Browser Test Automation (in 2018)

@ColinBendell
CTO Office, Cloudinary

O'REILLY

High Performance Images

SHRINK, LOAD, AND DELIVER IMAGES FOR SPEED

Colin Bendell, Tim Kadlec,
Yoav Weiss, Guy Podjarny,
Nick Doyle & Mike McCall

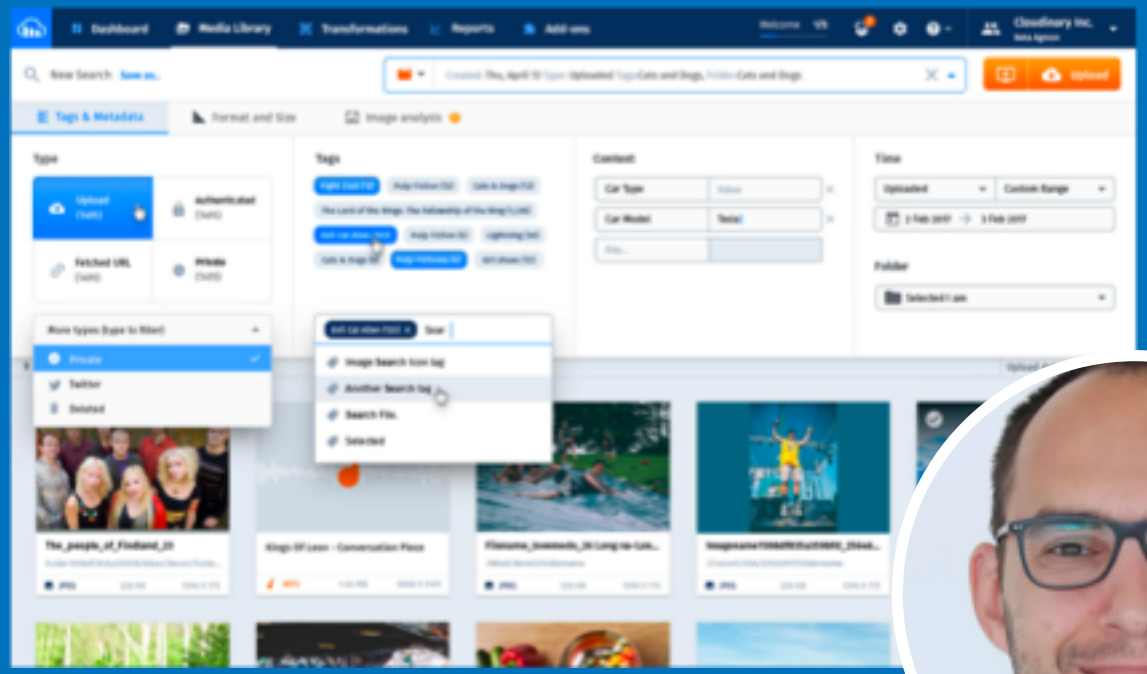
Agenda

- Test Methodologies
- WebDriver.io
- Mocha tests
- How Browsers Work & Browser APIs
- Web Platform Test
- Mobile & Device Emulation
- Accessibility
- Operational: DNS, Paralyzing, CI
- Real User Testing

9:00 AM Welcome | **10:30 AM** Break | **12:00 AM** Lunch | **2:30 PM** Break | **4:00 PM** Wrap-up & Q&A

Cloudinary's Two Problems

New Digital-Asset-Management Front-End



Automating CDN Changes

Felix Zilber

Why Do We Test?


NASA
E. ONIZUKA
NASA
NASA
NASA
MIKE SMITH
NASA
DICK
NASA
RON McNAIR
USA

"All the News That's Fit to Print"

The New York Times

Late Edition
Weather: Partly cloudy and cold today, chance of snow; chance of snow tonight. Partly cloudy, cold tomorrow. Temperatures: today 27-30, tonight 13-19; yesterday 14-23. Details, page C19.

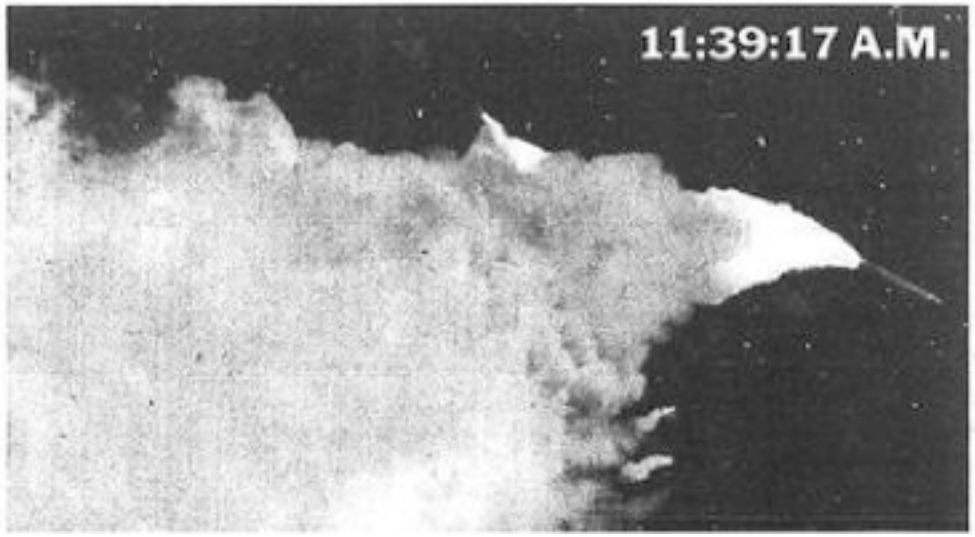
VOL. CXXXV . . . No. 46,669 NEW YORK, WEDNESDAY, JANUARY 29, 1986

Elsewhere beyond 75 miles from New York City, except on Long Island.

30 CENTS

THE SHUTTLE EXPLODES

6 IN CREW AND HIGH-SCHOOL TEACHER ARE KILLED 74 SECONDS AFTER LIFTOFF



Thousands Watch A Rain of Debris

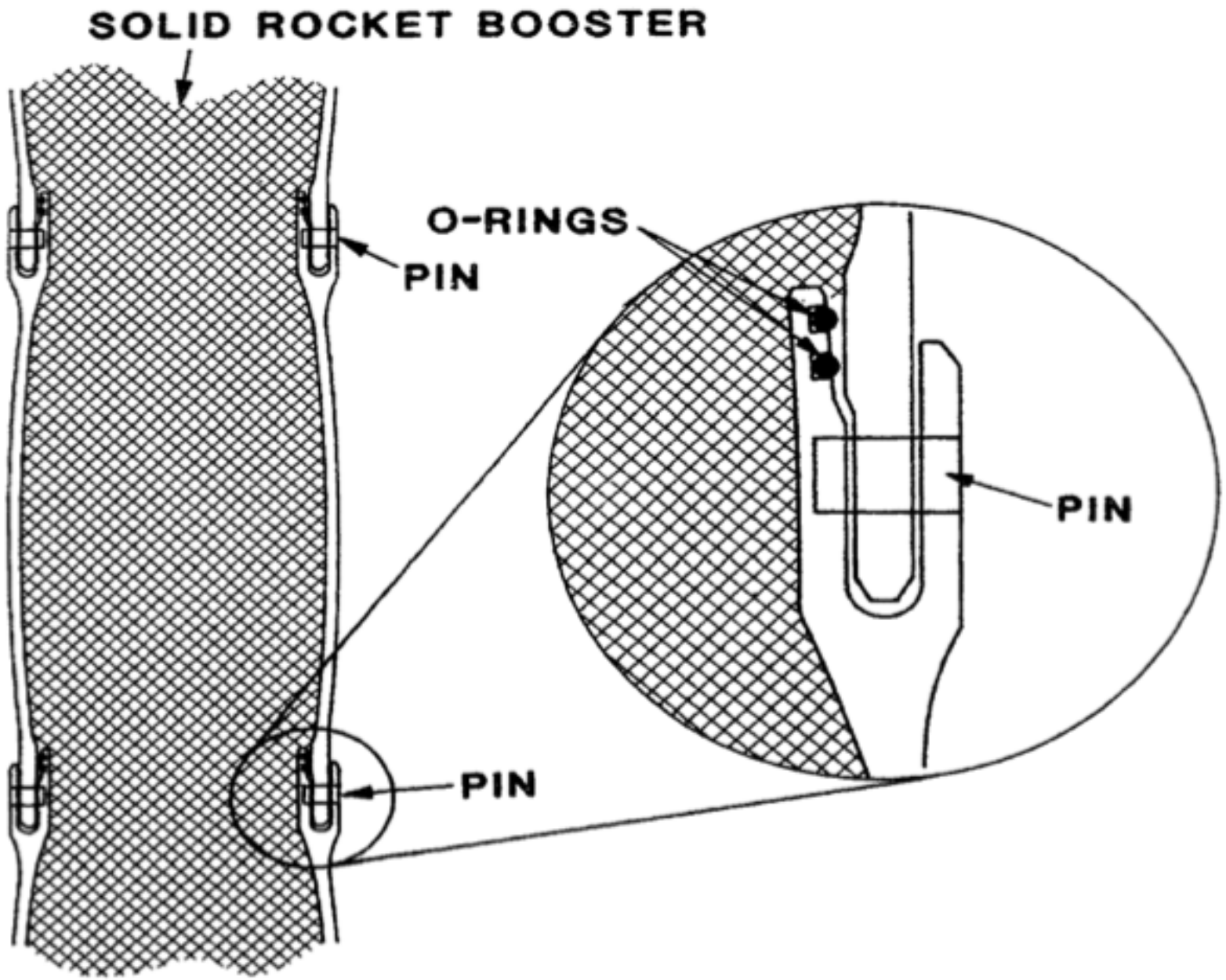
By WILLIAM J. BROAD
Special to The New York Times

CAPE CANAVERAL, Fla., Jan. 28 — The space shuttle Challenger exploded in a ball of fire shortly after it left the launching pad today, and all seven astronauts on board were lost.

The worst accident in the history of the American space program, it was witnessed by thousands of spectators who watched in wonder, then horror, as the ship blew apart high in the air.

Flaming debris rained down on the Atlantic Ocean for an hour after the explosion, which occurred just after 11:39 A.M. It kept rescue teams from reaching the area where the craft would have fallen into the sea, about 18 miles offshore.

It seemed impossible that anyone could have


SOLID ROCKET BOOSTER
O-RINGS
PIN
PIN
PIN

Testing: Prepare for the Unexpected

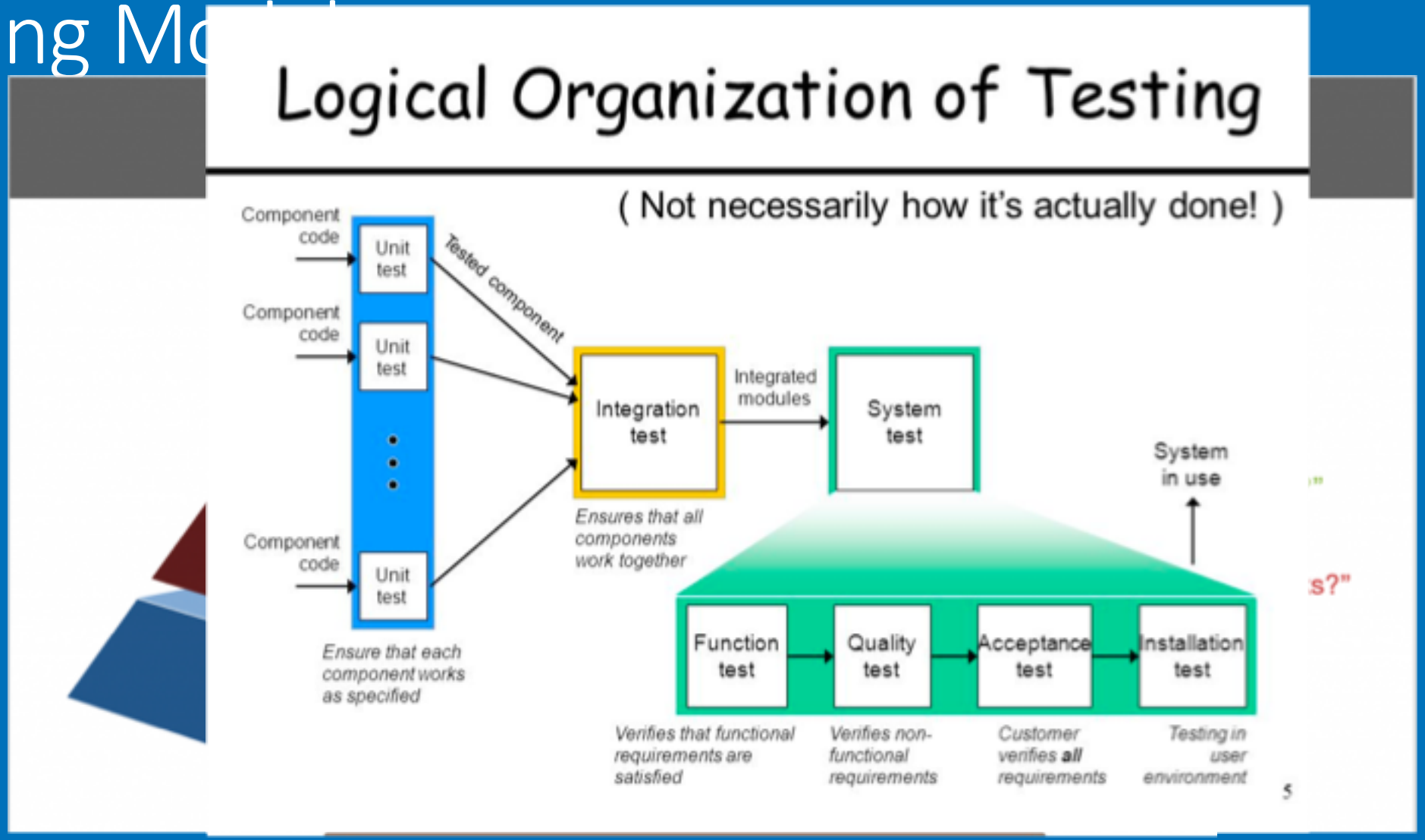
“I just want to buy my Mom’s birthday present.”

