

THE FUTURE OF FRONT-END PERFORMANCE

Sia Karamalegos

hi, i'm sia



Take out a piece of paper...

implement
(donow)

research
(interested)

seek assistance

WHY DO ELEVATORS HAVE MIRRORS?

WHY SHOULD I CARE?

Pinterest reduced load times by 40% and saw a 15% increase in sign ups.

<https://wpostats.com/>

*Starbucks implemented a 2x faster time
to interactive resulting in a 65%
increase in rewards registrations.*

Chrome Dev Summit

AliExpress reduced load by 36% and saw a 10.5% increase in orders.

<https://wpostats.com/>

Speed is now used as a ranking factor for mobile searches.

<https://developers.google.com/web/updates/2018/07/search-ads-speed>

MEASUREMENT AND ANALYSIS

Pareto Principle

Roughly 80% of the effects come from 20% of the causes.

Be lazy. Only optimize the worst offenders.



WHICH METRICS MATTER?

~~Load time~~ Speed Index

Measures how quickly the page contents are visually populated

- Expressed in milliseconds
- Dependent on size of the view port
- Use webpagetest.org to measure your pages

Time to Interactive

Interactivity Matters



End to End Apps with Polymer by Kevin Schaaf, Polymer Summit 2017

Jank or Responsiveness



MEASUREMENT

Synthetic Testing

Use WebPageTest and DevTools network tab to optimize load and speed index.

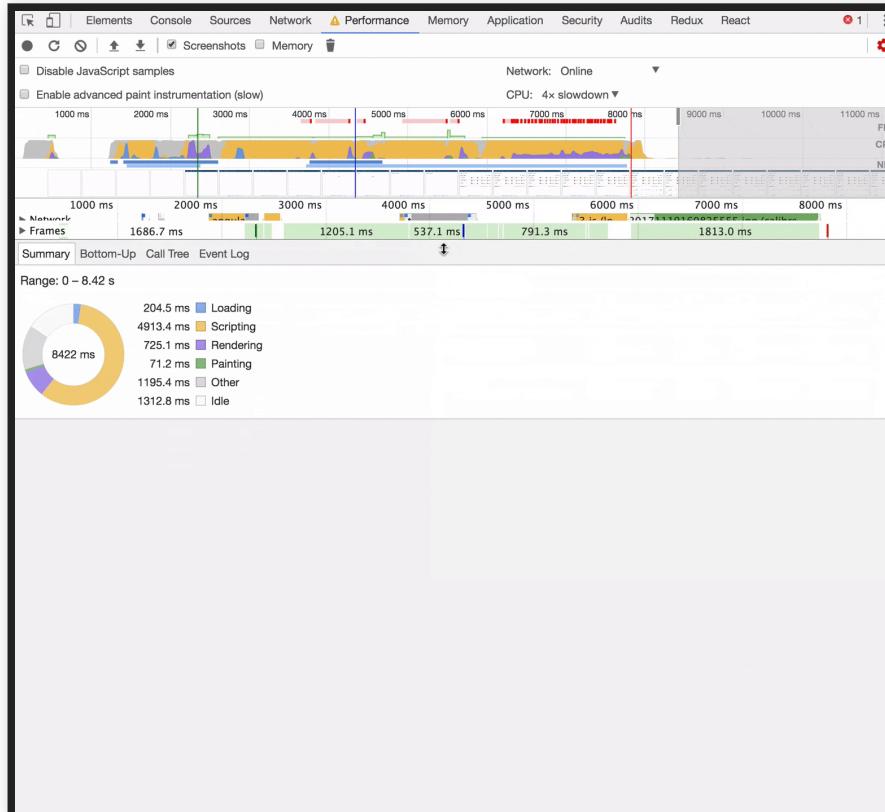
The screenshot shows the Chrome DevTools Network tab for the URL <https://soundcloud.com>. The timeline at the top displays various network requests with their start and end times. A large circular play button is overlaid on the timeline. Below the timeline is a table listing network requests:

Name	Met...	St...	Type	Initiator	Size	Time	Waterfall
sci?pid=9001...	GET	200	te...	ca.htm...	126 B	1.22	
sync?uid=459...	GET	204	te...	ca.htm...	92 B	94 ms	
?taboola_hm=...	GET	204	te...	ca.htm...	324 B	91 ms	
bct?pid=b151...	GET	200	gif	ca.htm...	348 B	175 ...	
cksync.php?c...	GET	200	gif	ca.htm...	45 B	830 ...	
?id=90223569...	GET	200	gif	fbeven...	98 B	89 ms	
pstats.html?rb...	GET	200	do...	tc.min....	1.5 KB	292 ...	

