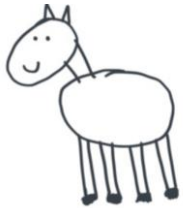
Need for speed 8

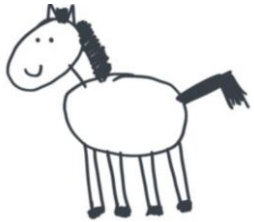
performance tuning your front end application

① DRAW 2 CIRCLES

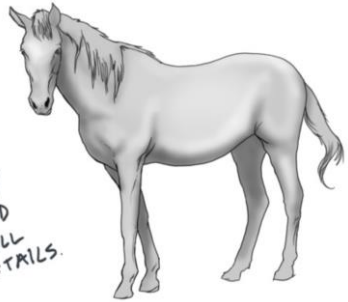
② DRAW THE LEGS



③ DRAW THE FACE



④ DRAW THE HAIR



⑤ ADD SMALL DETAILS.


When I Grow up...
I want to be like mommy!

Yaser Adel Mehraban

Software engineer, blogger, speaker, hiker

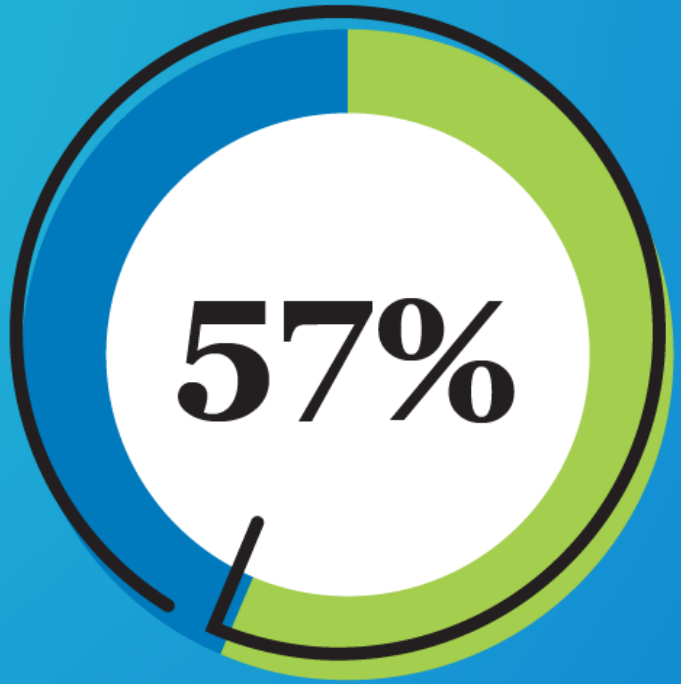
@yashints

yaser@mehraban.com.au

Why should we care?

5,200,000,000

@yashints



Lower bounce rate for mobile site visits that load < 5s

Source: DoubleClick

bit.ly/2c0Dfcv

Of shoppers who have trouble with site performance won't return

Source: Kissmetrics

bit.ly/1Bt0Ojx

@yashints


90% DISCOUNT?
YOU'RE KIDDING RIGHT?

1 second delay = 7% reduction in conversions

$3,440,547

Source: Kissmetrics

bit.ly/1Dj8Koy

@yashints

Watch out

Google Webmaster Central Blog

Official news on crawling and indexing sites for the Google index

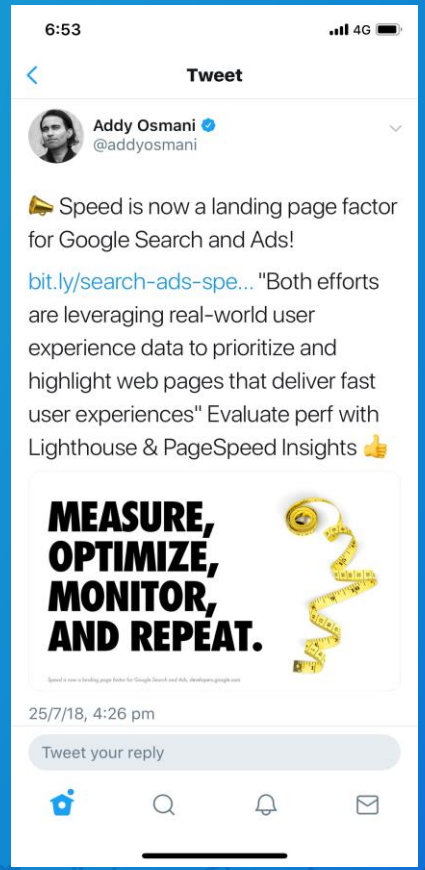
Mobile-first Indexing

Friday, November 04, 2016

Today, most people are searching on Google using a mobile device. However, our ranking systems still typically look at the desktop version of a page's content to evaluate its relevance to the user. This can cause issues when the mobile page has less content than the desktop page because our algorithms are not evaluating the actual page that is seen by a mobile searcher.

