Optimize Performance
(and not die trying)

Diego Cardozo
Sr. Web Performance Engineer @ NetSuite

Motivation

- 3 pillars in user experience
  - Design
  - Usability
  - Performance
- 80% - 90% load time comes from the front end
- Additionally...
  - Better user conversion
  - Affects Google search result rankings

Challenge

- Initial goal for this talk: teach **everything** required to build fast sites
  - 12 hour talk?
  - Very complex (a.k.a boring)
  - This group is too diverse for a technical talk
- New goal
  - Drill into the world of web performance
  - Go over the main optimization techniques
- If I was successful...
  - Continue learning by enrolling in Google's web performance course (link at the end)

Agenda / Focus

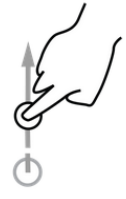
1. Measure
    - Have a clear goal
    - Measure early
    - Measure often
2. Optimize
    - Reduce amount of bytes
    - Reduce critical resources
    - Shorten the Critical Rendering Path (CRP)

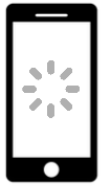
1. Measure

Measure

Have a clear goal

- Twitter - Time to first tweet
- NetSuite - Performance budget
- Google - RAIL



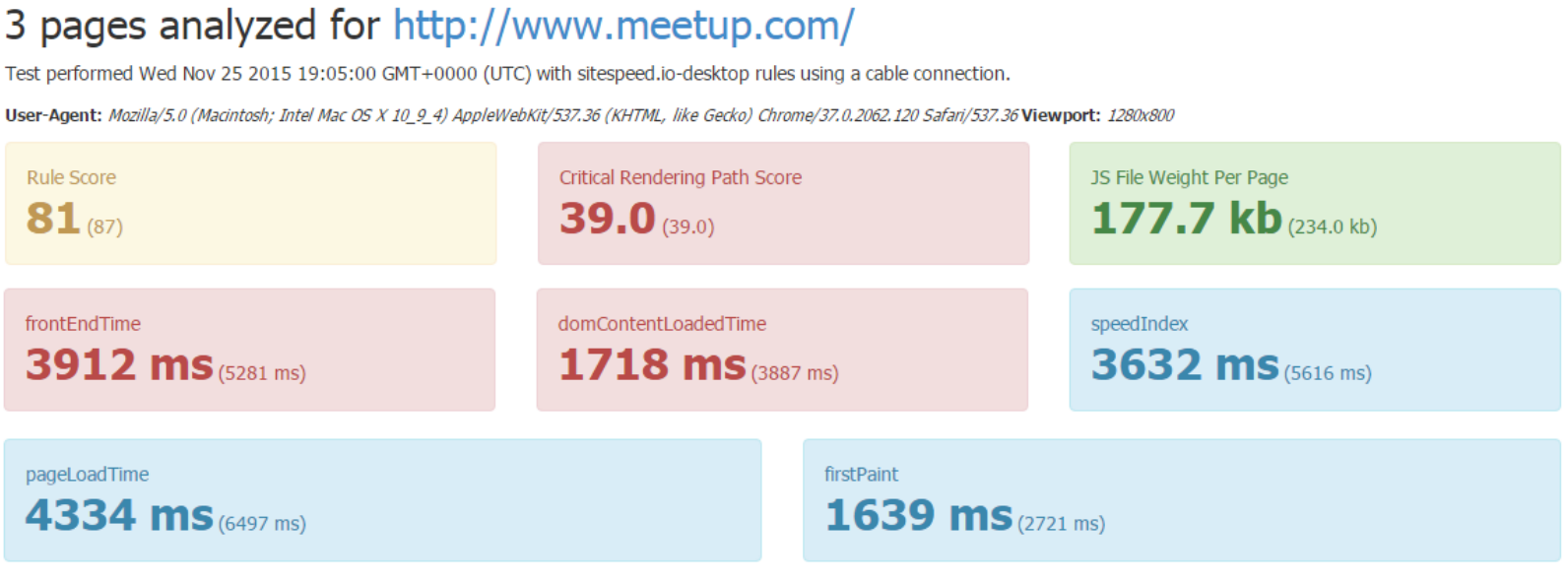


Response	Animation	Idle	Load
100ms	16ms	50ms	1s

Measure

Measure early

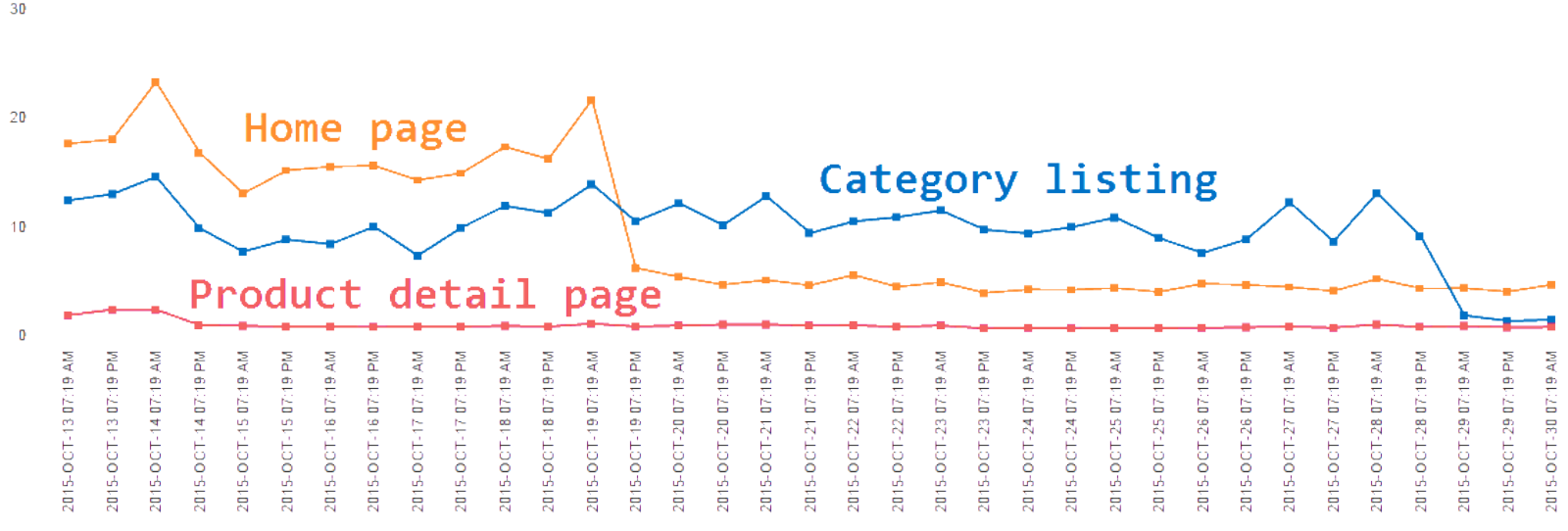
- WebPageTest
- Page Speed Insights
- sitespeed.io