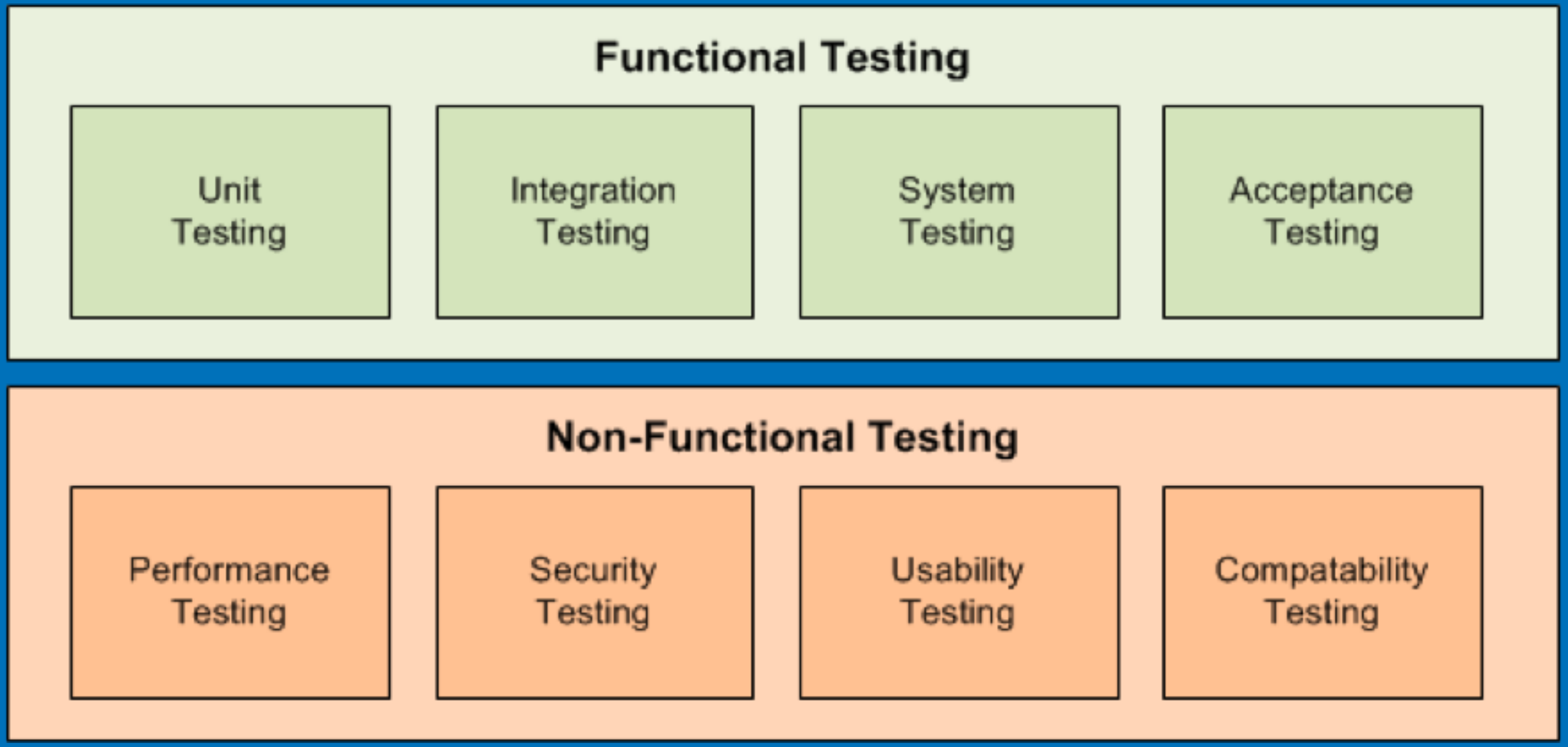
“I just want to know if it will rain today.”

“I just want to share pictures of my vacation”


Testing === Meeting User Expectations

Testing Mo


Logical Organization of Testing
(Not necessarily how it's actually done!)
Component code
Unit test
Component code
Unit test
Component code
Unit test
Tested component
Ensure that each component works as specified
Integration test
Ensures that all components work together
Integrated modules
System test
System in use
Function test
Quality test
Acceptance test
Installation test
Verifies that functional requirements are satisfied
Verifies non-functional requirements
Customer verifies all requirements
Testing in user environment
5



https://stackoverflow.com/questions/2741832/unit-tests-vs-functional-tests

Functional v. Non-Functional

Unit Testing

- Smallest testable component
- Usually simple inputs and outputs
- Moch objects v. skeleton dev services

Unit Testing

```
const expect = require("chai").expect;
const converter = require("../app/converter");

describe("Color Code Converter", function() {
  describe("RGB to Hex conversion", function() {
    it("converts the basic colors", function() {
      var redHex = converter.rgbToHex(255, 0, 0);
      var greenHex = converter.rgbToHex(0, 255, 0);
      var blueHex = converter.rgbToHex(0, 0, 255);
      expect(redHex).to.equal("ff0000");
      expect(greenHex).to.equal("00ff00");
      expect(blueHex).to.equal("0000ff");
    });
  });
});
```

Unit Testing

- Blurred line to integration tests
- Encourages small component logic v. monoliths
- Large volume of Tests

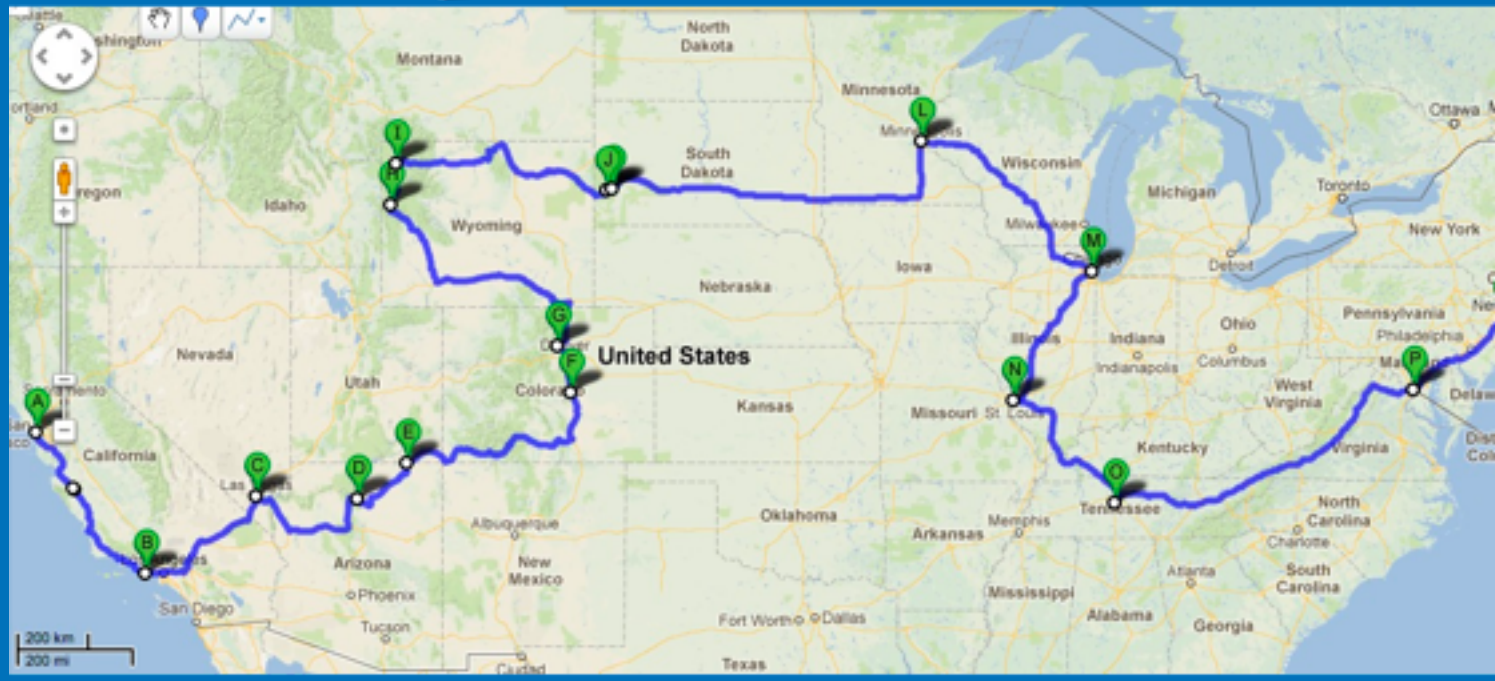
Integration Tests

Integration Tests

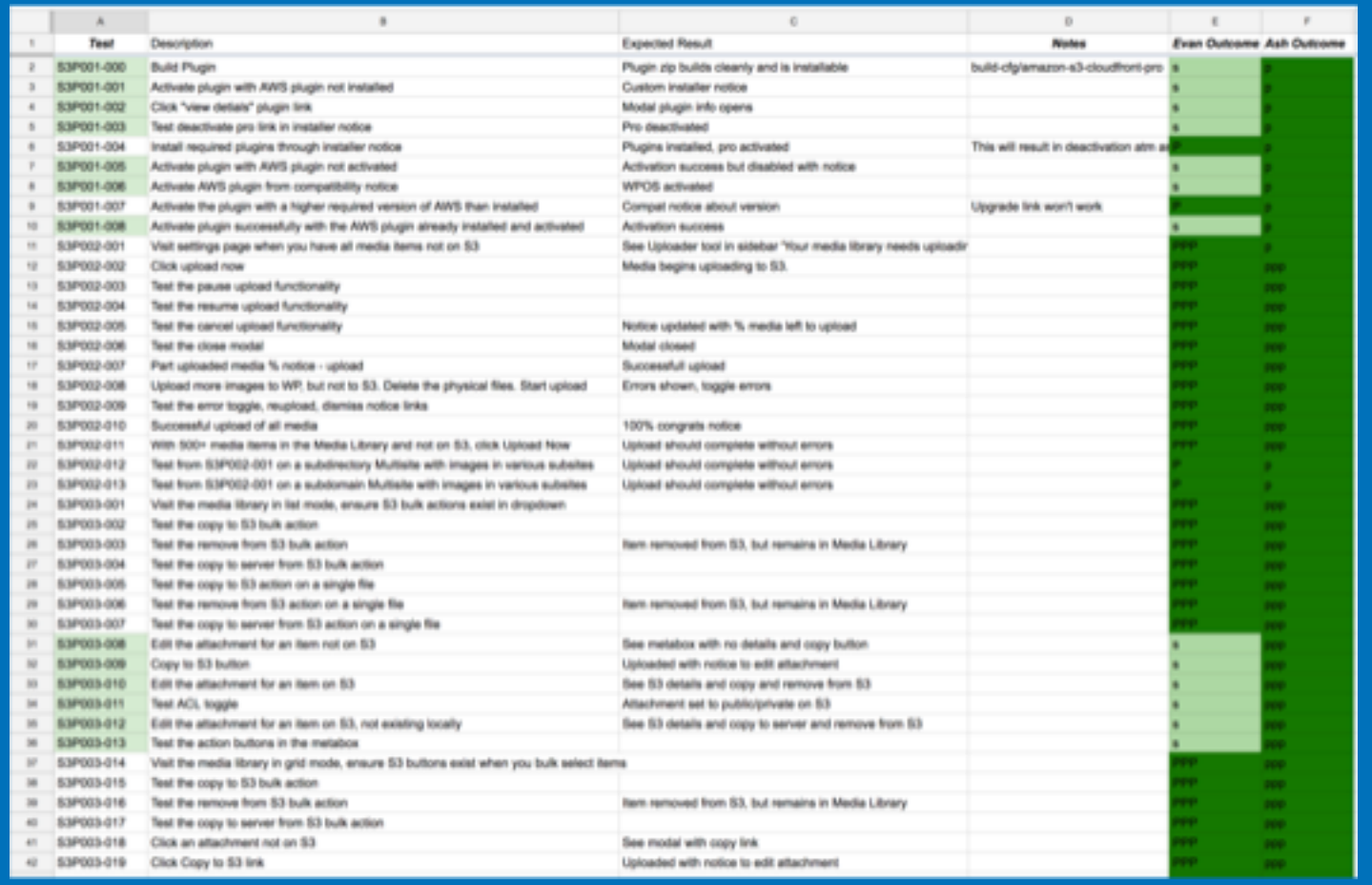
```
describe("Color Code Converter API", function() {
  describe("RGB to Hex conversion", function() {
    let url =
"http://localhost:3000/rgbToHex?red=255&green=255&blue=255";
    it("returns status 200", function() {});
    it("returns the color in hex", function() {});
  });

  describe("Hex to RGB conversion", function() {
    let url = "http://localhost:3000/hexToRgb?hex=00ff00";
    it("returns status 200", function() {});
    it("returns the color in RGB", function() {});
  });
});
```