At the bottom of the DevTools interface, there is a summary: "147 requests | 2.7 MB transferred | Finish: 30.33 s | DOMContentLoaded: 3.13 s | Load: 12.77 s". The browser's main window shows a SoundCloud page with a red background and a person's face.

Synthetic Testing

Use DevTools performance tab to optimize responsiveness.



Real User Monitoring (RUM)

- Navigation Timing API
- Resource Timing API
- User Timing API for custom timings

<https://developers.google.com/web/fundamentals/performance/navigation-and-resource-timing/>
<https://www.keycdn.com/blog/user-timing/>

Optimize for the device and network your users have

- 2-5x difference in fastest vs slowest phones
- 75% of worldwide mobile connections on 2G or 3G
- Not just developing countries but rural areas or spotty networks like conference wifi
- Use Google Analytics data to profile your users and configure webpagetest.org to reflect them more closely
- Set performance budgets using webpack

<https://infrequently.org/2017/10/can-you-afford-it-real-world-web-performance-budgets/>

BIGGEST BYTES

Images account for 39-43% of the bytes on average
needed to load a webpage.

<httparchive.org>, September 2018

Image Optimization Toolbox

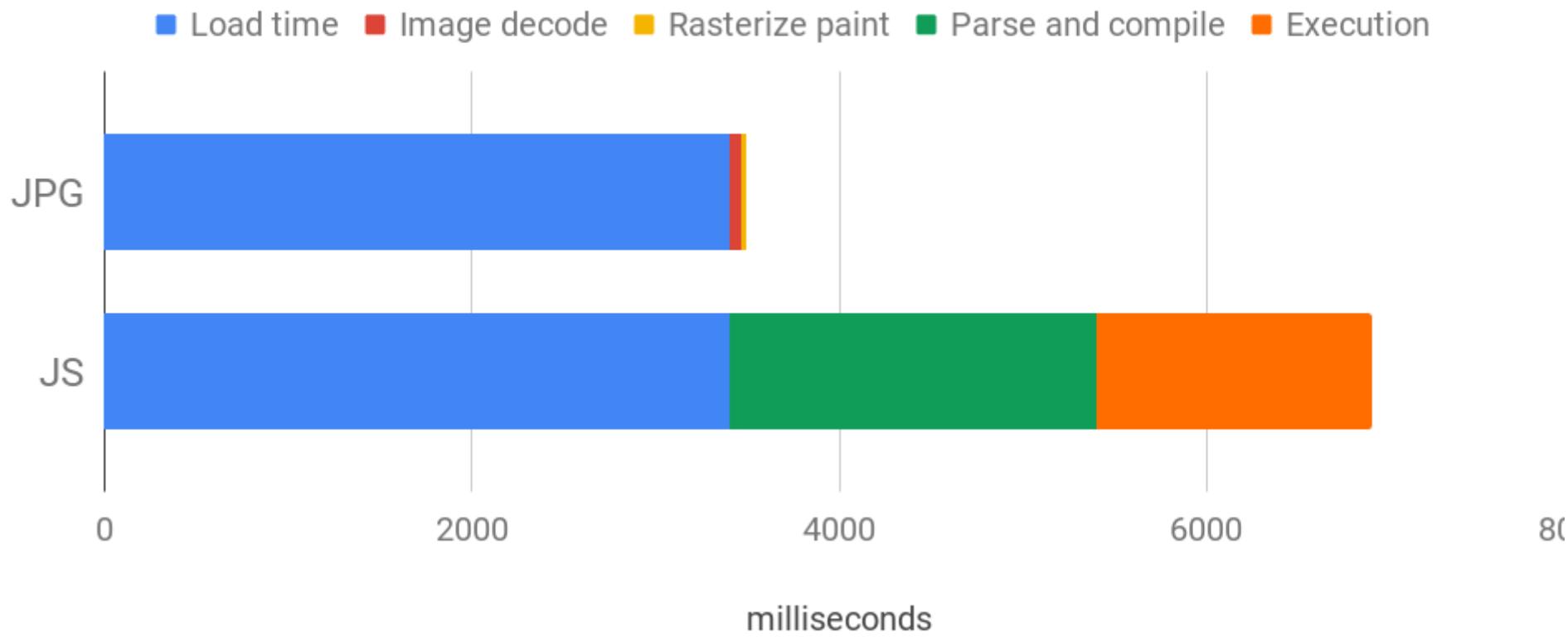
- Use the right image type (png vs jpg, gif vs video).
- Use the right size and src sets, and webpack loaders to auto-build src sets.
- Compress images with a tool like ImageOptim, or use a webpack plugin to auto-optimize them for you.
- Use newer, improved formats like webp.