Measure

Measure often

- Know how performance evolves
- Automation tools
  - Dynatrace / GTMetrix
- More in my talk from Testing Meetup



2. Optimize

Optimize

Theory

The CRP is the sequence of steps followed by the browser in order to render a page

1. Parse the HTML and build the DOM
2. Parse the CSS and build the CSSOM
3. Build the Render Tree
4. Position elements on the screen (Layout)
5. Paint the screen

Optimize

Theory

The CRP has 3 main components which can be optimized

1. Total amount of bytes to be downloaded
2. Amount of critical resources (HTML, CSS y blocking JS)
3. Amount of roundtrips needed in order to gather critical resources

Optimize

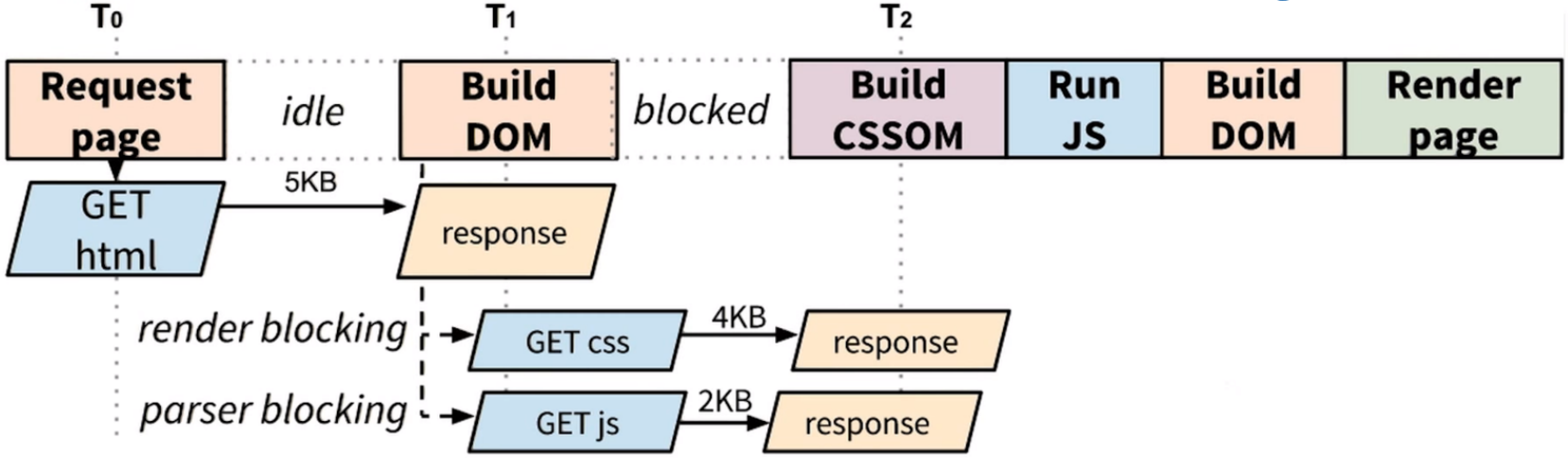
Example

```
<html>
    <head>
        <meta name="viewport" content="width=device-width,initial-scale=1.0">
        <link rel="stylesheet" href="styles.css">
    </head>
    <body>
        <p>Hola <span>meetup</span> de Front End MVD</p>
        <div><img src="foto.png"/></div>
        <script src="app.js"></script>
    </body>
</html>
```

Total KB: 11

Critical resources: 3

CRP length: 2



Optimize

Reduce total amount of bytes

- Minify, compress and cache
  - HTML
  - CSS
  - JavaScript
- Remove unused styles
  - unused-css.com
- Compress images
  - Example: compressor.io
- Compress and unify fonts

Optimize

Reduce amount of critical resources

- Concatenate JS and CSS files
- Use media queries on <link> tags to unblock rendering
- Write CSS directly on HTML (inline)
  - Leaving everything inline can be harmful
- Write JS directly on HTML (inline)
  - Still blocks rendering when executed if not marked as async

Optimize

Shorten the Critical Rendering Path (CRP)

- Delay JavaScript execution
- Add an async attribute to <script> tags so that they don't block rendering
- Code optimizations

Links

- Test performance and not die trying
- Google/Udacity course on performance optimization
- My blog post which summarizes the above course
- Article on removing unused CSS

Questions?

dcardozo@netsuite.com

slides.com/diegocard/optimize-performance