To make our results more useful, we've begun experiments to make our index mobile-first. Although our search index will continue to be a single index of websites and apps, our algorithms will eventually primarily use the mobile version of a site's content to rank pages from that site, to understand structured data, and to show snippets from those pages in our results. Of cours[illegible] ou[illegible] ind[illegible] ill be[illegible]ult from mobile documents, we're going to continue to h[illegible]d a g[illegible]eat [illegible]arch [illegible]xpe[illegible]nce [illegible]r all users, whether they come from mobile or desktop devices.

2016



@yashints

OK, but where should we start?

Choose your metrics

First Meaningful Paint

FMP, when primary content appears on the page

Hero Rendering Times

When the page's important content has finished rendering.

Source: SpeedCurve

yas.fyi/2trh1Ih

Time to Interactive

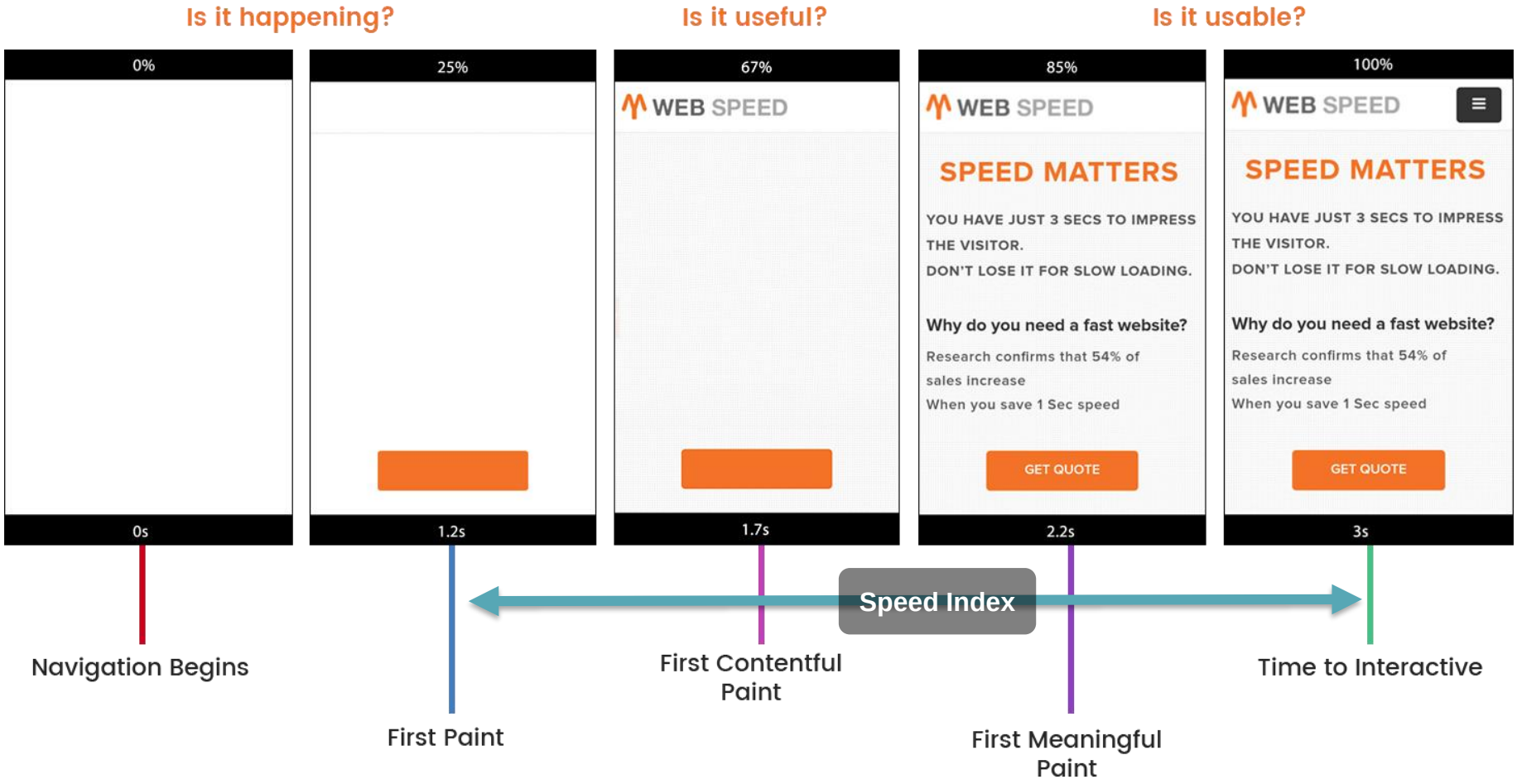
When a user can tap on UI and interact with it.