<https://www.udacity.com/course/responsive-images--ud882>

<https://survivejs.com/webpack/loading/images/#optimizing-images>

MOST EXPENSIVE ASSET

170 kB JPG vs 170 kB JS



Adapted from [The Cost of JavaScript](#) by Addy Osmani

TL;DR: Ship less code

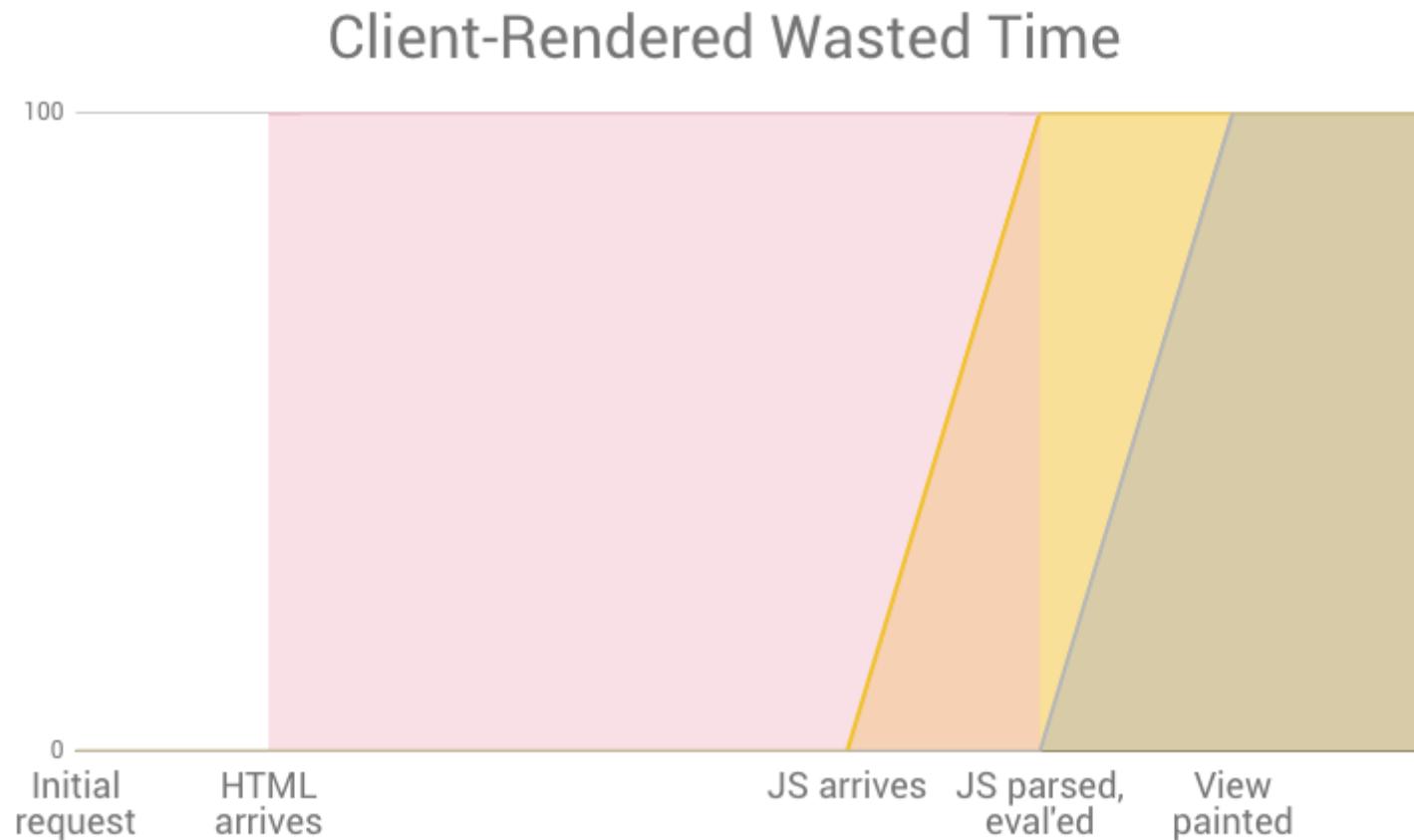
- less code = less load + less parse/compile
- holy grail = prioritize only what's needed in view

