show Console	Panel
Show Elements	Panel
Show JavaScript Profiler	Panel
Show Layers	Panel
Show Lighthouse	Panel

NO ADS

AD



Tom & Ben



Sustainable Loading

#TTFB, #FCP, #LCP, #CLS

DevTools → Network:

- Network Conditions
- Coverage



Demo App #1

Load CSS ~0.5KB

Load CSS ~27KB

DevTools → Network

The screenshot shows the Chrome DevTools Network tab. The top navigation bar includes 'Elements', 'Console', 'Recorder', and 'Network'. The 'Network' tab is active, showing a list of requests and a waterfall chart. The list includes requests for 'localhost', 'almond.css', 'bundle.js', 'node_modules_web-vitals_base_j...', 'ws', 'favicon.ico', and three 'chunk.*.css' files. The waterfall chart visualizes the timing of these requests. At the bottom, a summary bar shows: 10 requests, 148 kB transferred, 843 kB resources, Finish: 15.23 s, DOMContentLoaded: 2.68 s, and Load: 3.26 s.

Name	Protocol	Size	Time	Priority	Waterfall
localhost	http/1.1	1.5 kB	567 ms	Highest	
almond.css	http/1.1	5.1 kB	618 ms	Highest	
bundle.js	http/1.1	81.9 kB	1.05 s	Low	
node_modules_web-vitals_base_j...	http/1.1	3.3 kB	583 ms	Low	
ws	websocket	0 B	Pending		
favicon.ico	http/1.1	1.5 kB	567 ms	High	
chunk.1kb.css	http/1.1	510 B	570 ms	Highest	
chunk.27kb.css	http/1.1	26.9 kB	709 ms	Highest	
chunk.1kb.css	http/1.1	510 B	565 ms	Highest	

10 requests | 148 kB transferred | 843 kB resources | Finish: 15.23 s | DOMContentLoaded: 2.68 s | Load: 3.26 s

Load CSS ~0.5KB

Load CSS ~27KB

The screenshot shows the Network tab of a browser's developer tools. The interface includes a top navigation bar with tabs for Elements, Console, Recorder, and Network. The Network tab is active, displaying a list of requests. The filter is set to 'CSS', and a single request is visible. The request details are partially visible, showing a status of 200 and a size of 27KB. The interface also includes various control elements like 'Preserve log', 'Disable cache', and 'No throttling'.

Elements Console Recorder **Network** >> 1

Preserve log Disable cache No throttling

Filter Invert Hide data URLs

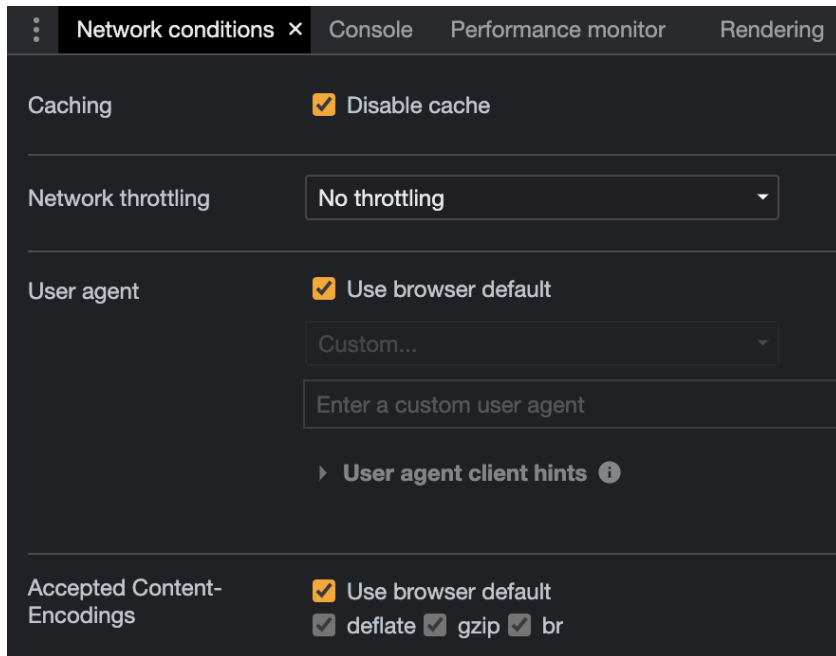
All **CSS** Fetch/XHR JS Media Font Doc WS Wasm Manifest Other Has blocked cookies

Blocked Requests 3rd-party requests

Request	Size	Status
css	27KB	200

DevTools → Network

Run > Show Network conditions



🧠 Приходится держать в уме:

- HTTP/2 приоритезация
- TCP slow-start
- DNS resolutions
- TCP (время соединения)
- PLR
- TLS handshake
- Редиректы

DevTools → Network

Run > Show Coverage

URL	Type	Total Bytes	Unused Bytes	Usage ... ▾
http://localhost:3... /chunk.27kb.css	CSS	205 439	202 762 98.7%	
http://localhost:3000/st... /bundle.js	JS (per function)	384 004	96 801 25.2%	
http://localhost:3000/almond.css	CSS	24 216	21 155 87.4%	
/node_modules_web-vitals_base_js.c	JS (per function)	11 020	4 643 42.1%	
http://localhost:3000/	JS (per function)	1 696	252 14.9%	
http://localhost:3000/chunk.1kb.css	CSS	76	37 48.7%	

LOAD CSS ~0.5KB

LOAD CSS ~27KB

Elements Console Recorder Performance insights Sources Network

Line 8, Column 1 Coverage: 51.3%

Console Performance monitor Rendering Network conditions Coverage x What's New Issues

Per function localhost All Content scripts

URL	Type	Total Bytes	Unused Bytes	Usage Visualiz...
http://localhost:3000/static/js/bundle.js	JS (per function)	384 004	95 860 25%	
http://localhost:3000/almond.css	CSS	24 216	21 547 89%	
http://localhost.../node_modules_web-vitals_base_js.chunk.js	JS (per function)	11 020	5 736 52.1%	
http://localhost:3000/	JS (per function)	1 696	1 078 63.6%	



Каков бюджет?