System Tests

- Does it match design?
- Meet expectations of design
- Often focused on workflows
- Volume, load, stress, security
- Usability, Accessibility



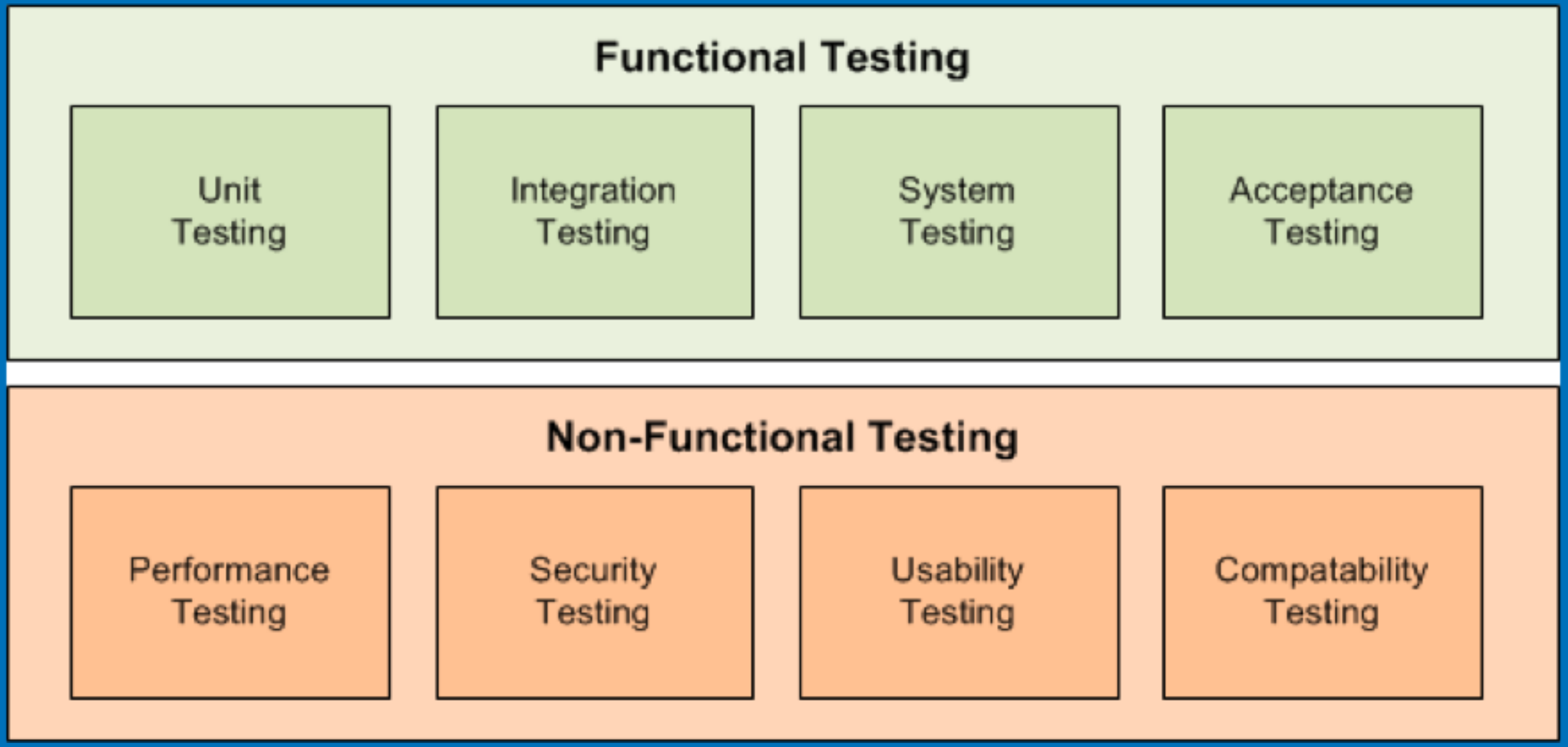
System Tests



	A	B	C	D	E	F
1	**Test**	Description	Expected Result	**Notes**	***Evan Outcome***	***Ash Outcome***
2	S3P001-000	Build Plugin	Plugin zip builds cleanly and is installable	build-cfg/amazon-s3-cloudfront-pro	s	p
3	S3P001-001	Activate plugin with AWS plugin not installed	Custom installer notice		s	p
4	S3P001-002	Click "view details" plugin link	Modal plugin info opens		s	p
5	S3P001-003	Test deactivate pro link in installer notice	Pro deactivated		s	p
6	S3P001-004	Install required plugins through installer notice	Plugins installed, pro activated	This will result in deactivation atm s	p	p
7	S3P001-005	Activate plugin with AWS plugin not activated	Activation success but disabled with notice		s	p
8	S3P001-006	Activate AWS plugin from compatibility notice	WPOS activated		s	p
9	S3P001-007	Activate the plugin with a higher required version of AWS than installed	Compat notice about version	Upgrade link won't work	p	p
10	S3P001-008	Activate plugin successfully with the AWS plugin already installed and activated	Activation success		s	p
11	S3P002-001	Visit settings page when you have all media items not on S3	See Uploader tool in sidebar "Your media library needs uploadir		ppp	p
12	S3P002-002	Click upload now	Media begins uploading to S3.		ppp	ppp
13	S3P002-003	Test the pause upload functionality			ppp	ppp
14	S3P002-004	Test the resume upload functionality			ppp	ppp
15	S3P002-005	Test the cancel upload functionality	Notice updated with % media left to upload		ppp	ppp
16	S3P002-006	Test the close modal	Modal closed		ppp	ppp
17	S3P002-007	Part uploaded media % notice - upload	Successfull upload		ppp	ppp
18	S3P002-008	Upload more images to WP, but not to S3. Delete the physical files. Start upload	Errors shown, toggle errors		ppp	ppp
19	S3P002-009	Test the error toggle, reupload, dismiss notice links			ppp	ppp
20	S3P002-010	Successful upload of all media	100% congrats notice		ppp	ppp
21	S3P002-011	With 500+ media items in the Media Library and not on S3, click Upload Now	Upload should complete without errors		ppp	ppp
22	S3P002-012	Test from S3P002-001 on a subdirectory Multisite with images in various subsites	Upload should complete without errors		p	p
23	S3P002-013	Test from S3P002-001 on a subdomain Multisite with images in various subsites	Upload should complete without errors		p	p
24	S3P003-001	Visit the media library in list mode, ensure S3 bulk actions exist in dropdown			ppp	ppp
25	S3P003-002	Test the copy to S3 bulk action			ppp	ppp
26	S3P003-003	Test the remove from S3 bulk action	Item removed from S3, but remains in Media Library		ppp	ppp
27	S3P003-004	Test the copy to server from S3 bulk action			ppp	ppp
28	S3P003-005	Test the copy to S3 action on a single file			ppp	ppp
29	S3P003-006	Test the remove from S3 action on a single file	Item removed from S3, but remains in Media Library		ppp	ppp
30	S3P003-007	Test the copy to server from S3 action on a single file			ppp	ppp
31	S3P003-008	Edit the attachment for an item not on S3	See metabox with no details and copy button		s	ppp
32	S3P003-009	Copy to S3 button	Uploaded with notice to edit attachment		s	ppp
33	S3P003-010	Edit the attachment for an item on S3	See S3 details and copy and remove from S3		s	ppp
34	S3P003-011	Test ACL toggle	Attachment set to public/private on S3		s	ppp
35	S3P003-012	Edit the attachment for an item on S3, not existing locally	See S3 details and copy to server and remove from S3		s	ppp
36	S3P003-013	Test the action buttons in the metabox			s	ppp
37	S3P003-014	Visit the media library in grid mode, ensure S3 buttons exist when you bulk select items			ppp	ppp
38	S3P003-015	Test the copy to S3 bulk action			ppp	ppp
39	S3P003-016	Test the remove from S3 bulk action	Item removed from S3, but remains in Media Library		ppp	ppp
40	S3P003-017	Test the copy to server from S3 bulk action			ppp	ppp
41	S3P003-018	Click an attachment not on S3	See modal with copy link		ppp	ppp
42	S3P003-019	Click Copy to S3 link	Uploaded with notice to edit attachment		ppp	ppp

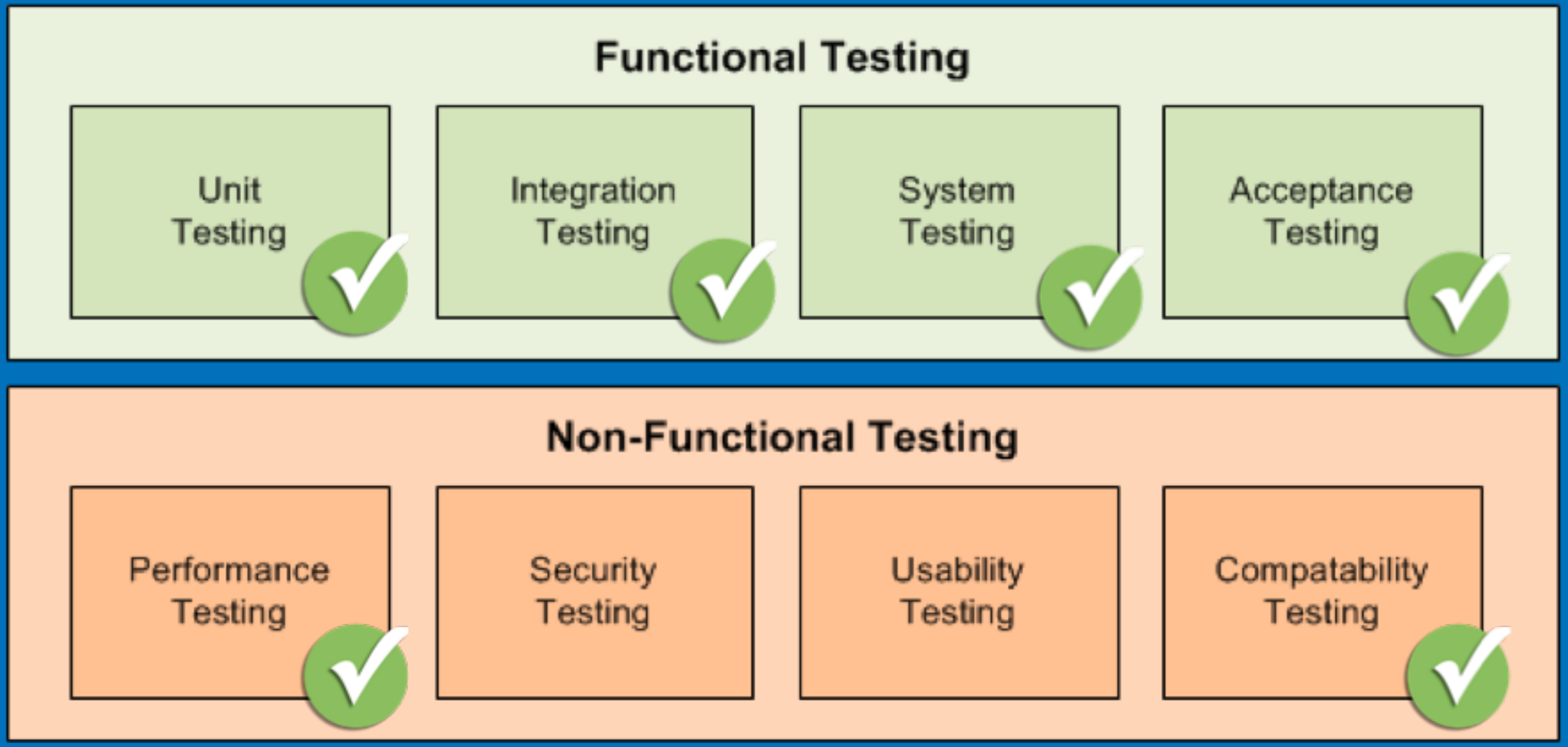
Acceptance Tests

- Meet business objectives
- End-to-End
- Can the end user accomplish what we wanted it to do?
- Alpha/Beta Feedback cycle