Input responsiveness

how much time it takes for an interface to respond to user's action.

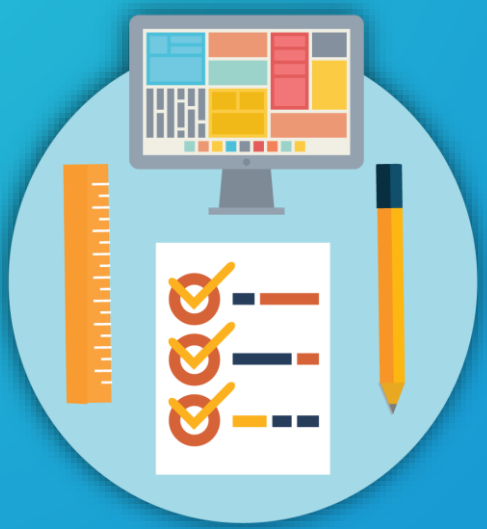
Perceptual Speed Index

measures how quickly the page contents are visually populated; the lower the score, the better.


Is it happening?
Is it useful?
Is it usable?
0%
25%
67%
85%
100%
WEB SPEED
SPEED MATTERS
YOU HAVE JUST 3 SECS TO IMPRESS THE VISITOR.
DON'T LOSE IT FOR SLOW LOADING.
Why do you need a fast website?
Research confirms that 54% of sales increase
When you save 1 Sec speed
GET QUOTE
0s
1.2s
1.7s
2.2s
3s
Speed Index
Navigation Begins
First Paint
First Contentful Paint
First Meaningful Paint
Time to Interactive