0.5~3KB ``br``

BUNDLE SIZE

5.5kB

MINIFIED

2kB

MINIFIED + GZIPPED

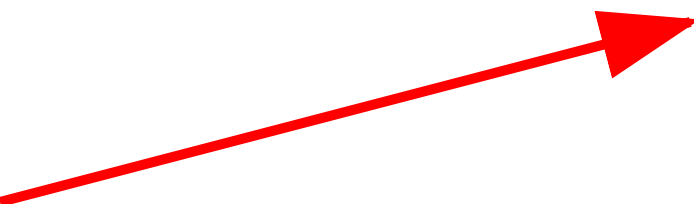
DOWNLOAD TIME

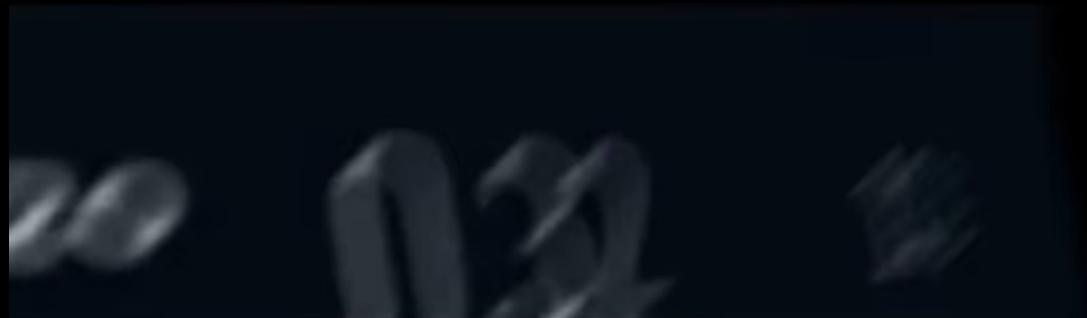
41ms

SLOW 3G ⓘ

2ms

EMERGING 4G ⓘ






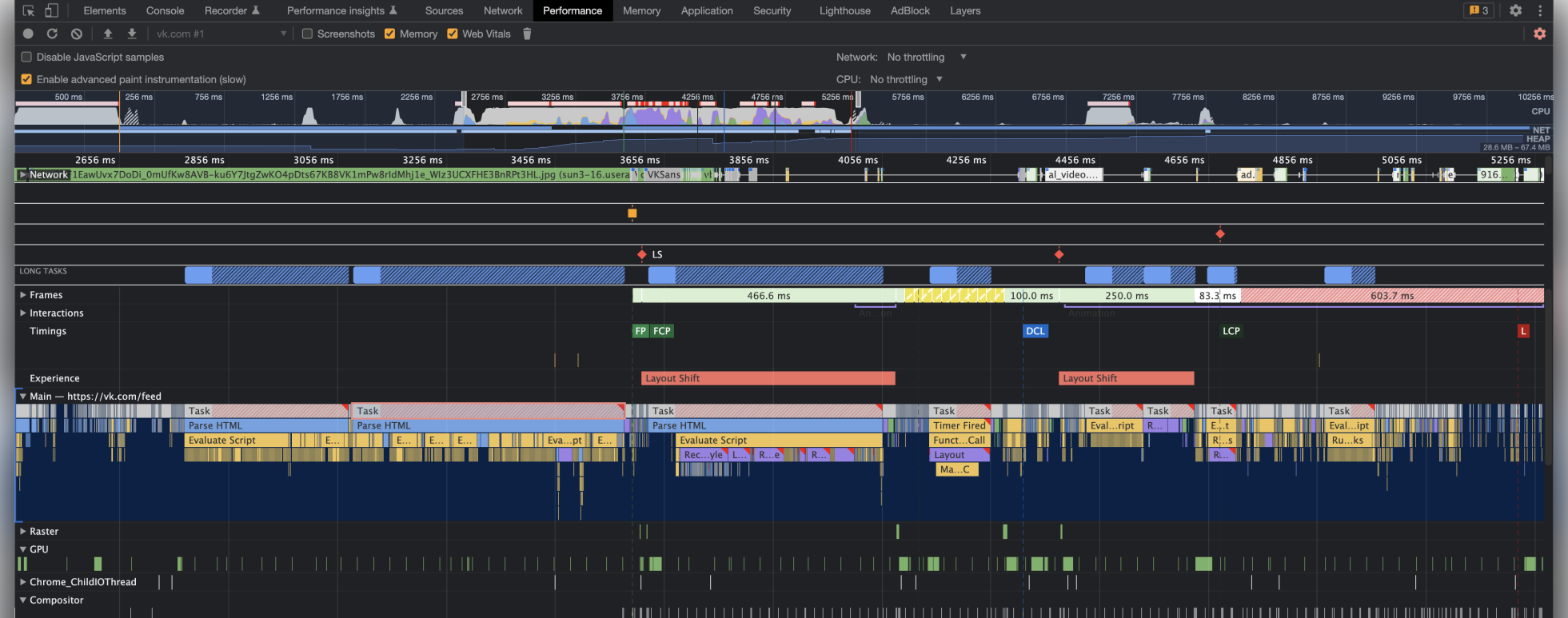
Sustainable Loading

Performance

#FID, #FCP, #LCP, #CLS

DevTools → Performance:

- Web Vitals Marks
- Screenshots
- Console  Sources



Summary **Bottom-Up** Call Tree Event Log

Filter No Grouping

Self Time	Total Time	Activity
183.8 ms 36.8 %	418.9 ms 84.0 %	Evaluate Script
183.0 ms 36.7 %	348.8 ms 69.9 %	Parse HTML
0.8 ms 0.2 %	4.1 ms 0.8 %	Evaluate Script
178.0 ms 35.7 %	178.0 ms 35.7 %	Compile Code
177.0 ms 35.5 %	177.0 ms 35.5 %	Evaluate Script
0.9 ms 0.2 %	0.9 ms 0.2 %	Compile Script
0.1 ms 0.0 %	0.1 ms 0.0 %	Event: load
42.2 ms 8.5 %	498.8 ms 100.0 %	Parse HTML
4.3 ms 0.9 %	12.7 ms 2.5 %	Evaluate Script
41.6 ms 8.3 %	41.6 ms 8.3 %	Parse Stylesheet
26.0 ms 5.2 %	26.0 ms 5.2 %	Recalculate Style


**Спасибо
за внимание!**



Performance

#FID, #FCP, #LCP, #CLS

DevTools → Performance:

- Web Vitals Marks
- Screenshots
- Console  Sources

 **Performance insights** 



DevTools → Performance insights

The image shows a screenshot of the Chrome DevTools interface. The 'Run' command palette is open, displaying a list of commands to show various panels. The 'Show Application' command is currently selected and highlighted in blue. Each command has a corresponding cyan 'Panel' button to its right. The background shows the DevTools tabs: Elements, Console, Recorder, Sources, Network, and Performance.

Command	Action
Show Application	Panel
Show CSS Overview	Panel
Show Console	Panel
Show Elements	Panel
Show JavaScript Profiler	Panel
Show Layers	Panel
Show Lighthouse	Panel

User-case Performance

#FID, #FCP, #LCP, #CLS, #INP

DevTools → Performance insights:

- Panel
- Session Window 👁️
- INP 👁️



Demo App #2

SINGLE LONG TASK 50MS

LONG TASK 1000MS



Листай ниже ↓

Листай ниже ↓

Листай ниже ↓

Листай ниже ↓

Листай ниже ↓

Листай ниже ↓

LOAD CSS ~0.5KB

LOAD CSS ~27KB

👁️ А что за код?

 /src/index.ts

```
1 // Idle ~100ms
2 requestIdleCallback(notImportantTask);
3
4 // Async ~75ms
5 setTimeout(standaloneTask, 0);
6
7 // Sync 30~50ms
8 task();
```

Chromium DevTools Performance tab showing a waterfall chart and task analysis for a page load on localhost #17.

Options: Disable JavaScript samples, Enable advanced paint instrumentation (slow), Network: No throttling, CPU: No throttling.