Client vs Server vs Progressive Rendering



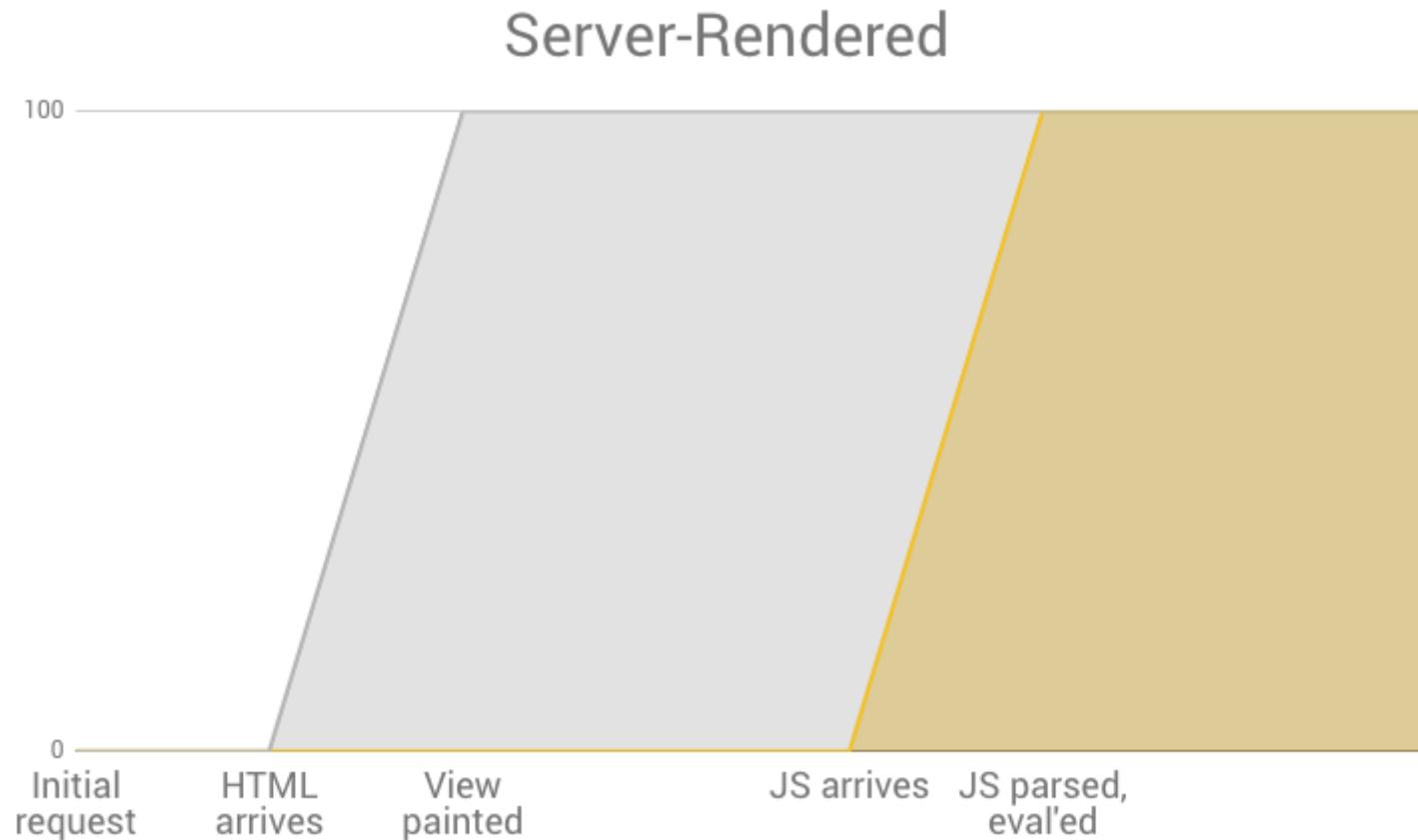
Inspired by <https://twitter.com/aerotwist/status/729712502943174657>