Gather data

Data sources



Field data

Lab data

Process

Field data

Lab data

Field data

Know your tools

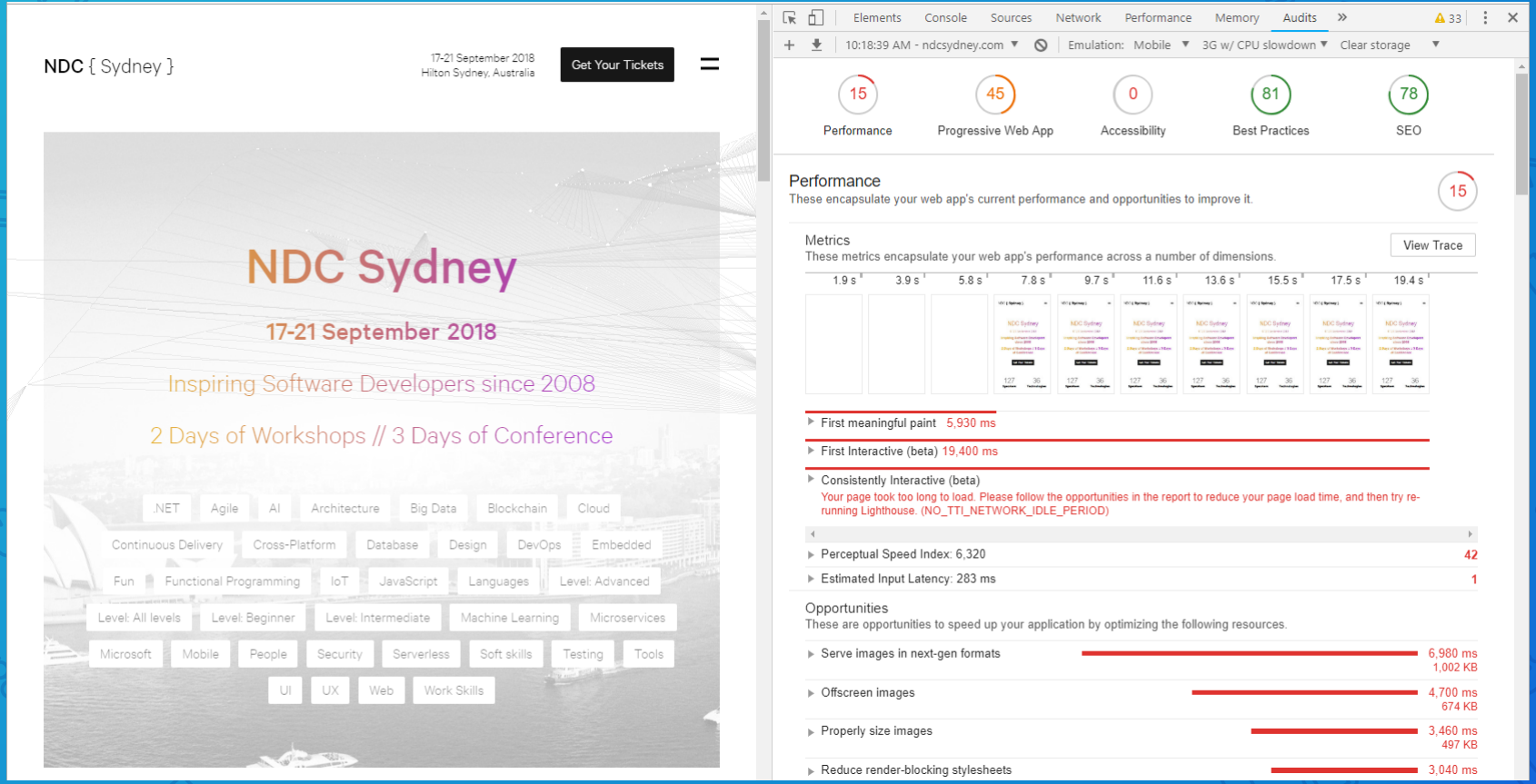
Passive monitoring tools

- Lighthouse
- WebPageTest
- Yahoo YSlow

Active monitoring tools

- SpeedCurve
- NewRelic

Lighthouse



@yashints

WebPageTest

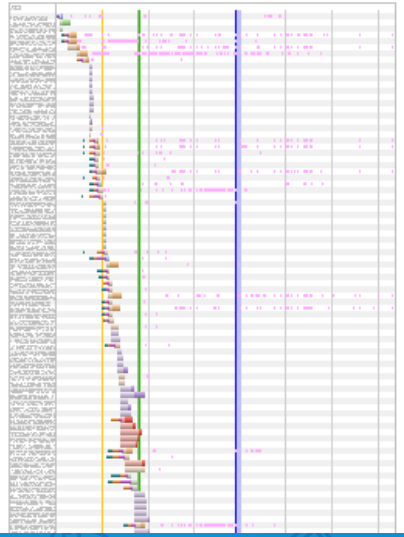
Performance Results (Median Run)

	Load Time	First Byte	Start Render	User Time	Speed Index	First Interactive (beta)	Document Complete Time	Document Complete Requests	Document Complete Bytes In	Fully Loaded Time	Fully Loaded Requests	Fully Loaded Bytes In	Fully Loaded Cost
First View (Run 3)	7.322s	0.192s	3.400s	3.421s	4515	> 14.941s	7.322s	186	2,489 KB	13.949s	401	3,416 KB	$$$$$

Plot Full Results

Test Results

Run 1:

Waterfall	Screen Shot	Video
		

@yashints

What next?

Goals

- 100 ms response
- 60 fps for animations
  - Only 16.6 ms for your code
- SpeedIndex < 1250
- TTI < 5s on 3G