Waterfall chart shows network requests with durations: 37 ms, 87 ms, 137 ms, 187 ms, 237 ms, 287 ms, 337 ms, 387 ms, 437 ms, 487 ms, 537 ms.

LONG TASKS: A task is highlighted with a duration of 156.8 ms.

Frames: FP, FCP, LCP, DCL, L.

Experience: Main — http://localhost:3000/

Task breakdown:

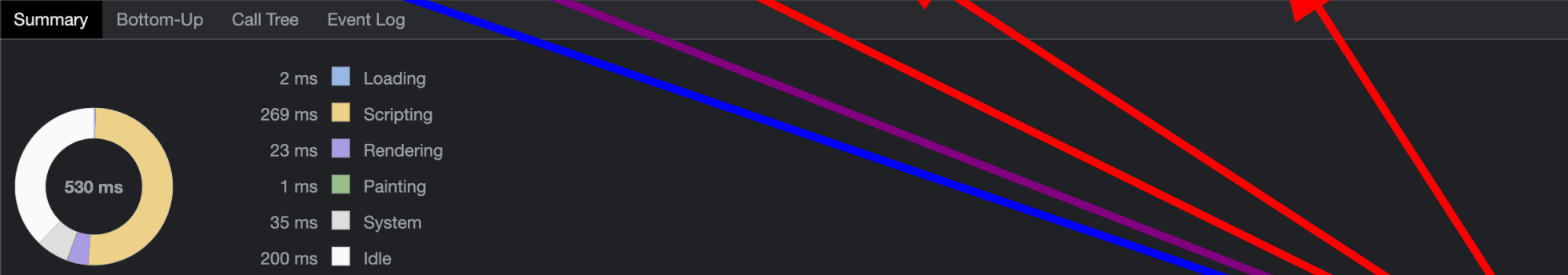
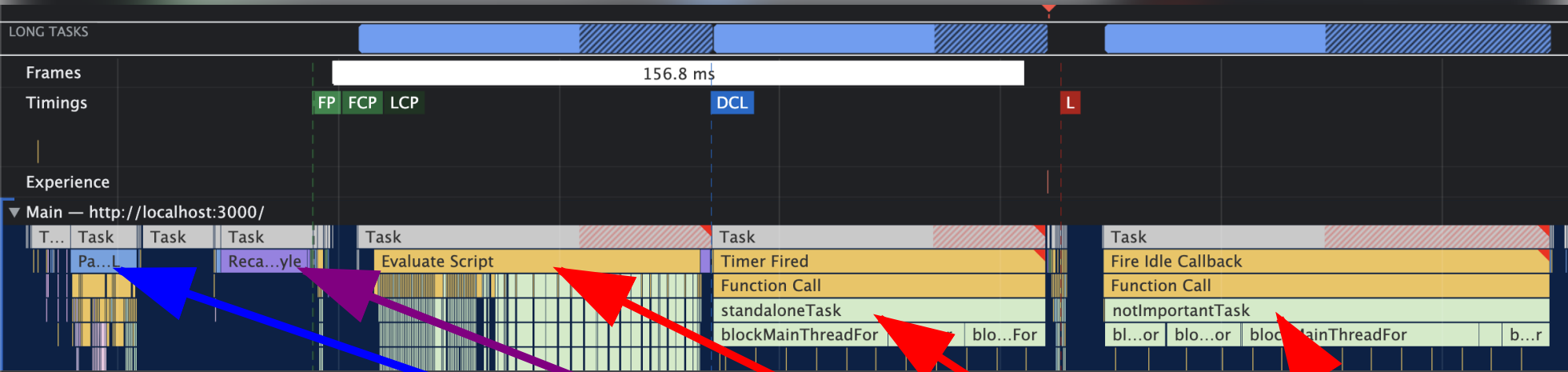
- Task 1: Pa...L
- Task 2: Reca...yle
- Task 3: Evaluate Script
- Task 4: Timer Fired
- Task 5: Fire Idle Callback
- Task 6: Function Call
- Task 7: standaloneTask
- Task 8: notImportantTask
- Task 9: blockMainThreadFor blo...or blo...For
- Task 10: bl...or blo...or blockMainThreadFor b...r

Summary: Bottom-Up, Call Tree, Event Log

Activity Legend:

- 2 ms Loading
- 269 ms Scripting
- 23 ms Rendering
- 1 ms Painting
- 35 ms System
- 200 ms Idle
- 530 ms Total

Total blocking time: 106.76ms (estimated) [Learn more](#)