Client vs Server vs Progressive Rendering



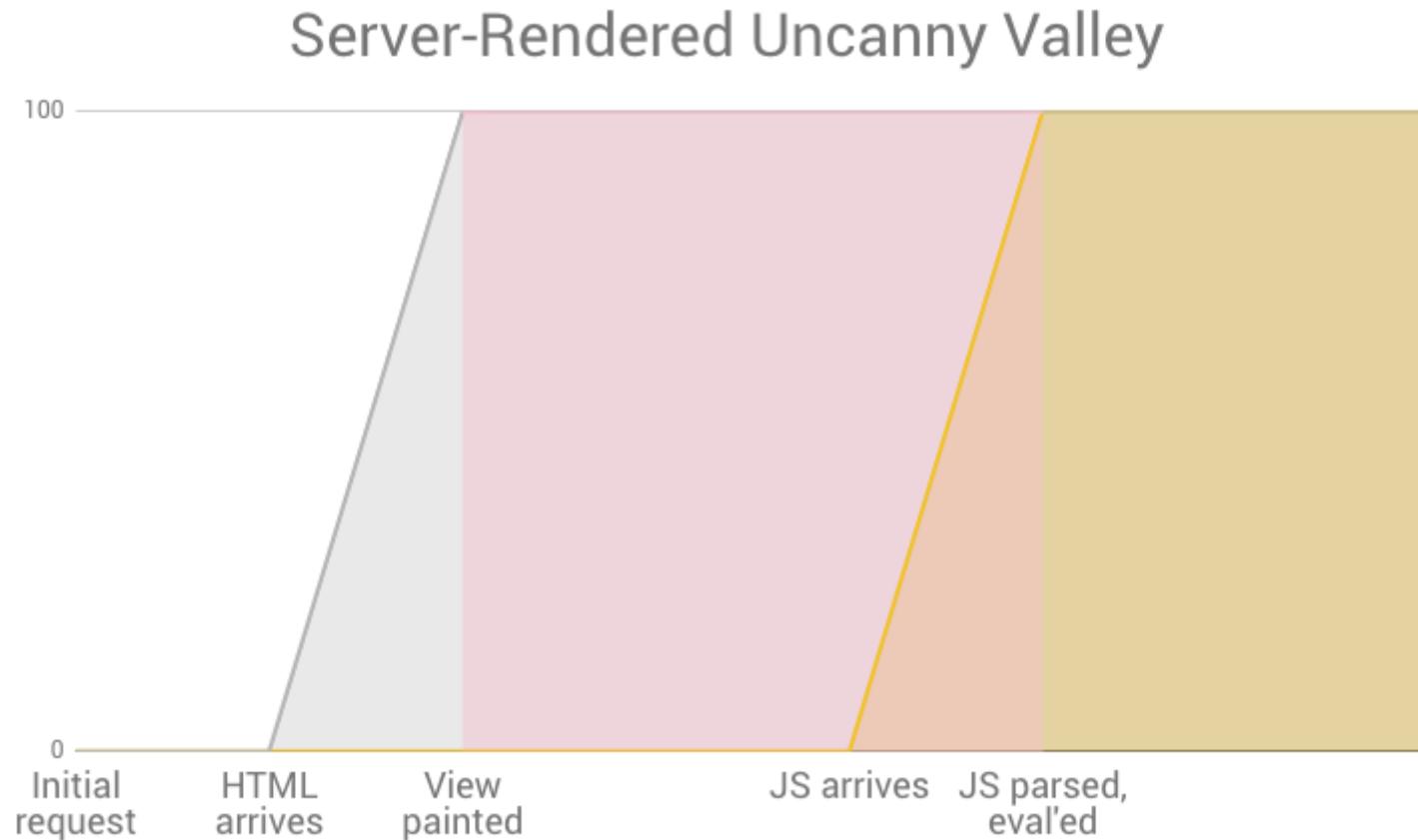
Inspired by <https://twitter.com/aerotwist/status/729712502943174657>

Client vs Server vs Progressive Rendering



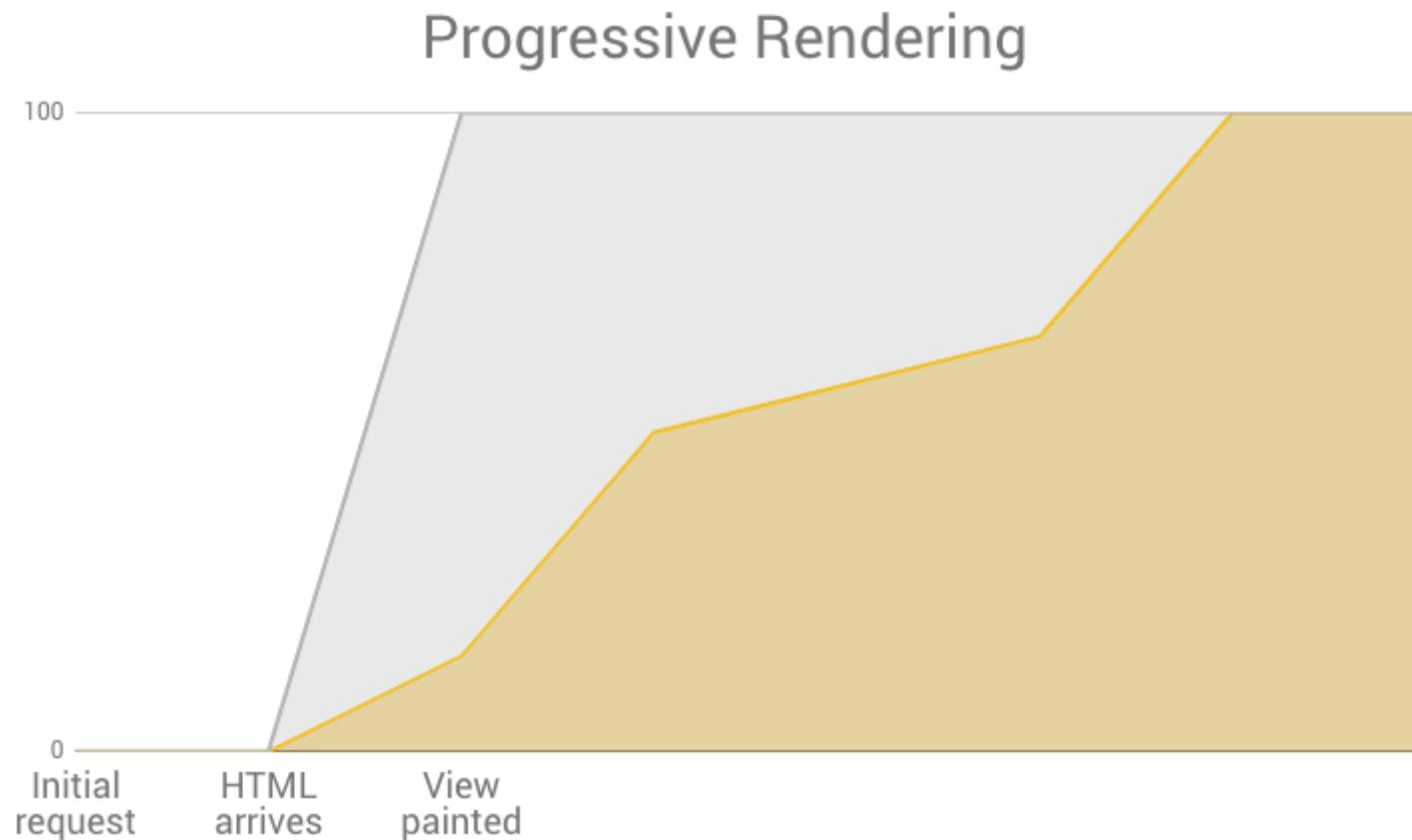
Inspired by <https://twitter.com/aerotwist/status/729712502943174657>