Functional Testing
Unit Testing
Integration Testing
System Testing
Acceptance Testing
Non-Functional Testing
Performance Testing
Security Testing
Usability Testing
Compatability Testing

“We develop in the Zoo,
But release in the Jungle”

-Melanie Cey

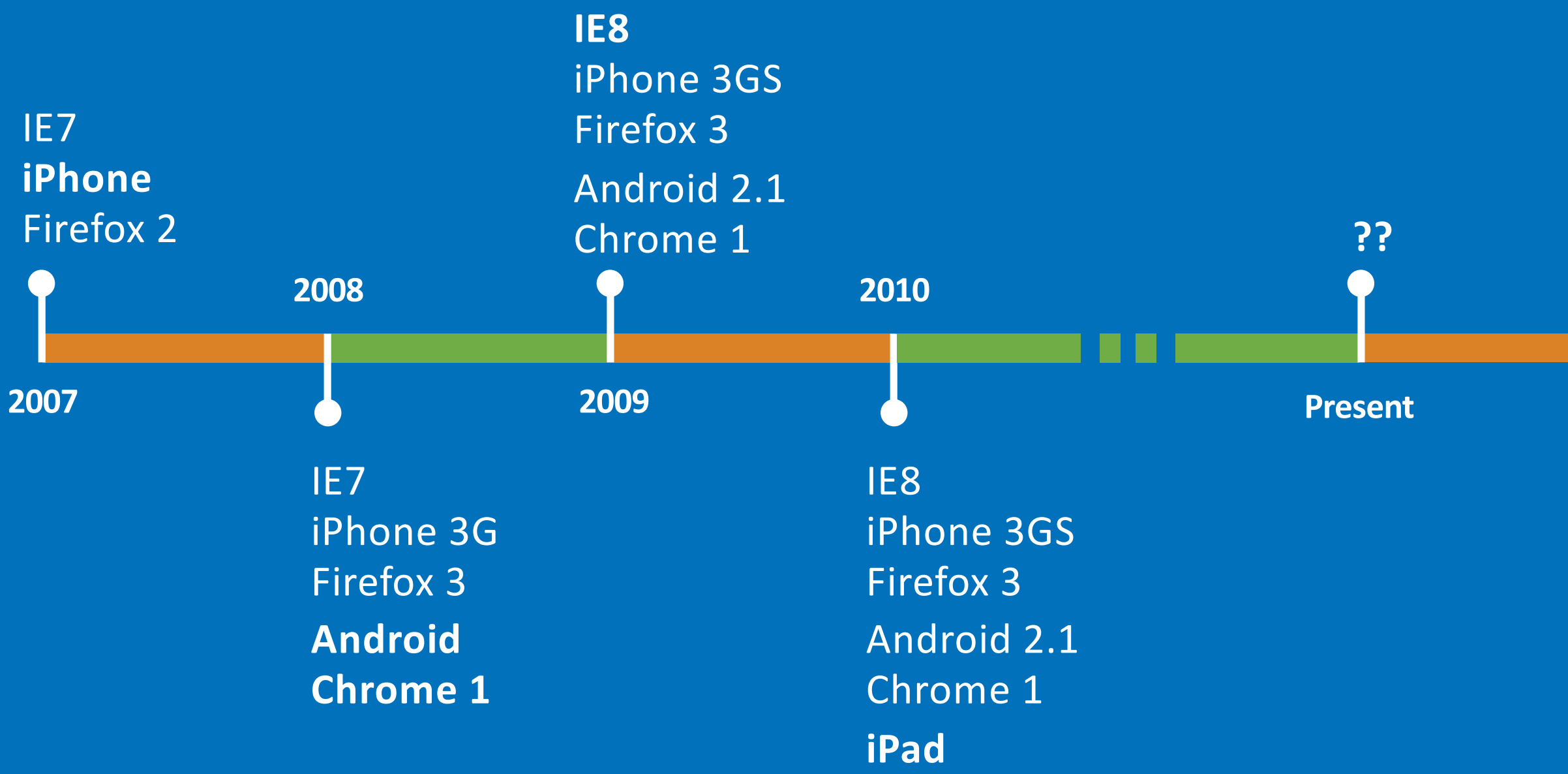


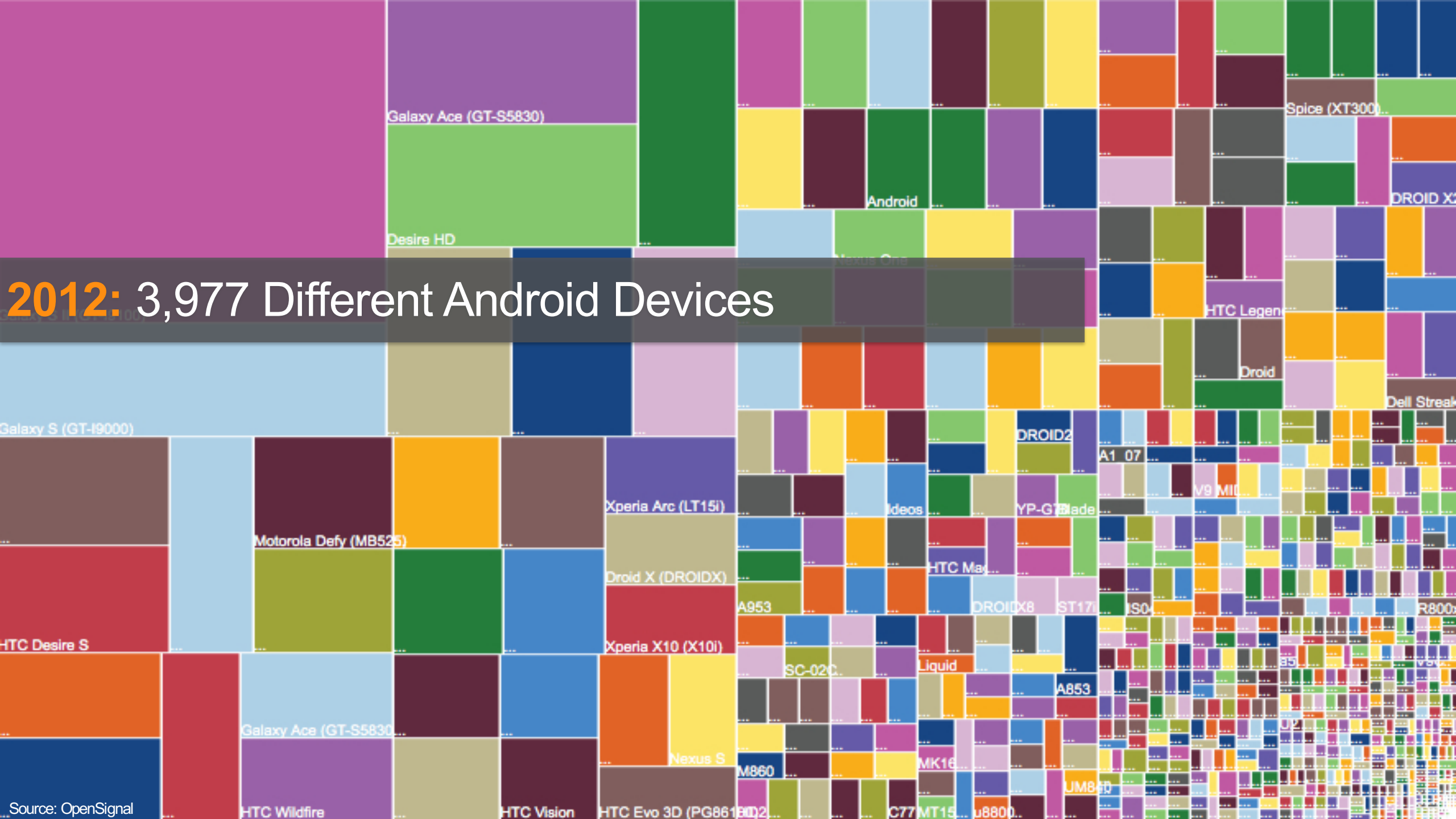
https://stackoverflow.com/questions/2741832/unit-tests-vs-functional-tests

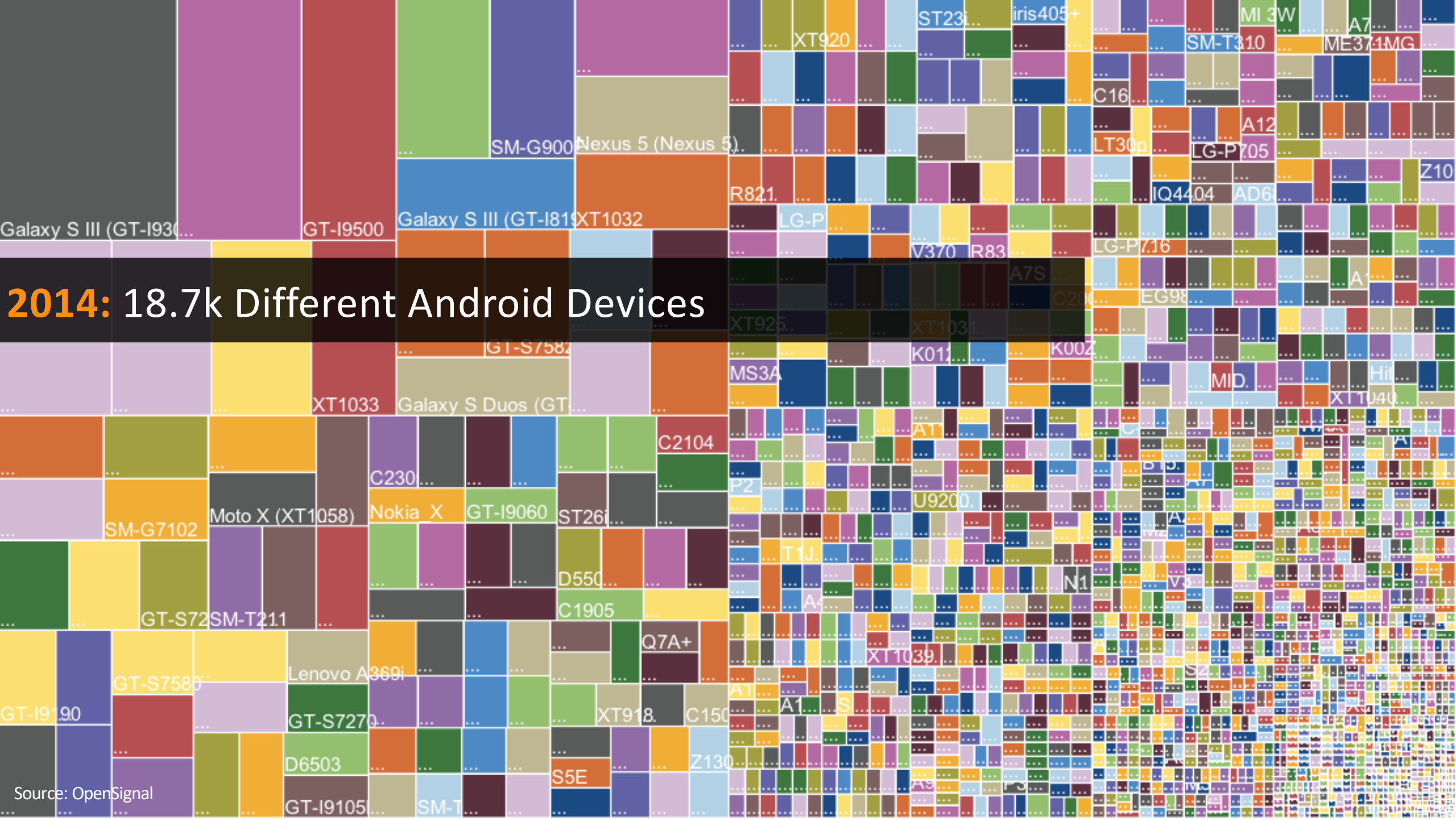
Why do we need browser testing

- Functional User experience
  - Workflows
- Design & Layout
  - Preserve Presentation
  - Above the fold, responsive web
  - Mobile experiences v. Watch v. ??
- Accessibility
  - Screen readers
  - audits
- Performance measurement
  - Performance budgets
  - Request flows