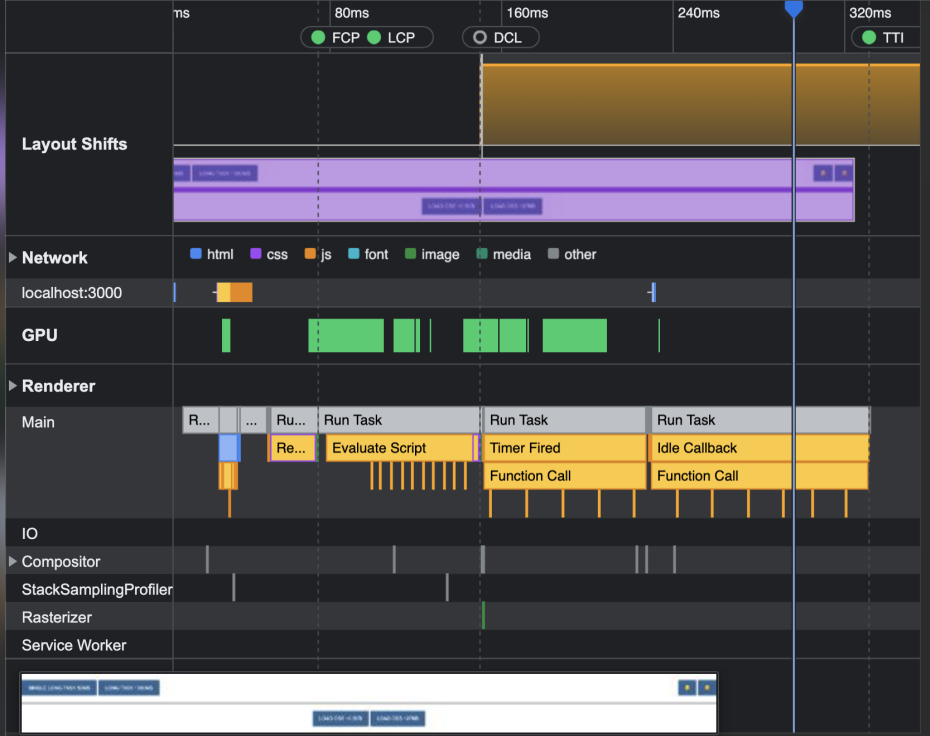


SINGLE LONG TASK 50MS LONG TASK 1000MS

LOAD CSS ~0.5KB LOAD CSS ~27KB

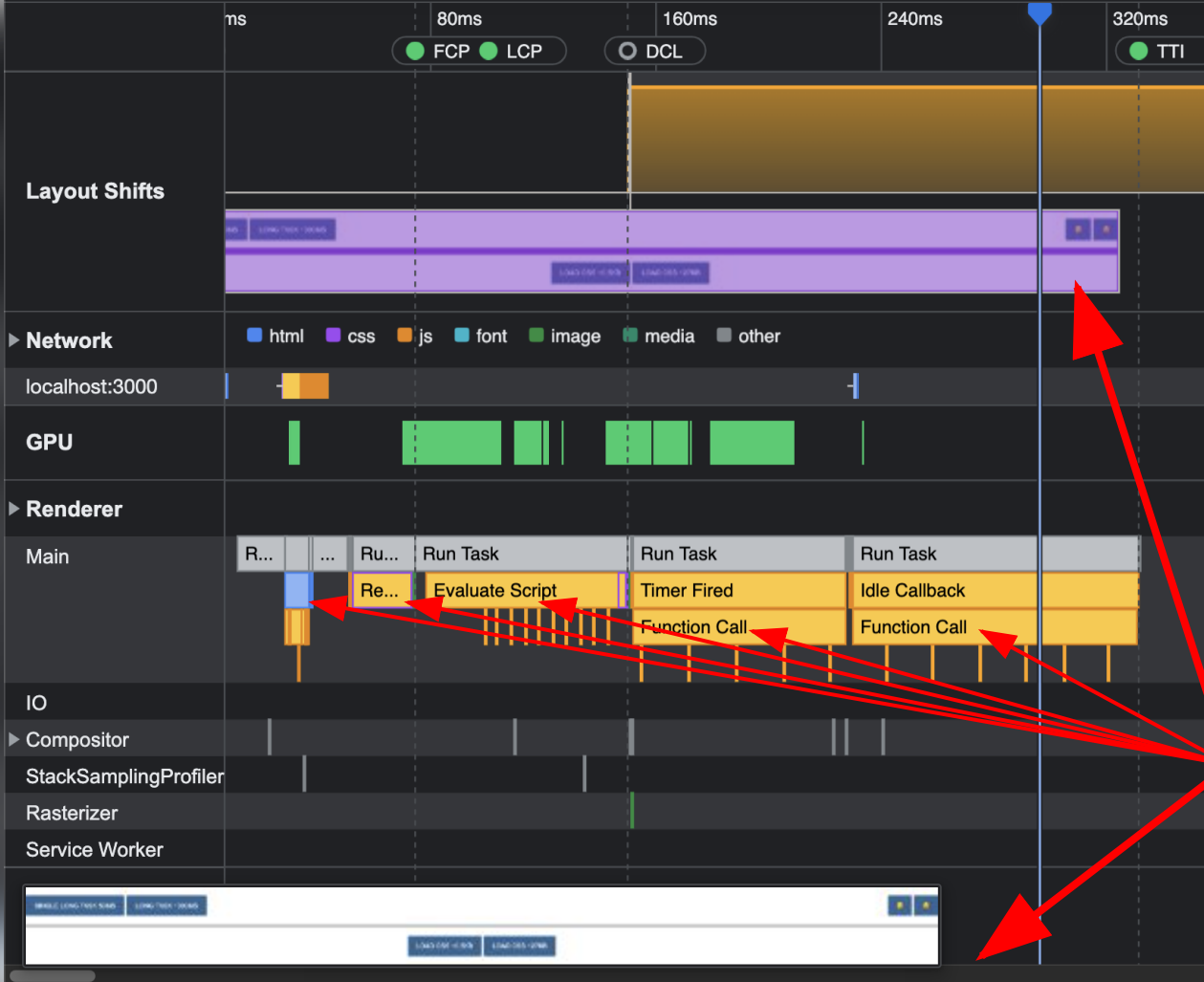
Performance insights

No throttling localhost (1)



Insights Details

- Navigation Event <http://localhost:3000/>
- [Render blocking request](#)
- First Contentful Paint 0.07s
- Largest Contentful Paint 0.07s
- [Long task](#)
- DOM Content Loaded 0.15s
- [Layout shift](#)
- Cumulative Layout Shift 0.217
- [Long task](#)
- [Long task](#)



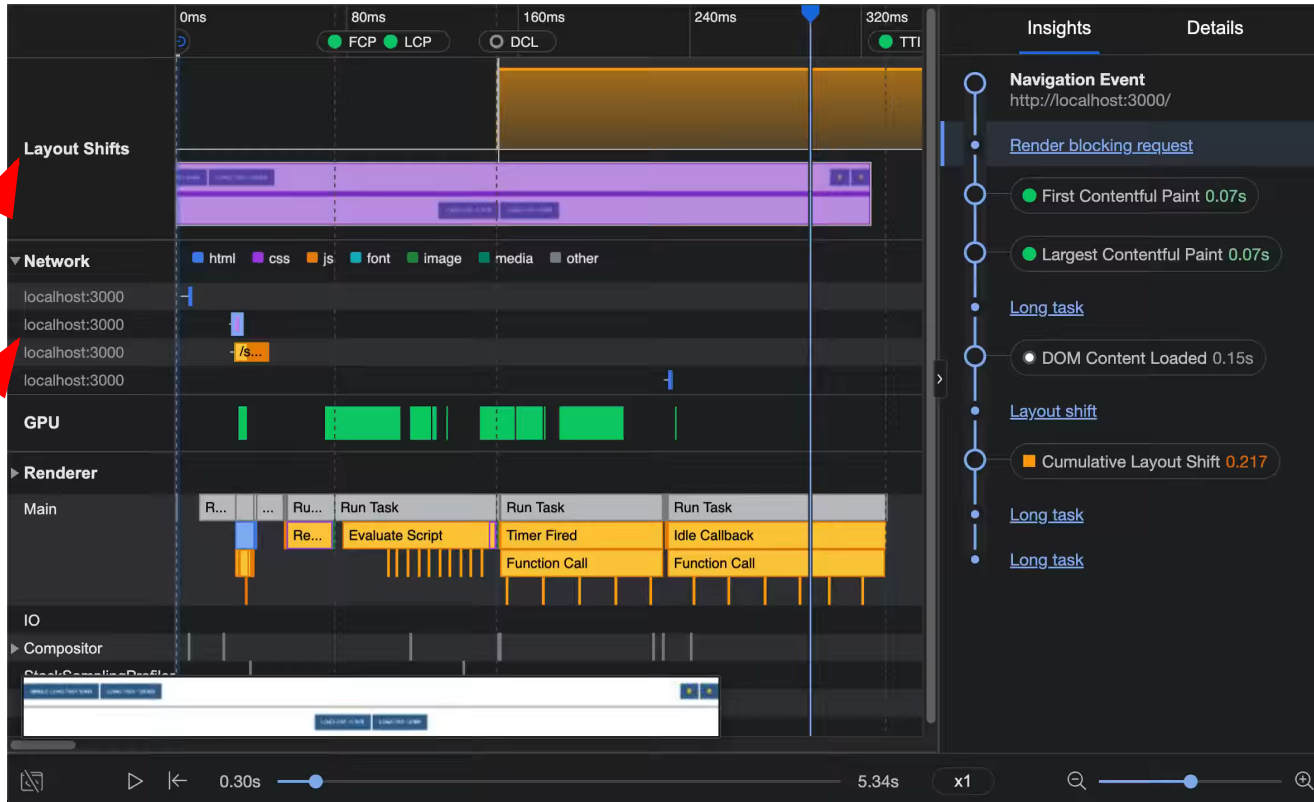
Insights Details

- **Navigation Event**
http://localhost:3000/
- [Render blocking request](#)
- ● **First Contentful Paint 0.07s**
- ● **Largest Contentful Paint 0.07s**
- [Long task](#)
- ● **DOM Content Loaded 0.15s**
- [Layout shift](#)
- ● **Cumulative Layout Shift 0.217**
- [Long task](#)
- [Long task](#)