Size

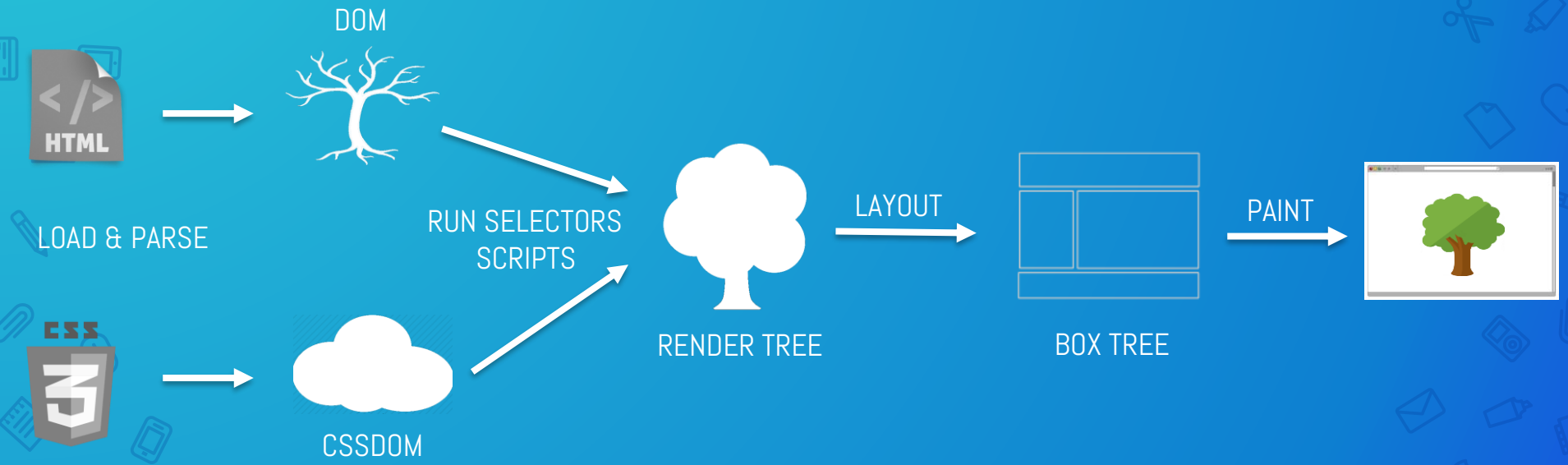
Max 170Kb gzipped

0.8-1MB decompressed

Average page size is 3Mb in 2018

Source: SpeedCurve yas.fyi/2MnCdXm

How browsers build the DOM



Critical Rendering Path

Source: google developers **yas.fyi/2zhGH0d**

@yashints

Rules to keep in mind

Stream HTML to the client

Get CSS to client fast

Did we miss something?

- Our friend JavaScript
- Images
- Web Fonts
- External Resources

JavaScript

- External
  - Requires separate network request
- Internal
  - Too much can delay initial load
- Inline
  - Don't even

DOM

```
16  var para = document.createElement("p");
17  var node = document.createTextNode("This is new.");
18  para.appendChild(node);
19
20  var element = document.getElementById("div1");
21  element.appendChild(para);
22
```

CSSDOM

```
15
16  document.getElementById('#id').style.fontSize = '20px';
17
```

Let’s optimise

PRPL

- P USH
- R ENDER
- P RE-CACHE
- L AZY-LOAD

https://developers.google.com/web/fundamentals/performance/prpl-pattern/

Styles

- <link rel="preload">
- Inline the critical CSS

- Example

```
<link rel="preload" href="style.css" as="style" onload="this.rel='stylesheet'">
```

- Use noscript

Using webpack?

preload-webpack-plugin


```
34 plugins: [
35   new HtmlWebpackPlugin(),
36   new PreloadWebpackPlugin({
37     rel: 'preload',
38     as: 'style'
39   })
40 ]
```

JavaScript

- <script defer src=...
- <script async src=...
- Right before </body>

JavaScript

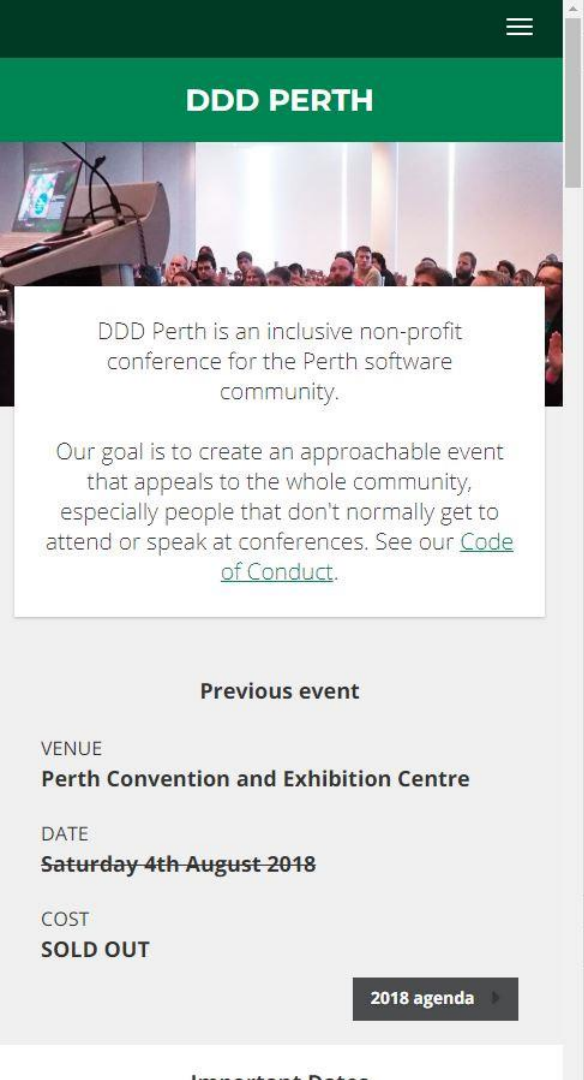
- Code splitting
  - CommonChunkPlugin
- Dynamic import/require
- Analyse your bundle (unused code/tree shaking)
  - webpack-bundle-analyzer