Client vs Server vs Progressive Rendering



Inspired by <https://twitter.com/aerotwist/status/729712502943174657>

Client vs Server vs Progressive Rendering



Inspired by <https://twitter.com/aerotwist/status/729712502943174657>

Optimizing Time to Interactive

- Analyze your loads and bundles! Don't over-optimize!
- Only ship what's immediately needed - use code splitting, pre-caching, and deferred or lazy loading.
- Minify to speed up both download and parse/compile.
- Compress with gzip or brotli.
- Remove unused code with tree shaking and using module imports effectively.

Module Imports

```
// Big
import _ from 'lodash';
_.isEmpty({});

// Little
import isEmpty from 'lodash/isEmpty';
isEmpty({})

// Big
import moment from 'moment';

// Little
import addMinutes from 'date-fns/addMinutes';
```

The Cost of Unnecessary Transpiling

```
<!-- Browsers with ES module support load this file. -->
<script type="module" src="main.mjs"></script>

<!-- Older browsers load this file (and module-supporting -->
<!-- browsers know *not* to load this file). -->
<script nomodule src="main.es5.js"></script>
```

The Cost of Unnecessary Transpiling

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

<https://philipwalton.com/articles/deploying-es2015-code-in-production-today/>

*Most of your time is spent using the app,
not waiting to load.*

Devin Villegas, Netflix senior dev ops engineer

Optimizing Responsiveness

- **Don't block the main thread!**
- Avoid memory leaks - garbage collection can pause execution
- Avoid long-running JS - chunk into smaller pieces with `requestAnimationFrame()` or `requestIdleCallback()` for scheduling
- Use up-to-date frameworks that prioritize user input (like React Fiber starting in React v16.0)

<https://medium.com/dev-channel/the-cost-of-javascript-84009f51e99e>

<https://philipwalton.com/articles/why-web-developers-need-to-care-about-interactivity/>

LATENCY & CACHING

latency

/lātənsē/

latency

/'lātənsē/

1. the state of existing but not yet being developed or manifest; concealment.

"tension, and the latency of violence, make the greatest impressions"