DevTools → Performance insights

Layout Shifts, Network



DevTools → Performance insights

Player

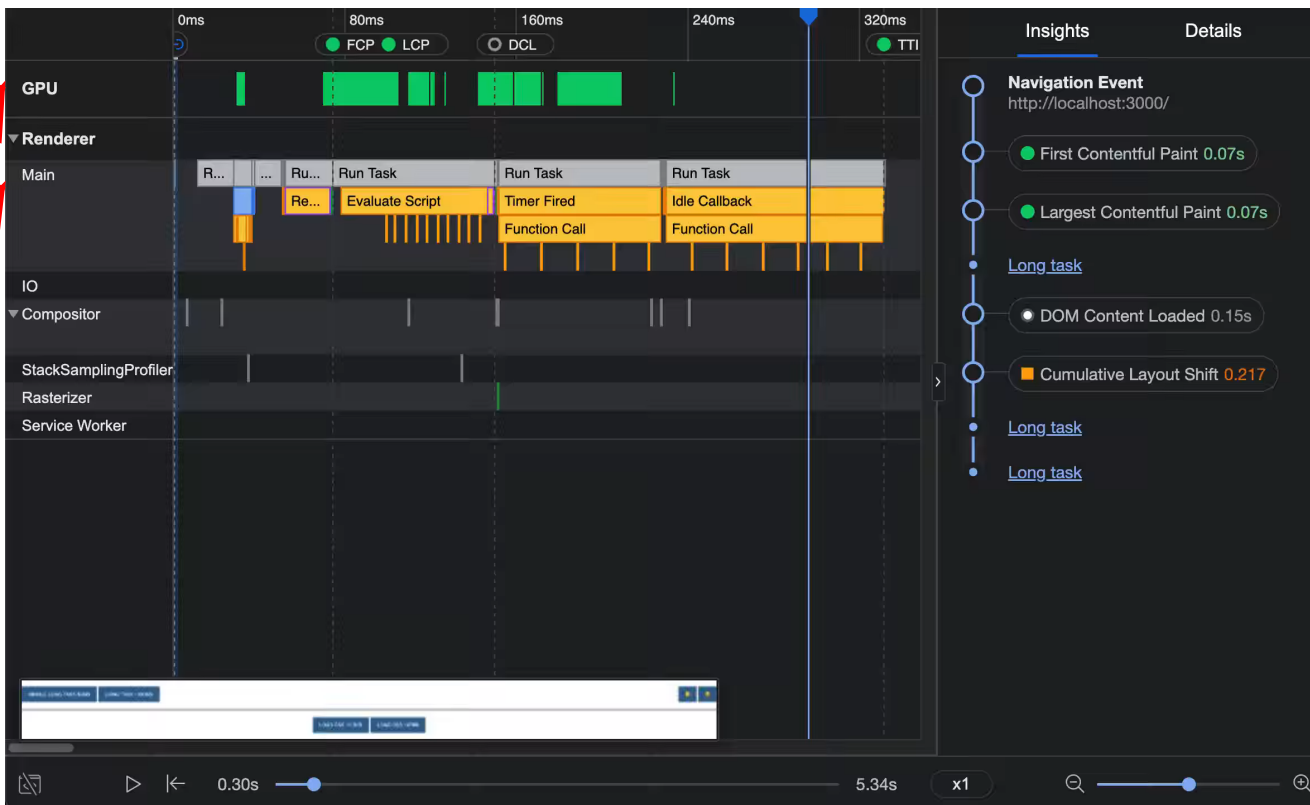


The screenshot displays the Chrome DevTools Performance Insights interface. At the top, the 'Performance insights' tab is active, showing a red recording button, a 'No throttling' dropdown, and the target 'youtube.com (1)'. The interface includes a 'Details' panel on the right with the instruction: 'Select a trace event in the timeline to view its details.' The main area shows a video player with a cow. The video player has a progress bar at the bottom showing '0.00s' out of '6.95s' and a zoom level of 'x1'. The video player's address bar shows 'youtube.com' and a search bar with the text 'Введите запрос'.