Browser Eco System

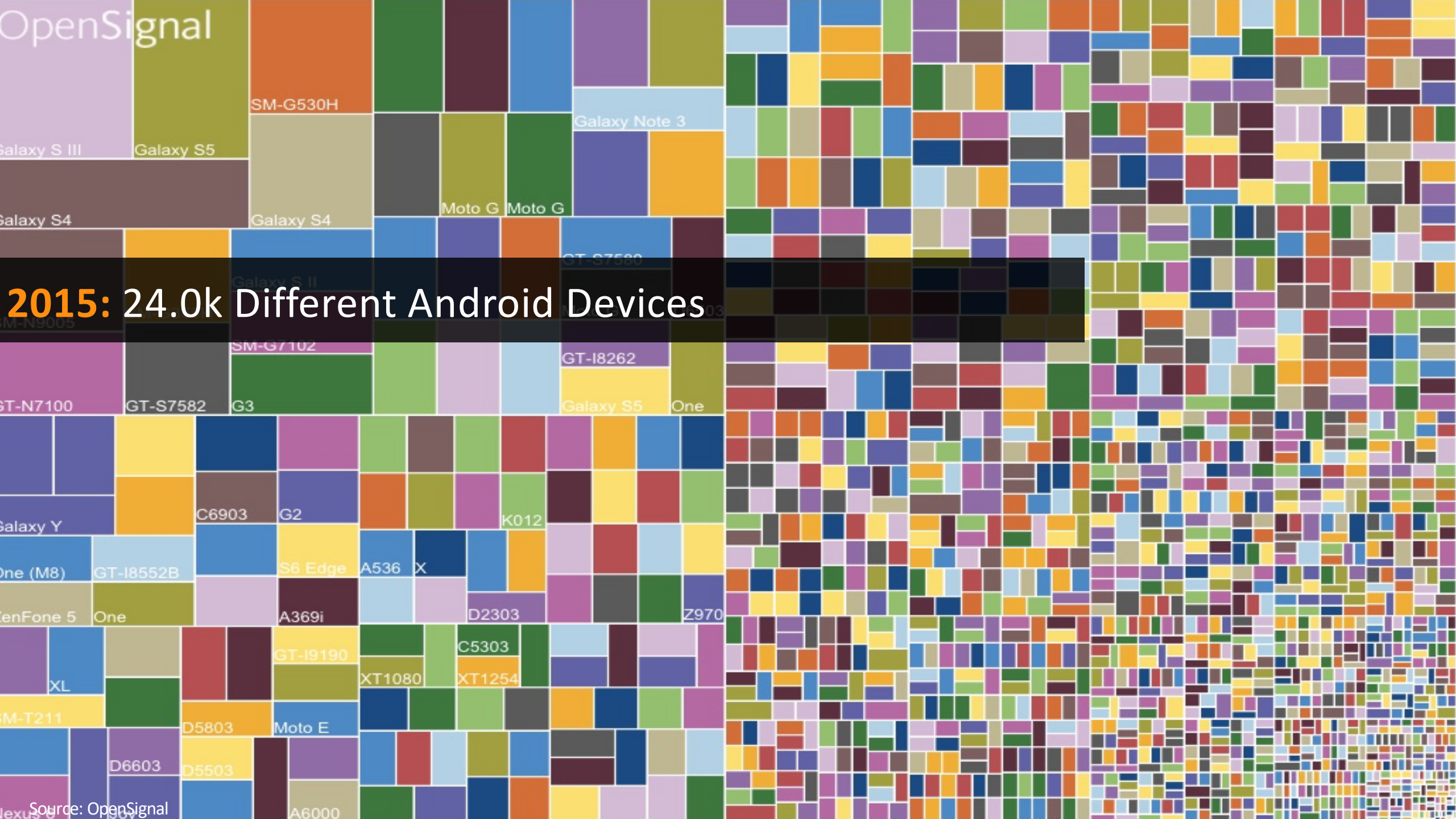

IE7
iPhone
Firefox 2
2007
2008
IE7
iPhone 3G
Firefox 3
Android
Chrome 1
IE8
iPhone 3GS
Firefox 3
Android 2.1
Chrome 1
2009
2010
IE8
iPhone 3GS
Firefox 3
Android 2.1
Chrome 1
iPad
??
Present


2012: 3,977 Different Android Devices
Galaxy Ace (GT-S5830)
Desire HD
Android
Spice (XT300)
DROID X
HTC Legen
Droid
Dell Streak
Galaxy S (GT-I9000)
DROID2
A1_07
Xperia Arc (LT15i)
Motorola Defy (MB525)
Ideos
YP-G
Blade
Droid X (DROIDX)
HTC Ma
A953
DROIDX8
ST17i
HTC Desire S
Xperia X10 (X10i)
SC-02C
Liquid
A853
Galaxy Ace (GT-S5830
Nexus S
M860
MK16
HTC Wildfire
HTC Vision
HTC Evo 3D (PG86100)
C77
MT15
u8800
Source: OpenSignal



2014: 18.7k Different Android Devices

Source: OpenSignal



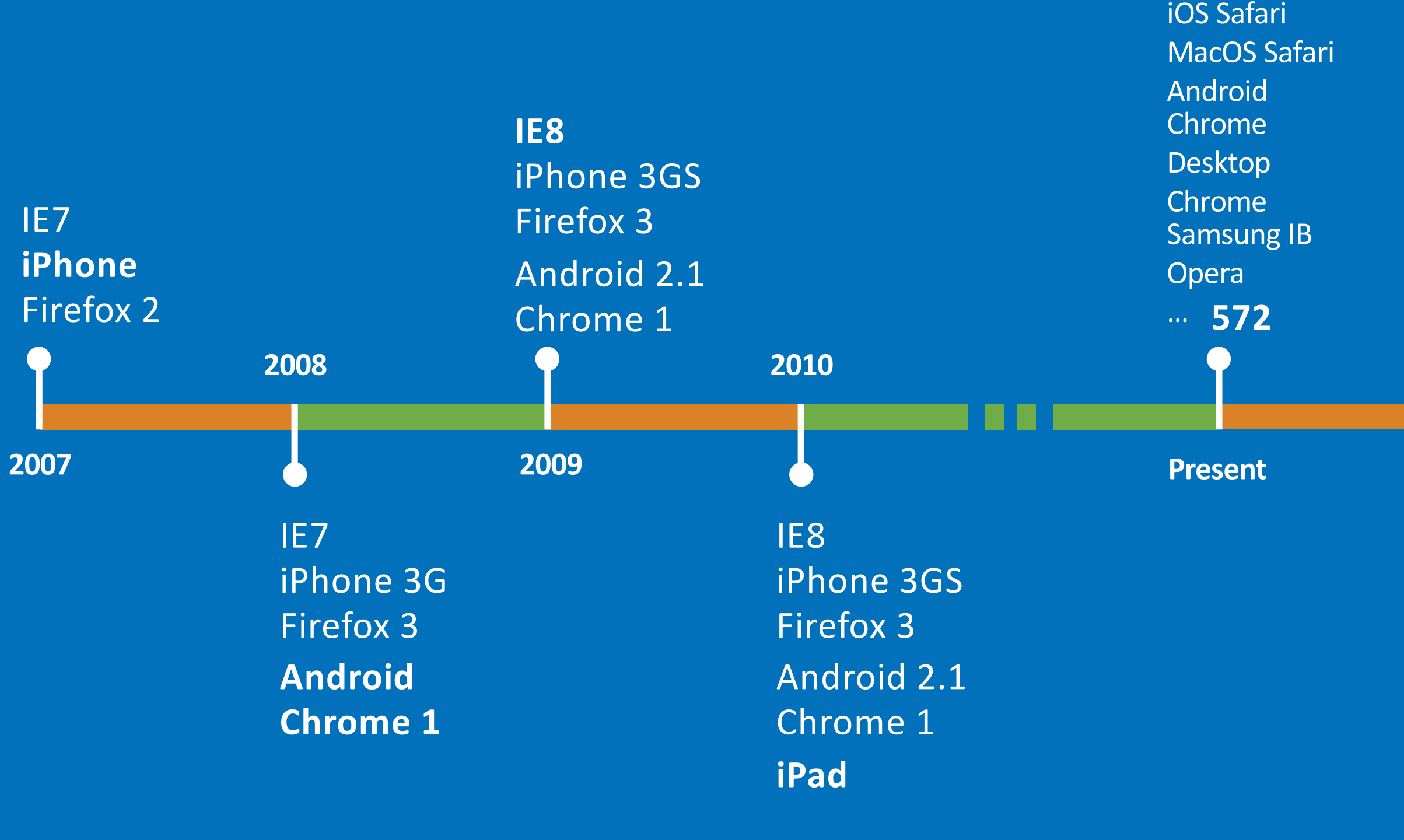
2015: 24.0k Different Android Devices

Source: OpenSignal

How many browsers in a typical Thanksgiving weekend?

How many browsers in a typical Thanksgiving weekend?

572


IE7
iPhone
Firefox 2
2007
2008
IE7
iPhone 3G
Firefox 3
Android
Chrome 1
IE8
iPhone 3GS
Firefox 3
Android 2.1
Chrome 1
2009
2010
IE8
iPhone 3GS
Firefox 3
Android 2.1
Chrome 1
iPad
iOS Safari
MacOS Safari
Android
Chrome
Desktop
Chrome
Samsung IB
Opera
… **572**
Present

“...25% of new Android phones have only 512MB of RAM.”

Jen Fitzpatrick
VP of product management for Google Maps

Browser Based Testing

How do we test the Browser?

- Checklists
- Support tickets

- Screen capture
- Pixel comparison

- Programmatic control
- RUM Beacons


#1 reason checklists fail to
catch browser regression:
Humans

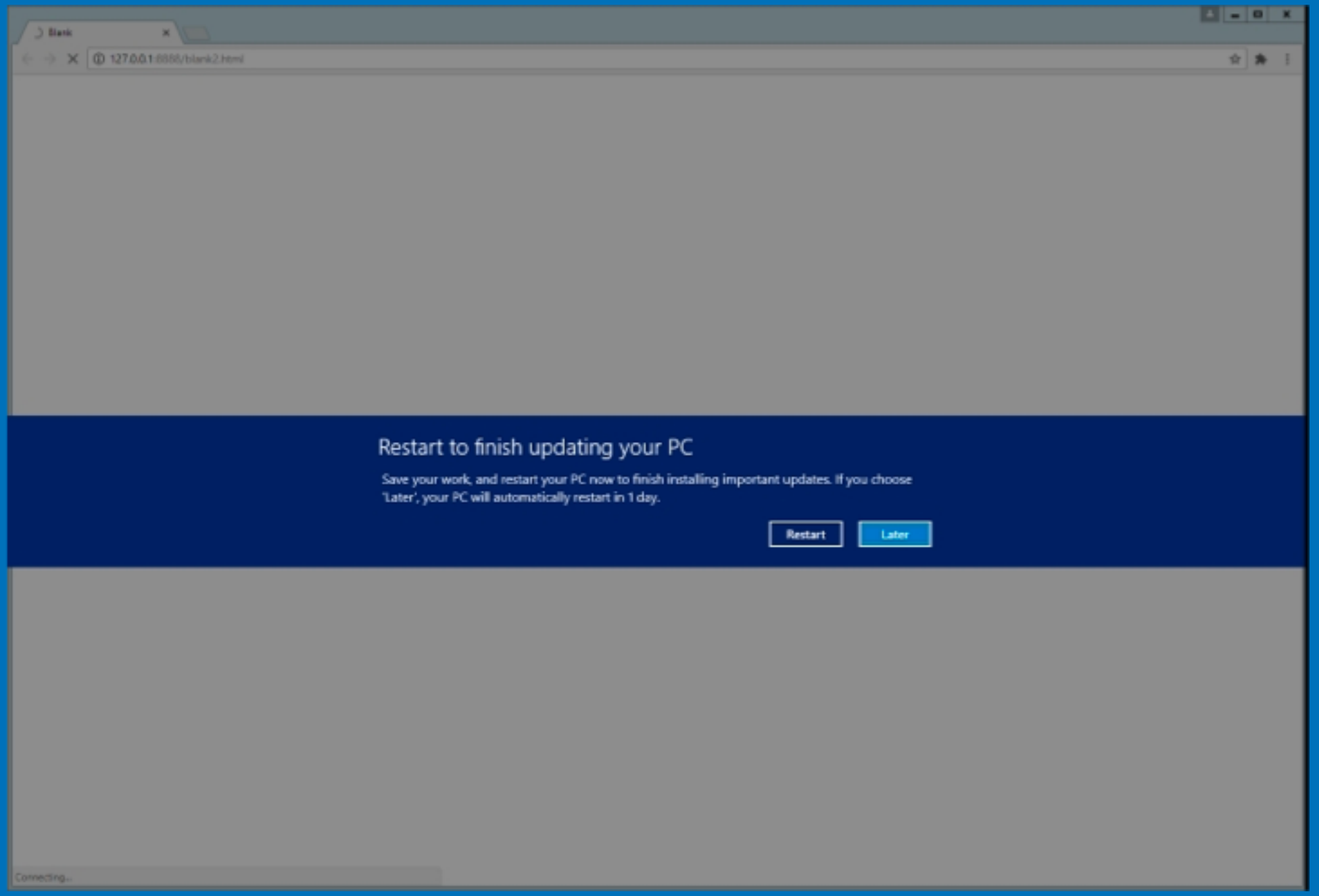
```
open -a Safari https://cloudinary.com
screencapture -x testoutput.png
killall Safari
compare -identify -metric MAE baseline.png testoutput.png null
>> same1.png[0] PNG 640x400 640x400+0+0 8-bit DirectClass
1.64KB 0.010u 0:00.019
>> same2.jpg[0] JPEG 640x400 640x400+0+0 8-bit DirectClass
3.65KB 0.000u 0:00.009
>> 0.196766 (3.00245e-06)
```

```
open -a Safari https://cloudinary.com
screencapture -x testoutput.png
killall Safari
compare -identify -metric MAE baseline.png testoutput.png null
>> same1.png[0] PNG 640x400 640x400+0+0 8-bit DirectClass
1.64KB 0.010u 0:00.019
>> same2.jpg[0] JPEG 640x400 640x400+0+0 8-bit DirectClass
3.65KB 0.000u 0:00.009
>> 0.196766 (3.00245e-06)
```

```
open -a Safari https://cloudinary.com
screencapture -x testoutput.png
killall Safari
compare -identify -metric MAE baseline.png testoutput.png null
>> same1.png[0] PNG 640x400 640x400+0+0 8-bit DirectClass
1.64KB 0.010u 0:00.019
>> same2.jpg[0] JPEG 640x400 640x400+0+0 8-bit DirectClass
3.65KB 0.000u 0:00.009
>> 0.196766 (3.00245e-06)
```

```
open -a Safari https://cloudinary.com
screencapture -x testoutput.png
killall Safari
```

```
compare -identify -metric MAE baseline.png testoutput.png null
>> same1.png[0] PNG 640x400 640x400+0+0 8-bit DirectClass
1.64KB 0.010u 0:00.019
>> same2.jpg[0] JPEG 640x400 640x400+0+0 8-bit DirectClass
3.65KB 0.000u 0:00.009
>> 0.196766 (3.00245e-06)
```