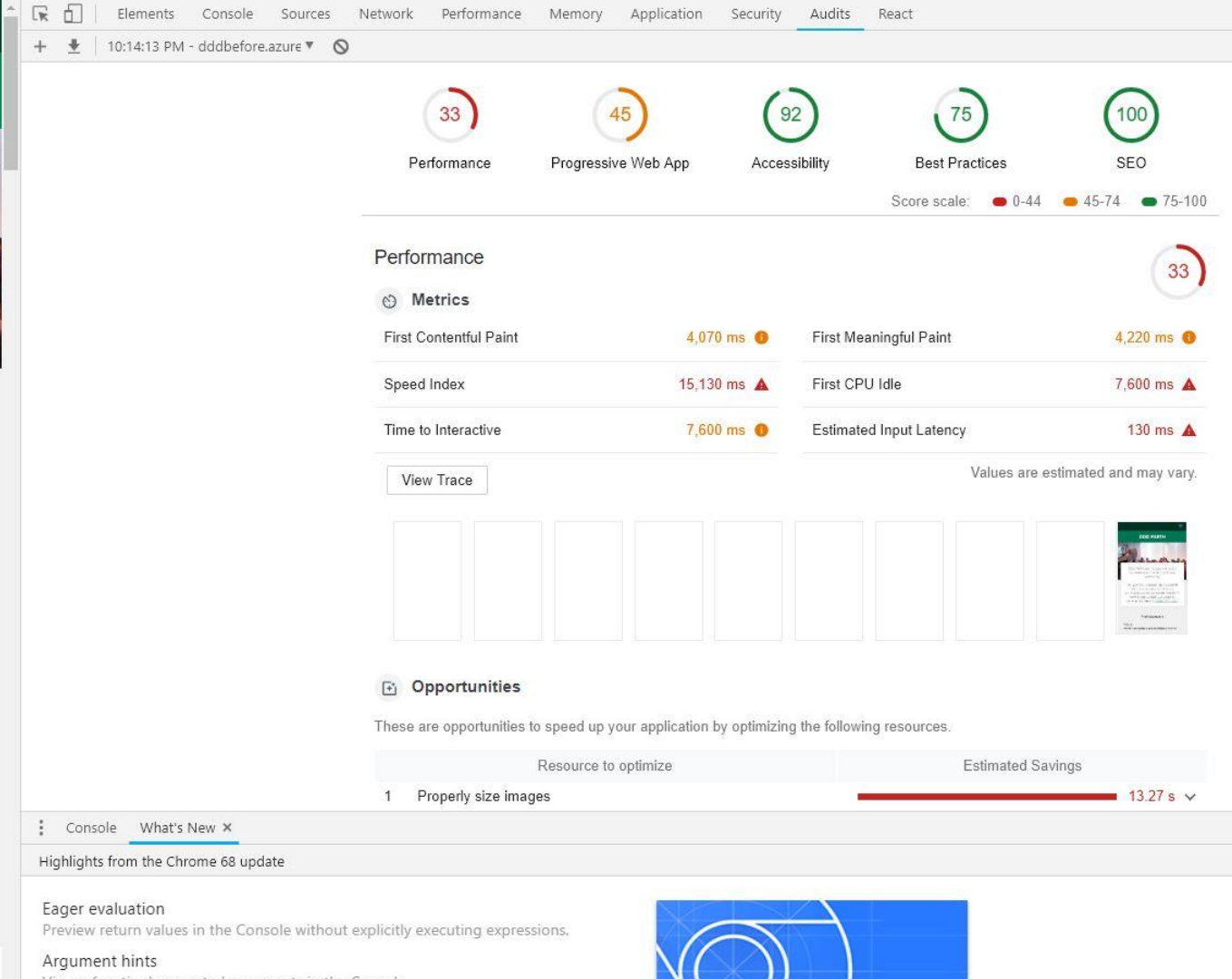
Web Fonts

- Can be cached
- Prefetch/Preload
- Do you even need it?
  - eBay's font loading strategy