DevTools → Performance insights

GPU, Renderer

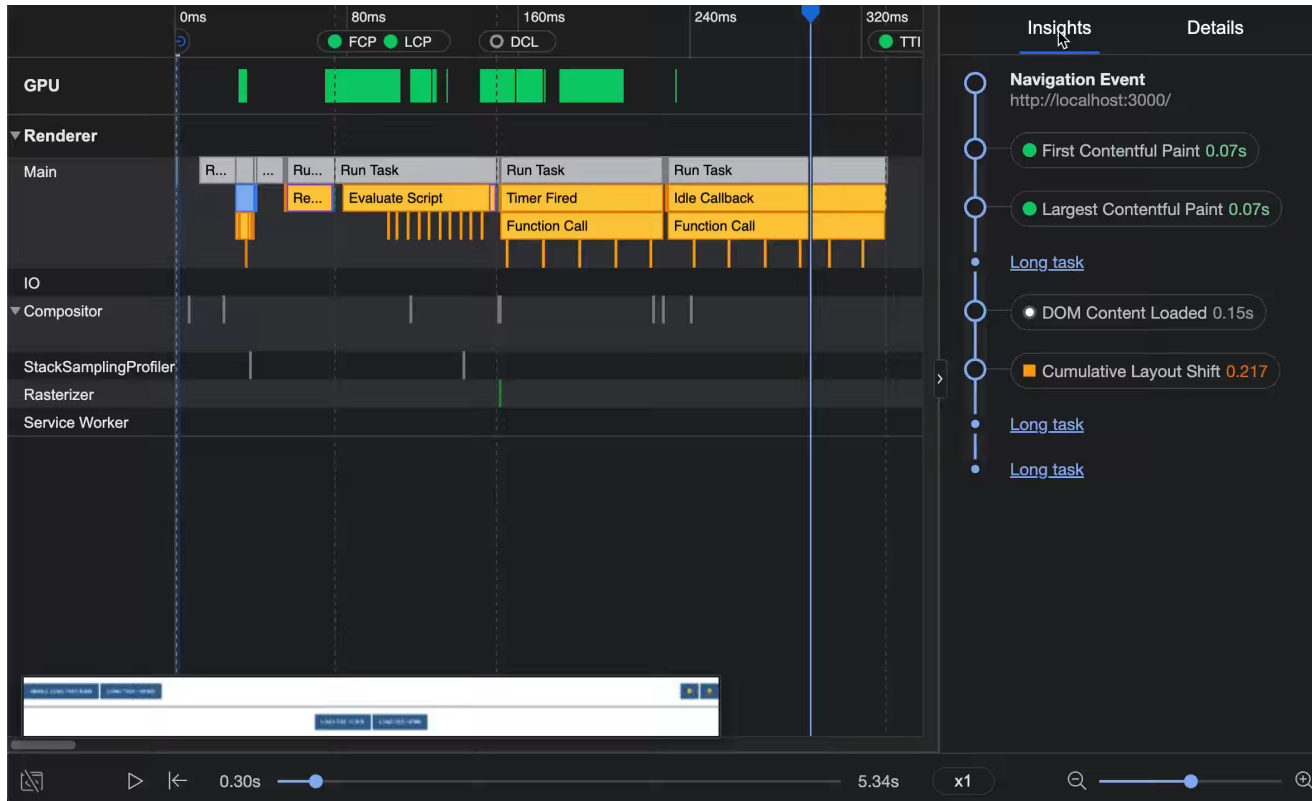




DevTools → Performance insights



Long Tasks



User-case Performance

#FID, #FCP, #LCP, #CLS, #INP

DevTools → Performance insights:

- Panel
- CLS & Session Window 👁️
- INP 👁️





Demo App #3



DevTools → Performance insights

Session Window





Experience

Layo...ift

Layout Shift

Occurrences: 1 Layout Shift

Summary

Bottom-Up

Call Tree

Event Log

Layout Shift

Warning [Cumulative Layout Shifts](#) can result in poor user experiences. It has recently [evolved](#).

Score 0.02510

Cumulative Score 0.07748

Current Cluster ID 1

Current Cluster Score 0.07748

Had recent input No

Moved from Location: [0,332], Size: [3554x396]

Moved to Location: [0,496], Size: [3554x232]

Related Node `div`

User-case Performance

#FID, #FCP, #LCP, #CLS, #INP

DevTools → Performance insights:

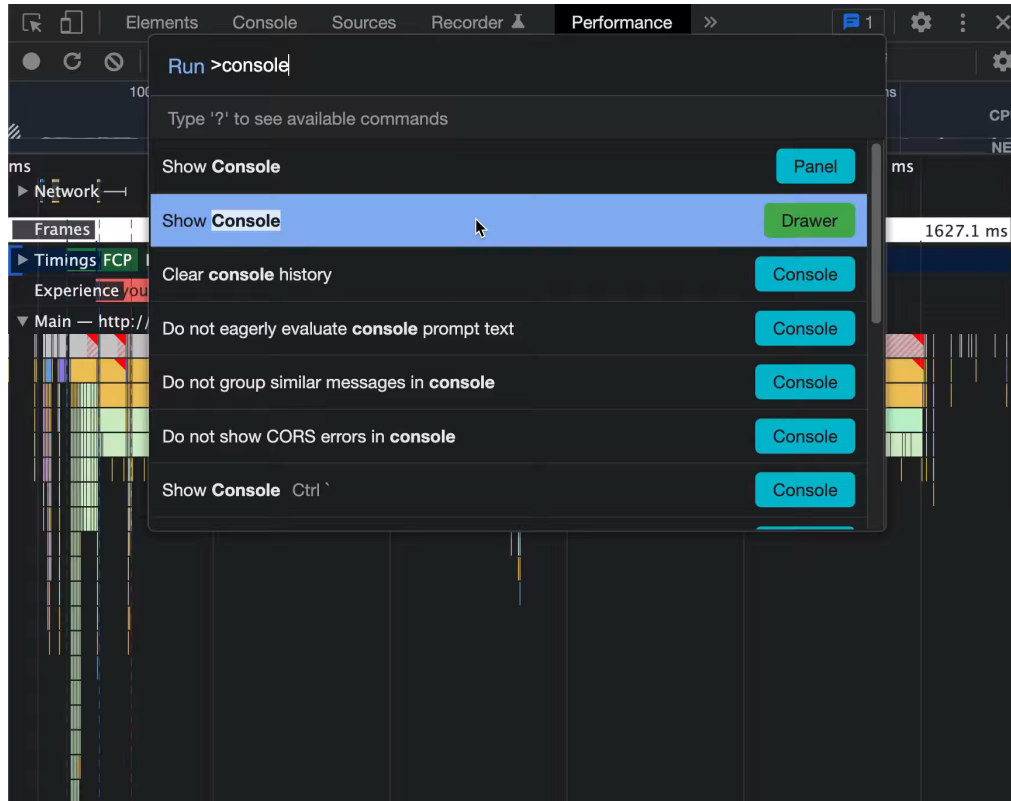
- Panel
- CLS & Session Window 👁️👁️
- INP 👁️👁️



Demo App #4

DevTools → Performance

Timings



The screenshot shows the Chrome DevTools Performance panel. A red arrow points from the left towards the 'Timings' section in the left-hand sidebar. A context menu is open over the 'Timings' section, listing various actions:

- Run > console
- Type '?' to see available commands
- Show Console Panel
- Show Console Drawer
- Clear console history
- Do not eagerly evaluate console prompt text
- Do not group similar messages in console
- Do not show CORS errors in console
- Show Console Ctrl`

The 'Timings' section in the sidebar is expanded, showing 'FCP' (First Contentful Paint) and 'Experience' (User Experience) metrics. The 'Main' frame is selected, showing a timeline of events with a red arrow pointing to a specific event.



DevTools → Performance insights

)



Check out what's new for the web platform at Google IO!

[Join the party](#)

[Dismiss](#)



Aw, Snap!

We couldn't find that page.

[Go back home](#)