latency

/lātənsē/

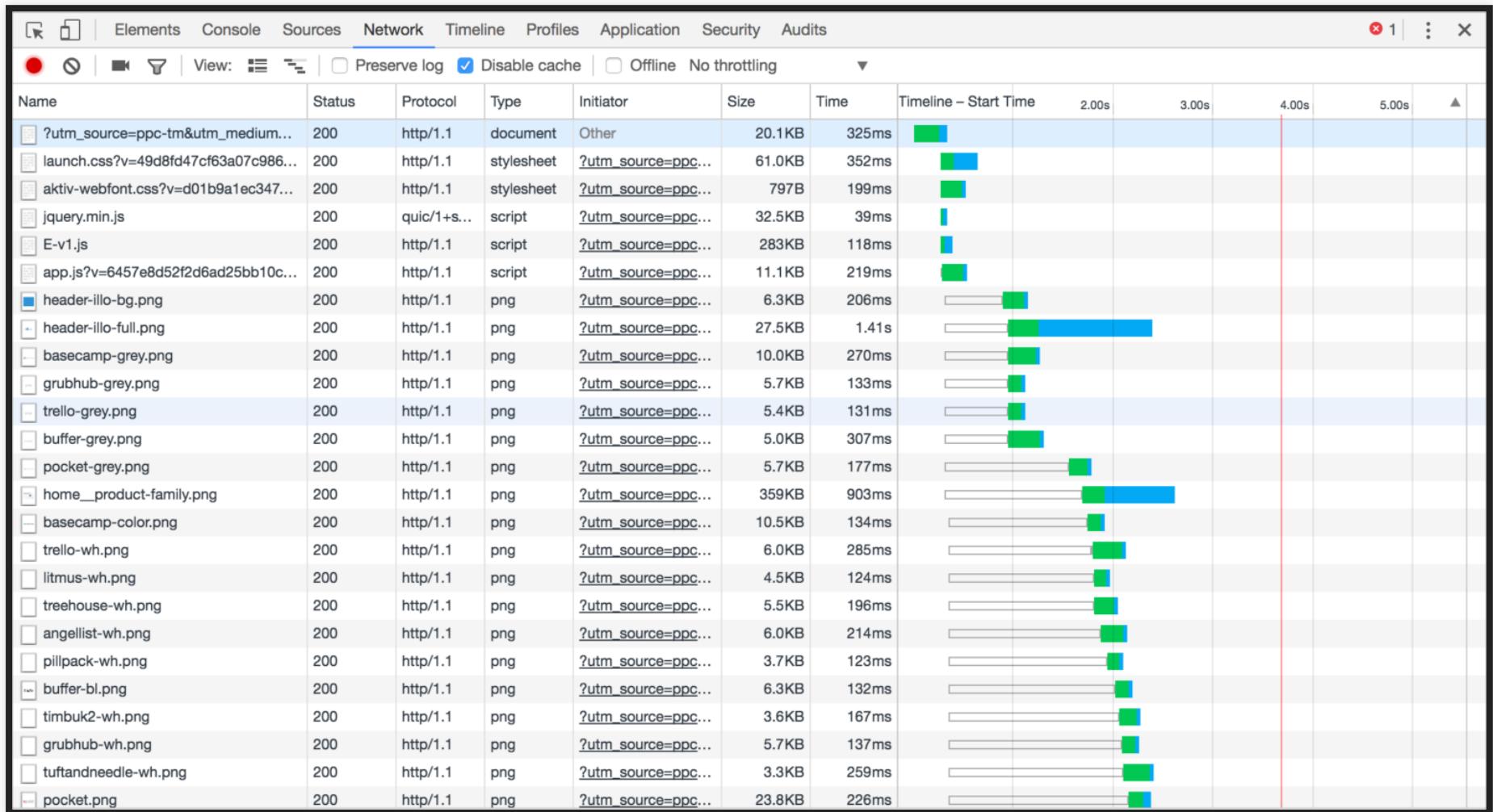
1. the state of existing but not yet being developed or manifest; concealment.

"tension, and the latency of violence, make the greatest impressions"

2. the delay before a transfer of data begins following an instruction for its transfer.