Images

- > 50% of page weight
- Right dimensions
- Next-gen formats
- Compression => into build process
- Lazy load'em
- Less of'em


DDD PERTH
DDD Perth is an inclusive non-profit conference for the Perth software community.
Our goal is to create an approachable event that appeals to the whole community, especially people that don't normally get to attend or speak at conferences. See our Code of Conduct.
Previous event
VENUE
Perth Convention and Exhibition Centre
DATE
~~Saturday 4th August 2018~~
COST
SOLD OUT
2018 agenda


Elements
Console
Sources
Network
Performance
Memory
Application
Security
Audits
React
10:14:13 PM - dddbefore.azure
33
Performance
45
Progressive Web App
92
Accessibility
75
Best Practices
100
SEO
Score scale: 0-44 45-74 75-100
Performance
33
Metrics
First Contentful Paint 4,070 ms
First Meaningful Paint 4,220 ms
Speed Index 15,130 ms
First CPU Idle 7,600 ms
Time to Interactive 7,600 ms
Estimated Input Latency 130 ms
View Trace
Values are estimated and may vary.
Opportunities
These are opportunities to speed up your application by optimizing the following resources.
Resource to optimize
Estimated Savings
1 Properly size images 13.27 s
Console
What's New
Highlights from the Chrome 68 update
Eager evaluation
Preview return values in the Console without explicitly executing expressions.
Argument hints

Progressive images

srcset

```
<picture>
  <source
    type="image/webp"
    srcSet="images/worlds-300.webp 300w,
            images/worlds-600.webp 600w,
            images/worlds-800.webp 800w,
            images/worlds.webp 1000w"
  />
  <source
    srcSet="images/worlds-300.jpg 300w,
            images/worlds-600.jpg 600w,
            images/worlds-800.jpg 800w,
            images/worlds.jpg 1000w"
  />
  <img src="images/worlds-300.jpg"
    alt="The Speed of Light and the Timing of Eclipses" />
</picture>
```

DDD PERTH

DDD Perth is an inclusive non-profit conference for the Perth software community.

Our goal is to create an approachable event that appeals to the whole community, especially people that don't normally get to attend or speak at conferences. See our Code of Conduct.

Previous event

VENUE
Perth Convention and Exhibition Centre

DATE
~~**Saturday 4th August 2018**~~

COST
SOLD OUT

2018 agenda

Elements Console Sources Network Performance Memory Application Security Audits React

9:01:16 PM - dddafter.azurewe

Performance 60

Metrics

First Contentful Paint	3,600 ms	First Meaningful Paint	3,780 ms
Speed Index	3,600 ms	First CPU Idle	7,770 ms
Time to Interactive	7,770 ms	Estimated Input Latency	566 ms

View Trace

Values are estimated and may vary.

Opportunities

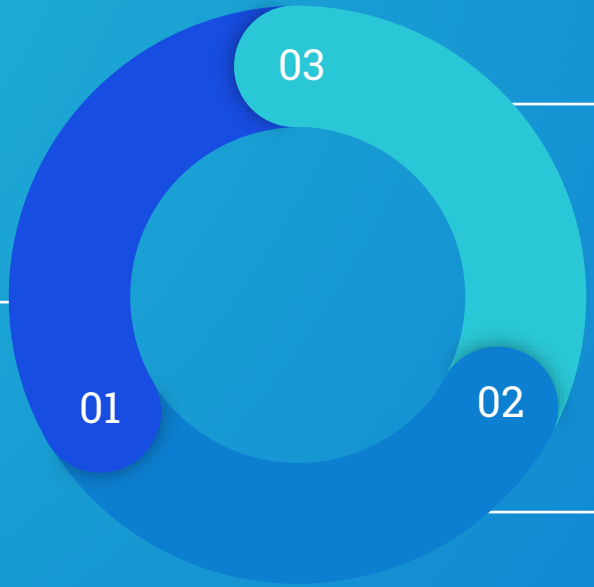
These are opportunities to speed up your application by optimizing the following resources.