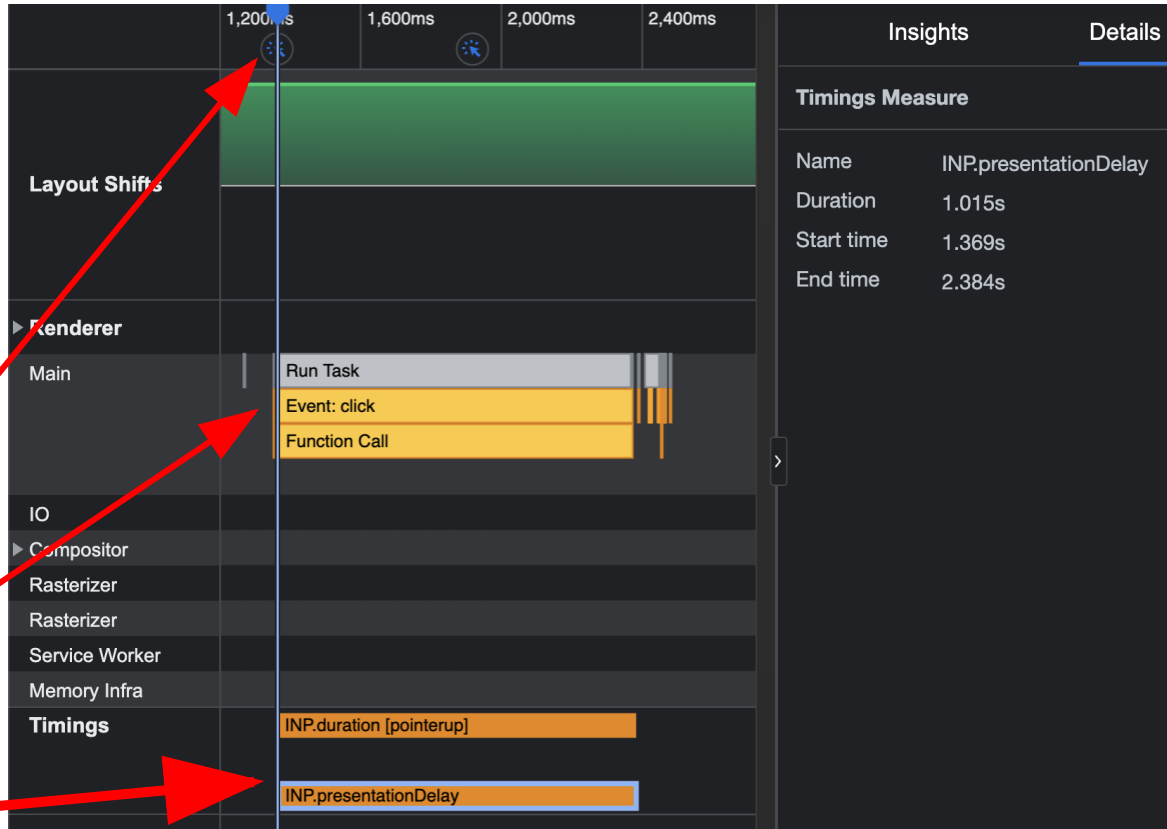


Сейчас бы теорий не от GDE послушать...



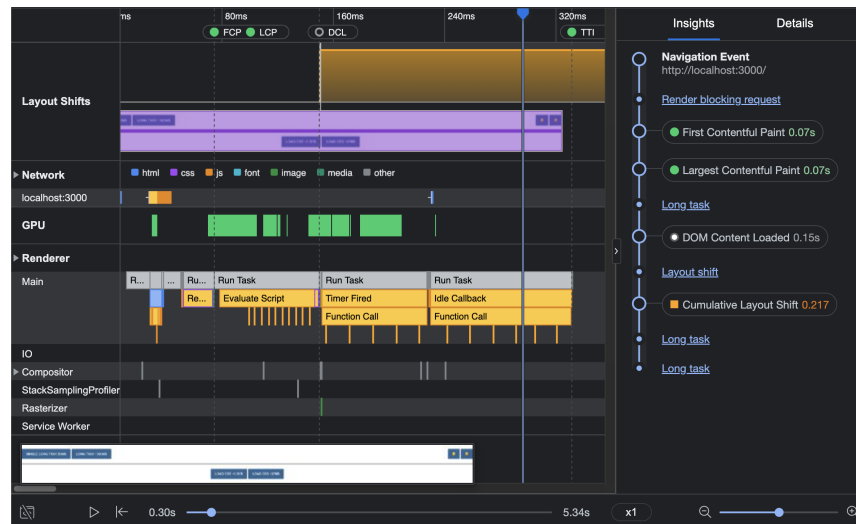
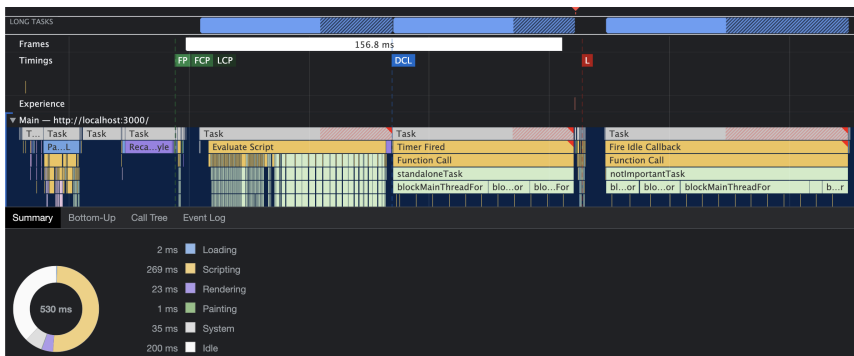
DevTools → Performance insights

Timings



DevTools

Performance 🤔 Performance insights



Performance  **Performance insights**

,

 **Найти**



Bald prediction



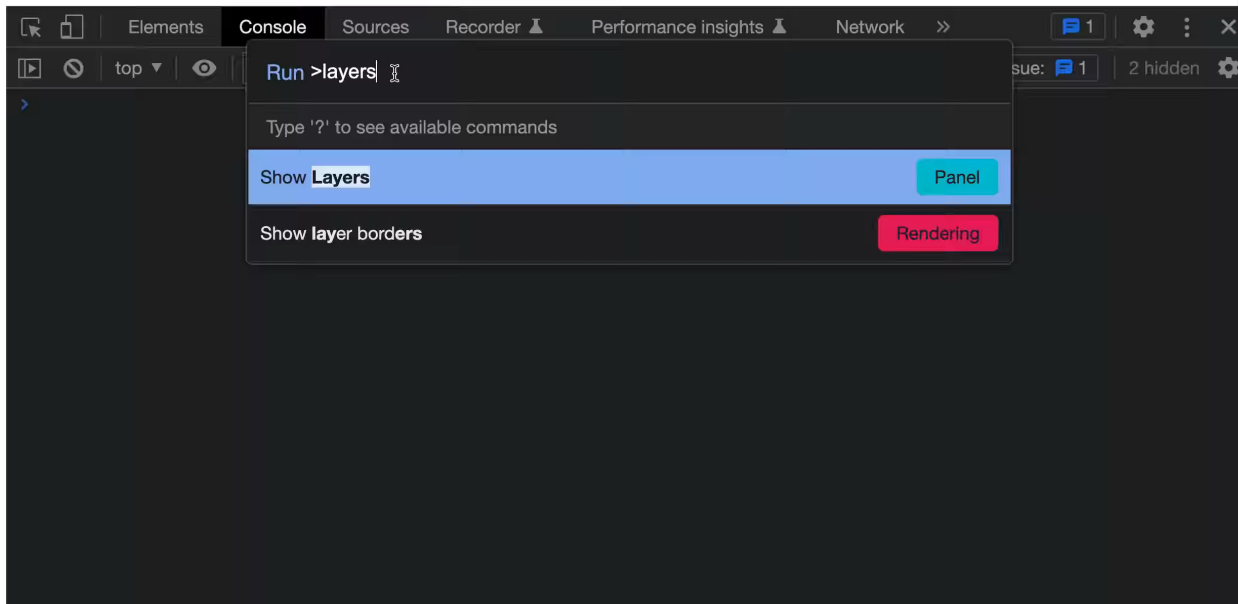
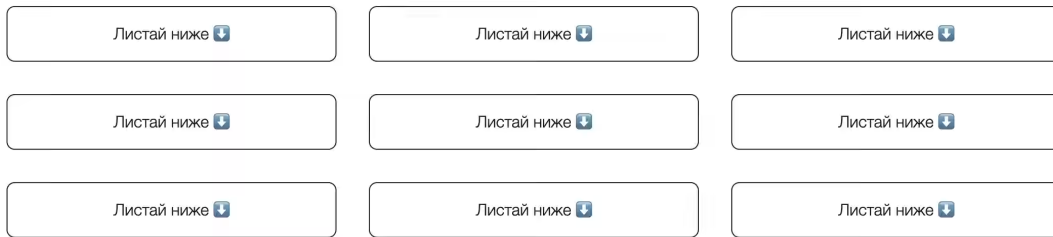
Rendering Performance

#TBT, #CLS, #INP, #FPS

DevTools:

- Layers ?