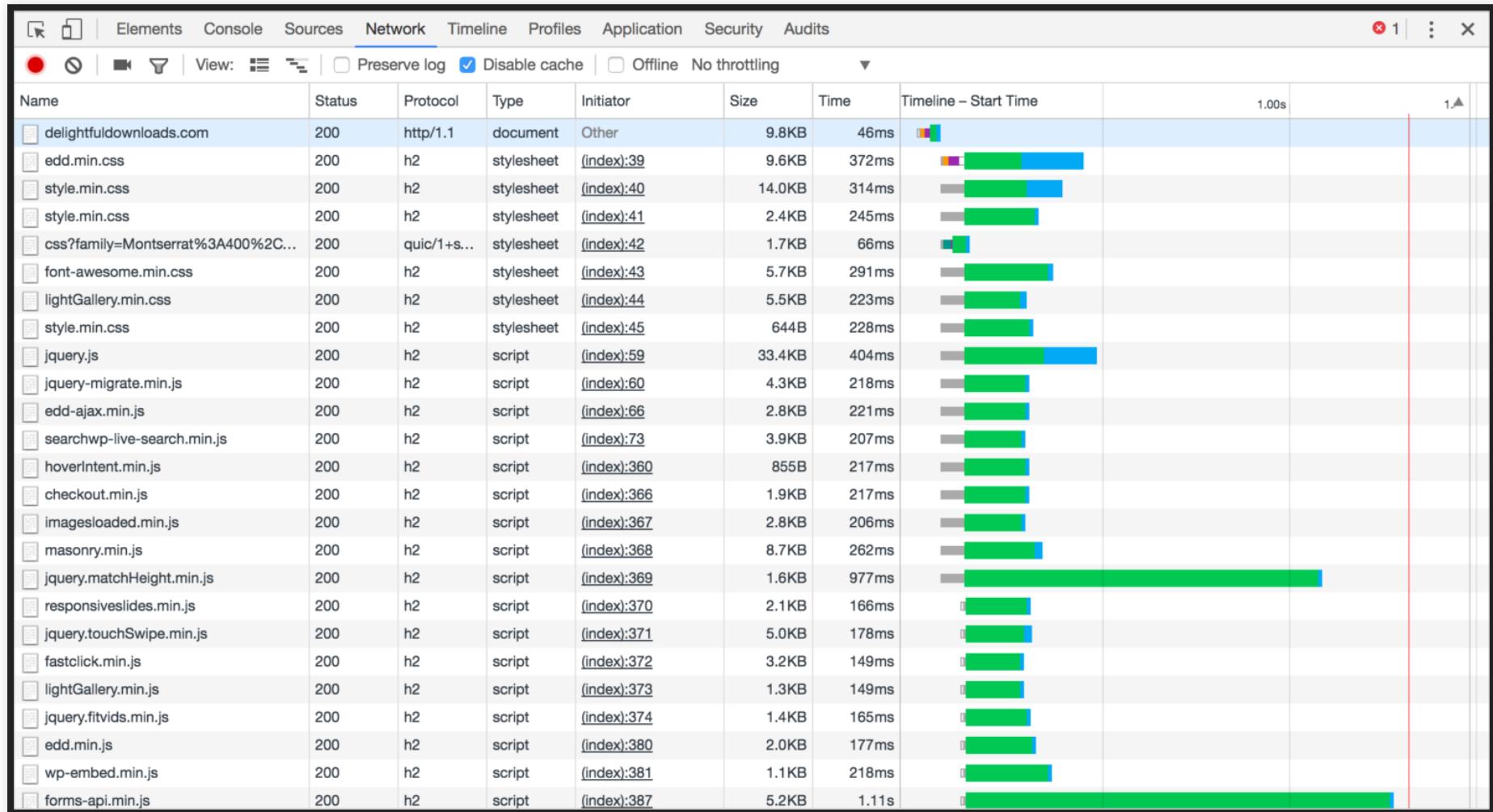
"poor performance due to network latency"

HTTP/1.1



<https://deliciousbrains.com/performance-best-practices-http2/>

HTTP/2



<https://deliciousbrains.com/performance-best-practices-http2/>



Jake Archibald

@jaffathecake

Following

HTTP vs HTTPS on public WiFi. HTTPS is important, even for static sites.

This domain is established to be used for illustrative examples in documents. You may use this domain in examples without prior coordination or asking for permission.

[More information...](#)

This domain is established to be used for illustrative examples in documents. You may use this domain in examples without prior coordination or asking for permission.

[More information...](#)



2:07 AM - 24 Sep 2018

1,001 Retweets 2,029 Likes



22

1.0K

2.0K



<https://twitter.com/jaffathecake/status/1044121129848377344>

Resource Hints

Hints to the browser that might prime the pump for resources you will need.

Preload is the only exception here, being more of an instruction than just a hint.

Preresolve DNS hostnames for assets

```
<link rel="dns-prefetch" href="https://my-site.com">
```

Begin a connection handshake in the background

```
<link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>
```

Declaratively fetch a resource without executing it

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

Prefetch a resource for a future navigation

```
<link rel="prefetch" href="/images/large.jpg">
```

Prerender a page in the background for future nav

```
<link rel="prerender" href="next.html">
```

@addyosmani

<https://twitter.com/addyosmani/status/743571393174872064?lang=en>

More Latency and Caching Strategies

- Adjust network download priority with **priority hints**
- Use appropriate **caching headers**
- Use **service workers** for precaching and offline optimization
- **Lazy-load** non-critical assets

[Preload, Prefetch And Priorities in Chrome](#) by Addy Osmani

HOUSTON'S BAGGAGE CLAIM COMPLAINTS



<http://www.nytimes.com/2012/08/19/opinion/sunday/why-waiting-in-line-is-torture.html>

Are you better off making the site load faster or ensuring that users complete their tasks?

Christine Perfetti, [The Truth About Download Time](#) 2006

THANKS!

Slides, resources, and more at bit.ly/siaspeaks