Blank
127.0.0.1:8888/blank2.html
Restart to finish updating your PC
Save your work, and restart your PC now to finish installing important updates. If you choose 'Later', your PC will automatically restart in 1 day.
Restart
Later
Connecting...

How do we test the Browser?

- Checklists
- Support tickets

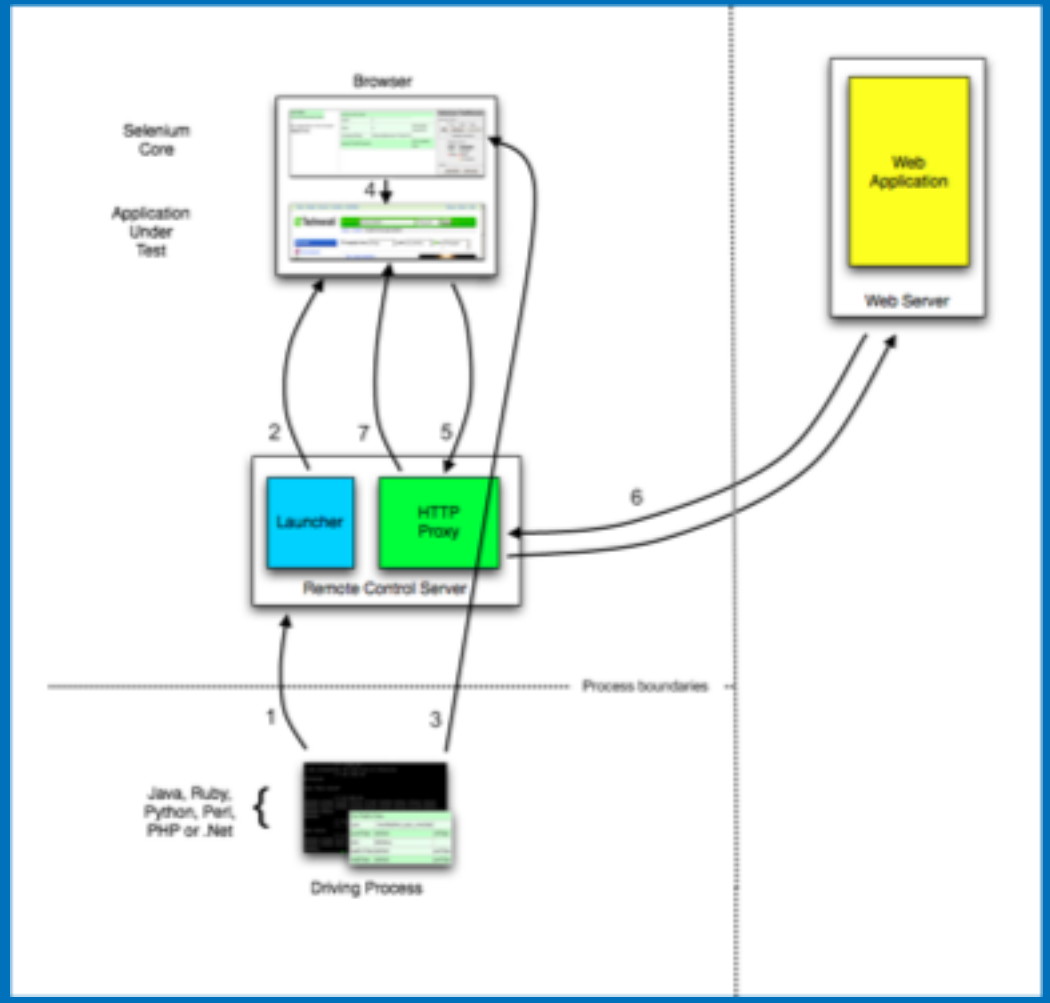
- Screen capture
- Pixel comparison

- Programmatic control
- RUM Beacons

OS API application control

- Access the window model directly
- Simulate Keyboard strokes and mouse movements
- Interact with any application like a human
- Screen Readers use this approach to supersede the OS accessibility features

Programmatic Control



How do we test the Browser?

- Checklists
- Support tickets

- Screen capture
- Pixel comparison

- Programmatic control
- RUM Beacons

W3C Recommendation

TABLE OF CONTENTS

WebDriver

W3C Recommendation 05 June 2018

This version:
https://www.w3.org/TR/2018/REC-webdriver1-20180605/

Latest published version:
https://www.w3.org/TR/webdriver1/

Latest editor's draft:
https://w3c.github.io/webdriver/

Implementation report:
https://github.com/w3c/webdriver/blob/master/implementation-report.md

Previous version:
https://www.w3.org/TR/2018/PR-webdriver1-20180426/

Editors:
Simon Stewart
David Burns (Mozilla)

Participate:
GitHub w3c/webdriver
File a bug
Commit history
Pull requests

Channel:
#webdriver on irc.w3.org

Please check the **errata** for any errors or issues reported since publication.

See also **translations**.

Abstract

WebDriver is a remote control interface that enables introspection and control of user agents. It provides a platform- and language-neutral wire protocol as a way for out-of-process programs to remotely instruct the behavior of web browsers.

W3C Recommendation

Navigation Timing

W3C Recommendation 17 December 2012

This version:
http://www.w3.org/TR/2012/REC-navigation-timing-20121217/
Latest version:
http://www.w3.org/TR/navigation-timing/
Previous version:
http://www.w3.org/TR/2012/PR-navigation-timing-20120726/
Editor:
Zhiheng Wang (Google Inc.) <zhihengw@google.com>

Please refer to the **errata** for this document, which may include some normative corrections.

See also translations.

Abstract

This specification defines an interface for web applications to access timing information related to navigation and elements.

Status of this document

This section describes the status of this document at the time of its publication. Other documents may supersede this document. A list of current W3C publications and the latest revision of this technical report can be found in the W3C technical reports index at http://www.w3.org/TR/.

This is the W3C Recommendation for "Navigation Timing Specification". An implementation report, produced during the Candidate Recommendation phase in 2012, is available based on the Navigation Timing test suite.

Please send comments to public-web-perf@w3.org (archived) with `[NavigationTiming]` at the start of the subject line.

This document is produced by the Web Performance Working Group. A diff document with the previous draft is available.

This document has been reviewed by W3C Members, by software developers, and by other W3C groups and interested parties, and is endorsed by the Director as a W3C Recommendation. It is a stable document and may be used as reference material or cited from another document. W3C's role in making the Recommendation is to draw attention to the specification and to promote its widespread deployment. This enhances the functionality and interoperability of the Web.

This document was produced by a group operating under the 5 February 2004 W3C Patent Policy. W3C maintains a public list of any patent disclosures made in connection with the deliverables of the group; that page also includes instructions for disclosing a patent. An individual who has actual knowledge of a patent which the individual believes contains Essential Claim(s) must disclose the information in accordance with section 6 of the W3C Patent Policy.

W3C Candidate Recommendation

TABLE OF CONTENTS

Beacon

W3C Candidate Recommendation 13 April 2017

This version:
https://www.w3.org/TR/2017/CR-beacon-20170413/

Latest published version:
https://www.w3.org/TR/beacon/

Latest editor's draft:
https://w3c.github.io/beacon/

Previous version:
https://www.w3.org/TR/2016/WD-beacon-20161028/

Editors:
Ilya Grigorik, Google, igrigorik@gmail.com
Alois Reitbauer, Compuware Corp., alois.reitbauer@compuware.com
Arvind Jain, Google Inc., arvind@google.com (Until January 2015)
Jatinder Mann, Microsoft Corp., jmann@microsoft.com (Until February 2014)

Repository:
We are on Github.
File a bug.
Commit history.

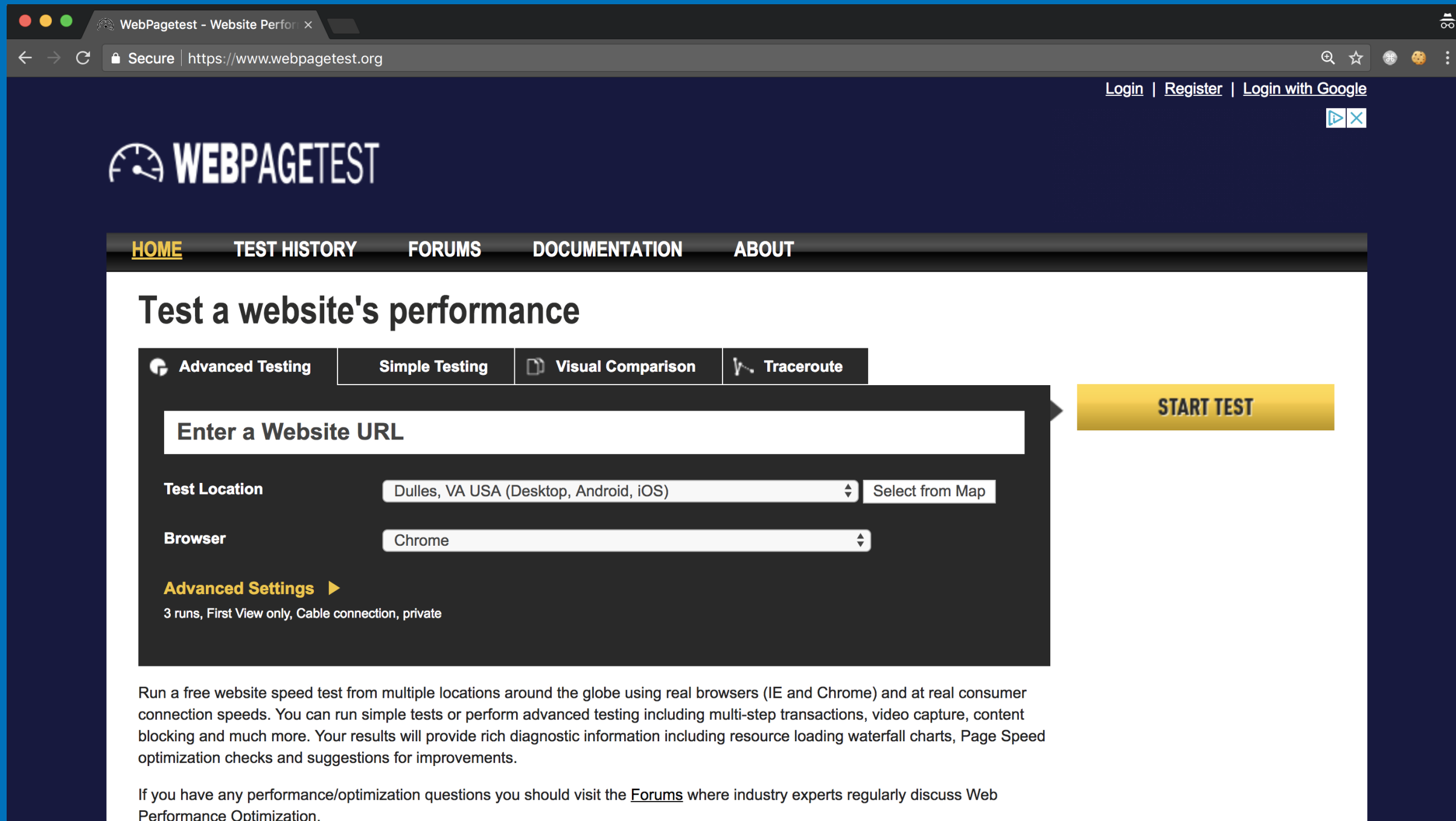
Mailing list:
public-web-perf@w3.org

Test Suite:
Test Suite

Implementation:
Implementation report
Can I use Beacon?