DevTools → Layers



Rendering Performance

#TBT, #CLS, #INP, #FPS

DevTools:

- Layers 🙄
- Resource usage ?



DevTools → Performance monitor

Rendering Performance

#TBT, #CLS, #INP, #FPS

DevTools:

- Layers 🙄
- Resource usage 😐
- Repaints ?



DevTools → Paint flashing

The screenshot shows the DevTools interface with the Rendering panel open. The top part of the interface displays a social media profile with a navigation menu on the left and a profile picture on the right. The bottom part of the interface shows the DevTools toolbar with the following tabs: Elements, Console, Sources, Recorder, and Rendering. The Rendering tab is active, and the 'Paint flashing' checkbox is checked. Below the checkbox, the text reads: 'Paint flashing Highlights areas of the page (green) that need to be repainted.'

Rendering Performance

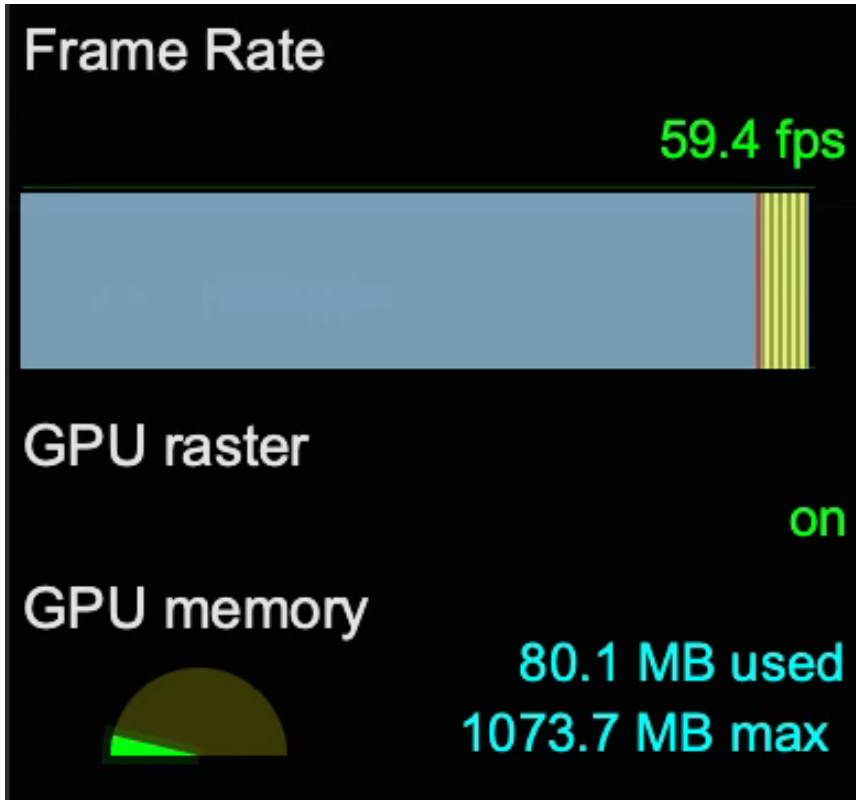
#TBT, #CLS, #INP, #FPS

DevTools:

- Layers 😬
- Resource usage 😬
- Repaints 😬
- Frame Rendering Stats ?



DevTools → Frame Rate



ChromeFrameReporter: `State` & `FrameDropReason`

```
1 // The frame was presented with some updates, but also missed some updates
2 // (e.g. missed updates from the main-thread, but included updates from the
3 // compositor thread).
4 State['STATE_PRESENTED_PARTIAL'];
5
6 // The frame was dropped, i.e. some updates were desired, but wasn't presented.
7 State['STATE_DROPPED'];
8
9 // Frame was dropped by the display-compositor.
10 // The display-compositor may drop a frame some times (e.g. the frame missed
11 // the deadline, or was blocked on surface-sync, etc.)
12 FrameDropReason['REASON_DISPLAY_COMPOSITOR'];
13
14 // Frame was dropped because of the main-thread.
15 // The main-thread may cause a frame to be dropped, e.g. if the main-thread
16 // is running expensive javascript, or doing a lot of layout updates, etc.
17 FrameDropReason['REASON_MAIN_THREAD'];
18
19 // Frame was dropped by the client compositor.
20 // The client compositor can drop some frames too (e.g. attempting to
21 // recover latency, missing the deadline, etc.).
22 FrameDropReason['REASON_CLIENT_COMPOSITOR'];
```

🕶️ Partially presented frames



developer.chrome.com/blog/new-in-devtools-100/#perf



Rendering Performance

#TBT, #CLS, #INP, #FPS

DevTools:

- Layers 🙄
- Resource usage 😐
- Repaints 🙄
- Frame Rendering Stats 🤔





- ✓ Рабочее окружение
- ✓ Метрики
- ✓ Sustainable Loading
- ✓ Performance

PROBLEMS ARE COOL ♡

ПРИВЕТСТВУЕМ В КОМАНДЕ

- 0 Кому вы хотите присоединиться?
- 0 Почему вы хотите присоединиться?
- 0 Почему вы хотите присоединиться?

Волонтеры приветствуют в Команде только лучших. Если ты читаешь эти строки, подумай — тебе есть чем поделиться?

Теперь ты — важная часть Команды, которая каждый день поддерживает и развивает наше сообщество волонтеров в крупнейшую социальную сеть России. Спасибо! Волонтеры объединяют, развивают, просвещают, создают новые проекты. Спасибо! Бесплатные возможности каждого пользователя. И для каждого сотрудника.

Но ты не просто состоишь в команде, ты становишься ее частью. Ты живешь, дышишь, работаешь. Ты учишься, растешь, развиваешься. Ты учишься, растешь, развиваешься.

Ты учишься, растешь, развиваешься. Ты учишься, растешь, развиваешься. Ты учишься, растешь, развиваешься.

Ты учишься, растешь, развиваешься. Ты учишься, растешь, развиваешься. Ты учишься, растешь, развиваешься.



872222222

872222222

VK @bel

TW @belov