	Resource to optimize	Estimated Savings
1	Serve images in next-gen formats	0.9 s
2	Eliminate render-blocking resources	0.45 s
3	Properly size images	0.3 s
4	Defer unused CSS	0.15 s

Console What's New ×

Highlights from the Chrome 68 update

Recap

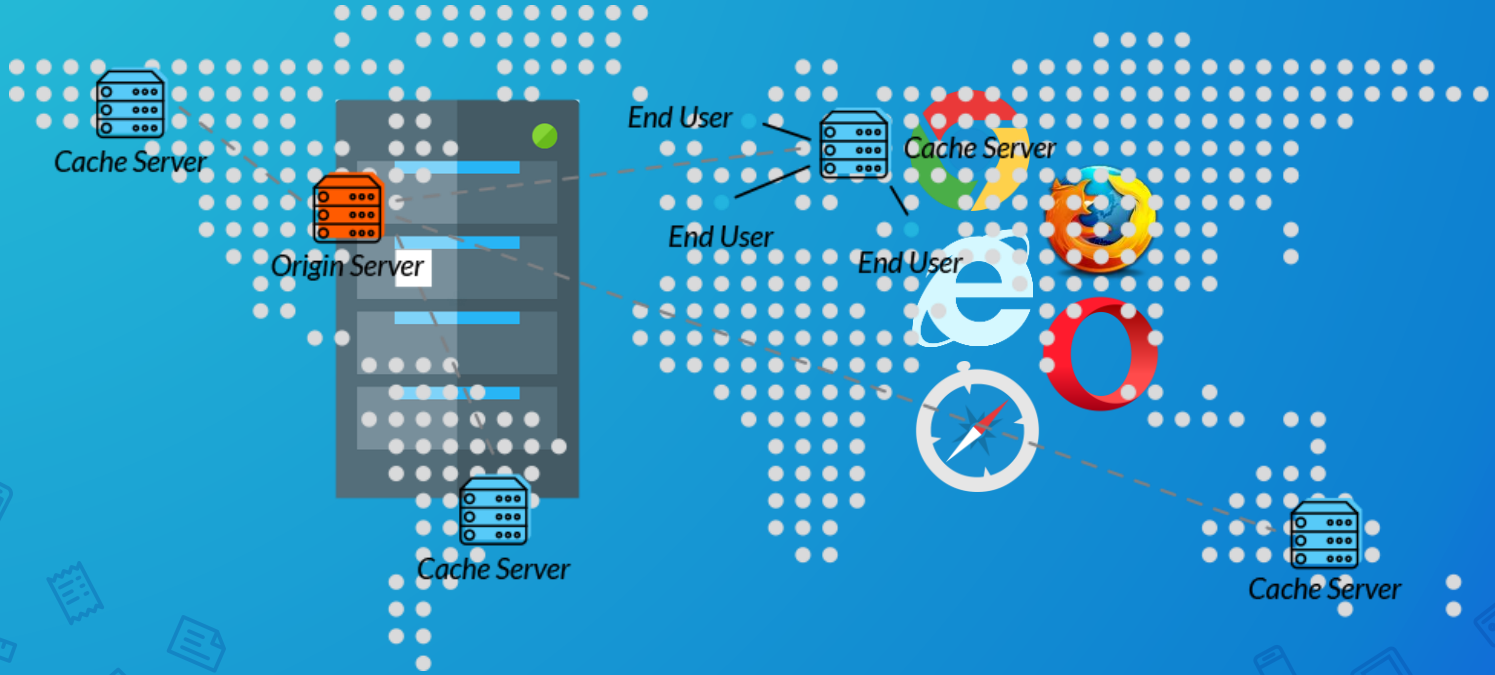


Field Data

Deploy –
Field Data

Improve –
Lab Data

Caching


Cache Server
Origin Server
Cache Server
End User
End User
Cache Server
End User
Cache Server

Demo

References

- Front-end performance checklist — yas.fyi/2uOOAoh
- Google developers – Perf Matters — yas.fyi/2JGQfS1
- High Performance Web Fonts — yas.fyi/2LBWnNl
- Google developers – Image Optimisation — yas.fyi/2LCyipJ
- Slides — yas.fyi/2LCyipJ

Thanks!

Any questions?

Find me at

@yashints

yaser@mehraban.com.au