Abstract

This specification defines an interface that web developers can use to schedule asynchronous and non-blocking delivery of data that minimizes resource contention with other time-critical operations, while ensuring that such requests are still processed and delivered to destination.


WebPagetest - Website Perfor
Secure | https://www.webpagetest.org
Login | Register | Login with Google
WEBPAGETEST
HOME
TEST HISTORY
FORUMS
DOCUMENTATION
ABOUT
Test a website's performance
Advanced Testing
Simple Testing
Visual Comparison
Traceroute
Enter a Website URL
Test Location
Dulles, VA USA (Desktop, Android, iOS)
Select from Map
Browser
Chrome
Advanced Settings
3 runs, First View only, Cable connection, private
START TEST
Run a free website speed test from multiple locations around the globe using real browsers (IE and Chrome) and at real consumer connection speeds. You can run simple tests or perform advanced testing including multi-step transactions, video capture, content blocking and much more. Your results will provide rich diagnostic information including resource loading waterfall charts, Page Speed optimization checks and suggestions for improvements.
If you have any performance/optimization questions you should visit the Forums where industry experts regularly discuss Web Performance Optimization.

What is Selenium?

Selenium automates browsers. That's it! What you do with that power is entirely up to you. Primarily, it is for automating web applications for testing purposes, but is certainly not limited to just that. Boring web-based administration tasks can (and should!) be automated as well.

Selenium has the support of some of the largest browser vendors who have taken (or are taking) steps to make Selenium a native part of their browser. It is also the core technology in countless other browser automation tools, APIs and frameworks.

Which part of Selenium is appropriate for me?

Selenium WebDriver

If you want to

- create robust, browser-based regression automation suites and tests
- scale and distribute scripts across many environments

Then you want to use Selenium WebDriver; a collection of language specific bindings to drive a browser -- the way it is meant to be driven.

Selenium WebDriver is the successor of Selenium Remote Control which has been officially deprecated. The Selenium Server (used by both WebDriver and Remote Control) now also includes built-in grid capabilities.

Selenium IDE

If you want to

- create quick bug reproduction scripts
- create scripts to aid in automation-aided exploratory testing

Then you want to use Selenium IDE; a Firefox add-on that will do simple record-and-playback of interactions with the browser.

Selenium is a suite of tools to automate web browsers across many platforms.

Selenium...

- runs in many browsers and operating systems
- can be controlled by many programming languages and testing frameworks.

Download Selenium

Donate to Selenium

with PayPal

Donate

through sponsorship

You can sponsor the Selenium project if you'd like some public recognition of your generous contribution.

Selenium Sponsors

Want to support the Selenium project? Learn more about sponsorship or view the full list of sponsors.

Selenium-Level Sponsors

Selenium Evolution

Selenium 1

- Selenium
- Selenium Core
- Selenium RC

~~Firefox Driver~~
~~Safari Driver~~
...

Selenium 2

- Selenium WebDriver

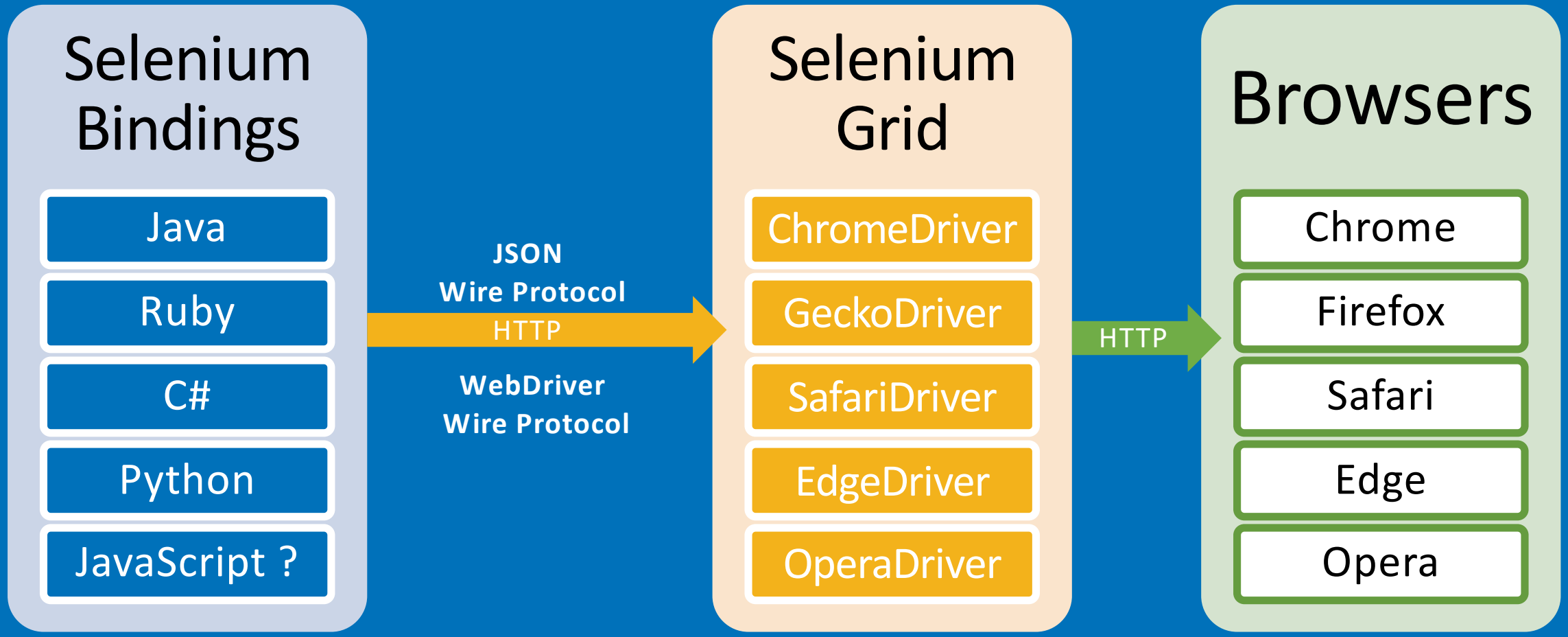
Selenium 3

- WebDriver

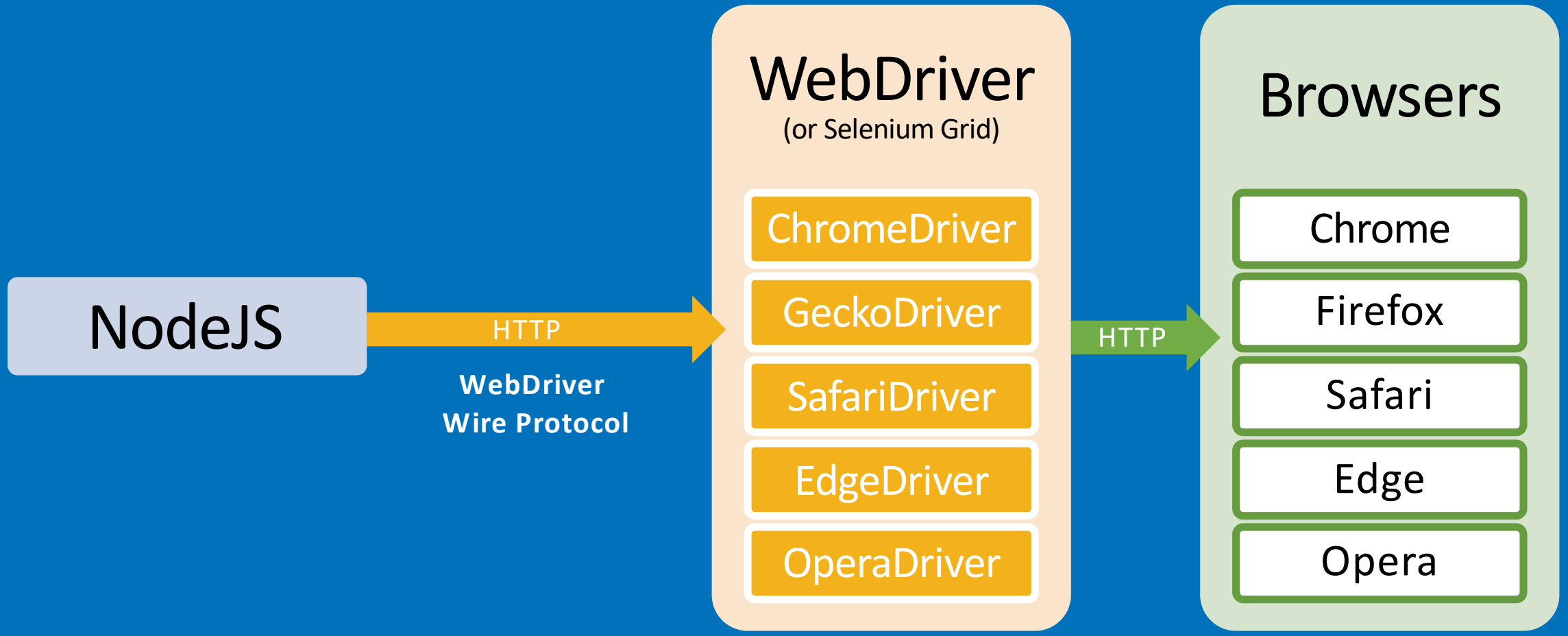
Gecko Driver (+Marionette)
Chrome Driver
EdgeDriver
Apple Safari Driver

https://www.seleniumhq.org/download/

Selenium 3 (W3C WebDriver)



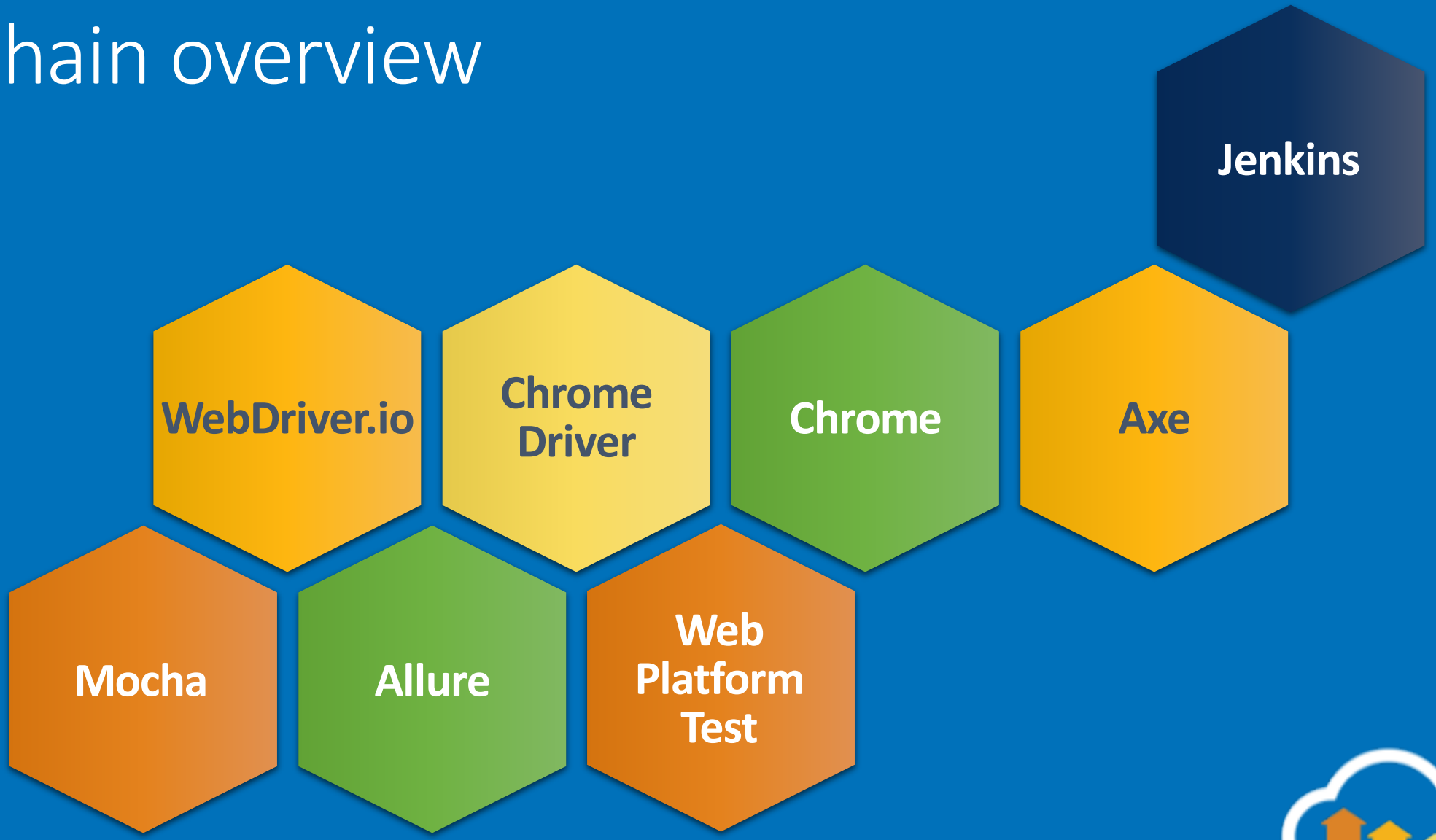
WebDriver.io



```
curl -X POST http://localhost:9515/session
       -d '{"desiredCapabilities":{"browserName":"chrome"}}'
{"sessionId":"your-session-id-here","status":0,"value":{...}}

curl http://localhost:9515/session/$SESSION_ID/url \
       -d '{"url":"http://www.cnn.com/"} '
{"sessionId":"...","status":0,"value":null}

curl http://localhost:9515/session/$SESSION_ID/element
       -d '{"using":"tagName","value":"h1"}'
{"sessionId":"...","status":0,
 "value":{"ELEMENT":"element-object-id-here"}}

curl http://localhost:9515/session/$SESSION_ID/element/$OBJECT_ID/text
{"sessionId":"...","status":0,"value":"Example Domain"}

curl -X DELETE http://localhost:9515/session/your-session-id-here
{"sessionId":"...","status":0,"value":null}
```

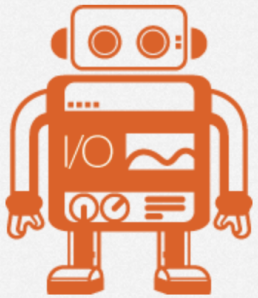
WebDriver.io

Toolchain overview



webdriver.io

Home Developer Guide API Contribute Reporters, click, .. API Version



WEBDRIVER I/O

WebDriver bindings for Node.js

npm package 4.12.0 | build failing | coverage 58%

Like 321 | G+ Star 4,034 | Tweet | Follow @webdriverio 1,678 followers

Extendable

Adding helper functions, or more complicated sets and combinations of existing commands is **simple** and really **useful**

Compatible

WebdriverIO works in combination with most of the **TDD** and **BDD** test frameworks in the JavaScript world

Feature-Rich

It implements all Webdriver protocol commands and provides **useful integrations** with other tools.

Exercise 1: Getting Setup

- https://github.com/colinbendell/webdriver-workshop
- Request https://duckduckgo.com
- Enter search term (eg: "Mr. Dressup")
- Navigate to the 4th result and report the <title>

```
git checkout
     https://github.com/colinbendell/webdriver-workshop
```

```
npm install chromedriver
```

```
npm install webdriverio
```

```
const webdriverio = require('webdriverio');
```

```
const webdriverio = require('webdriverio');

let options = {
    host: "localhost",
    port: 9515,
    path: "/",
    desiredCapabilities: { browserName: 'chrome'
};
```

```
const webdriverio = require('webdriverio');

let options = {
    host: "localhost",
    port: 9515,
    path: "/",
    desiredCapabilities: { browserName: 'chrome'
};

let browser = webdriverio.remote(options);
```

```
const webdriverio = require('webdriverio');

let options = {
    host: "localhost",
    port: 9515,
    path: "/",
    desiredCapabilities: { browserName: 'chrome'
};

let browser = webdriverio.remote(options);

browser.init();
```

```
const webdriverio = require('webdriverio');

let options = {
    host: "localhost",
    port: 9515,
    path: "/",
    desiredCapabilities: { browserName: 'chrome'
};

let browser = webdriverio.remote(options);

browser.init();
browser.url('https://amazon.com/');
browser.saveScreenshot(amazon.png');
```

```
const webdriverio = require('webdriverio');

let options = {
    host: "localhost",
    port: 9515,
    path: "/",
    desiredCapabilities: { browserName: 'chrome'
};

let browser = webdriverio.remote(options);

browser.init();
browser.url('https://amazon.com/');
browser.saveScreenshot('amazon.png');
browser.end();
```

```
const webdriverio = require('webdriverio');

let options = {
    host: "localhost",
    port: 9515,
    path: "/",
    desiredCapabilities: { browserName: 'chrome'
};

let browser = webdriverio.remote(options);

browser.init();
browser.url('https://amazon.com/');
browser.saveScreenshot(amazon.png');
browser.end();
```

Improve this doc

SELECTORS

The JsonWireProtocol provides several strategies to query an element. WebdriverIO simplifies these to make it more familiar with the common existing selector libraries like Sizzle. The following selector types are supported:

CSS Query Selector

```
1 browser.click('h2.subheading a');
```

Link Text

To get an anchor element with a specific text in it, query the text starting with an equal (=) sign. For example:

```
1 <a href="http://webdriver.io">WebdriverIO</a>
```

```
1 console.log(browser.getText('=WebdriverIO')); // outputs: "WebdriverIO"
2 console.log(browser.getAttribute('=WebdriverIO', 'href')); // outputs: "http://webdriver.i
```

Partial Link Text

To find a anchor element whose visible text partially matches your search value, query it by using `*=` in front of the query string (e.g. `*=driver`)

```
1 <a href="http://webdriver.io">WebdriverIO</a>
```

```
1 console.log(browser.getText('*=driver')); // outputs: "WebdriverIO"
```

webdriver.io/api.html

Home Developer Guide API Contribute Reporters, click, .. API Version

Search...

- Action
  - addValue
  - clearElement
  - click
  - doubleClick
  - dragAndDrop
  - leftClick
  - middleClick
  - moveToObject
  - rightClick
  - selectByAttribute
  - selectByIndex
  - selectByValue
  - selectByVisibleText
  - selectorExecute
  - selectorExecuteAsync
  - setValue
  - submitForm
- Cookie
  - deleteCookie
  - getCookie
  - setCookie
- Grid
  - getGridNodeDetails
  - gridProxyDetails
  - gridTestSession
- Mobile
  - background
  - closeApp
  - context
  - contexts
  - currentActivity
  - deviceKeyEvent

Improve this doc

WEBDRIVERIO API DOCS

Welcome to the WebdriverIO docs page. These pages contain reference materials for all implemented selenium bindings and commands. WebdriverIO has all JSONWire protocol commands implemented and also supports special bindings for Appium.

Examples

Each command documentation usually comes with an example that demonstrates the usage of it using WebdriverIO's testrunner running its commands synchronously. If you run WebdriverIO in standalone mode you still can use all commands but need to make sure that the execution order is handled properly by chaining the commands and resolving the promise chain. So instead of assigning the value directly to a variable, as the wdio testrunner allows it:

```
1 it('can handle commands synchronously', function () {
2     var value = browser.getValue('#input');
3     console.log(value); // outputs: some value
4 });
```

you need return the command promise so it gets resolved properly as well as access the value when the promise got resolve:

```
1 it('handles commands as promises', function () {
2     return browser.getValue('#input').then(function (value) {
3         console.log(value); // outputs: some value
4     });
5 });
```

Exercise 1: Getting Setup

- https://github.com/colinbendell/webdriver-workshop
- Request https://duckduckgo.com
- Enter search term (eg: "Mr. Dressup")
- Navigate to the 4th result and report the <title>

Testing

with Mocha

Improve this doc

GETTING STARTED

WebdriverIO comes with its own test runner to help you get started with integration testing as quickly as possible. All the fiddling around hooking up WebdriverIO with a test framework belongs to the past. The WebdriverIO runner does all the work for you and helps you to run your tests as efficiently as possible.

To see the command line interface help just type the following command in your terminal:

```
$ ./node_modules/.bin/wdio --help

WebdriverIO CLI runner

Usage: wdio [options] [configFile]
config file defaults to wdio.conf.js
The [options] object will override values from the config file.
An optional list of spec files can be piped to wdio that will override configured specs.

Options:
  --help, -h            prints WebdriverIO help menu
  --version, -v         prints WebdriverIO version
  --host                Selenium server host address
  --port                Selenium server port
  --path                Selenium server path (default: /wd/hub)
  --user, -u            username if using a cloud service as Selenium backend
  --key, -k             corresponding access key to the user
  --watch               watch specs for changes
  --logLevel, -l        level of logging verbosity (default: silent)
  --coloredLogs, -c     if true enables colors for log output (default: true)
  --bail                stop test runner after specific amount of tests have failed (defa
  --screenshotPath, -s  saves a screenshot to a given path if a command fails
  --baseUrl, -b         shorten url command calls by setting a base url
  --waitforTimeout, -w  timeout for all waitForXXX commands (default: 1000ms)
  --framework, -f       defines the framework (Mocha, Jasmine or Cucumber) to run the spe
  --reporters, -r       reporters to print out the results on stdout
  --suite               runs the defined suite, can be combined with --spec
```

webdriver.io/guide/testrunner/configurationfile.html

I/O Home Developer Guide API Contribute Reporters, click, .. API Version

- Get Started
- Usage
- Testrunner
  - Getting Started
  - Configuration File
  - Frameworks
  - The Browser Object
  - Organizing Suites
  - Timeouts
  - Pageobject Pattern
  - Jenkins Integration
  - Debugging
  - Retry Flaky Tests
  - Custom Reporter
  - Custom Service
- Reporters
- Services
- Plugins
- Examples

Improve this doc

CONFIGURATION FILE

The configuration file contains all necessary information to run your test suite. It is a node module that exports a JSON. Here is an example configuration with all supported properties and additional information:

```
exports.config = {

    // ====================
    // Server Configurations
    // ====================
    // Host address of the running Selenium server. This information is usually obsolete
    // WebdriverIO automatically connects to localhost. Also if you are using one of the
    // supported cloud services like Sauce Labs, Browserstack or Testing Bot you also do
    // need to define host and port information because WebdriverIO can figure that out
    // according to your user and key information. However if you are using a private Se
    // backend you should define the host address, port, and path here.
    //
    host: '0.0.0.0',
    port: 4444,
    path: '/wd/hub',
    //
    // ==================
    // Service Providers
    // ==================
    // WebdriverIO supports Sauce Labs, Browserstack and Testing Bot (other cloud provid
    // should work too though). These services define specific user and key (or access k
    // values you need to put in here in order to connect to these services.
    //
    user: 'webdriverio',
    key:  'xxxxxxxxxxxxxxxx-xxxxxx-xxxxx-xxxxxxxxx',
    //
    // ==================
    // Specify Test Files
    // ==================
    // Define which test specs should run. The pattern is relative to the directory
```

Improve this doc

FRAMEWORKS

The wdio runner currently supports Mocha, Jasmine (v2.0) and Cucumber. To integrate each framework with WebdriverIO there are adapter packages on NPM that need to be downloaded and installed. Note that these packages need to be installed at the same place WebdriverIO is installed. If you've installed WebdriverIO globally make sure you have the adapter package installed globally as well.

Within your spec files or step definition you can access the webdriver instance using the global variable `browser`. You don't need to initiate or end the Selenium session. This is taken care of by the wdio testrunner.

Using Mocha

First you need to install the adapter package from NPM:

```
npm install wdio-mocha-framework --save-dev
```

If you like to use Mocha you should additionally install an assertion library to have more expressive tests, e.g. Chai. Initialise that library in the `before` hook in your configuration file:

```
before: function() {
    var chai = require('chai');
    global.expect = chai.expect;
    chai.Should();
}
```

Once that is done you can write beautiful assertions like:

mochajs.org

simple, flexible, fun

Mocha is a feature-rich JavaScript test framework running on Node.js and in the browser, making asynchronous testing *simple* and *fun*. Mocha tests run serially, allowing for flexible and accurate reporting, while mapping uncaught exceptions to the correct test cases. Hosted on GitHub.

BACKERS

Find Mocha helpful? Become a backer and support Mocha with a monthly donation.

SPONSORS

https://mochais.

www.chaijs.com

Chai Assertion Library

Guide API Plugins

Chai is a BDD / TDD assertion library for node and the browser that can be delightfully paired with any javascript testing framework.

Download Chai for Node

4.1.2 / 2017-08-31

Another platform? Browser Rails

The **chai** package is available on npm.

```
$ npm install chai
```

View Node Guide

Issues | Fork on GitHub | Releases | Google Group | Build Status

Getting Started

Learn how to install and use Chai through a series of guided walkthroughs.

API Documentation

Explore the BDD & TDD language specifications for all available assertions.

Plugin Directory

Extend Chai's with additional assertions and vendor integration.

Chai has several interfaces that allow the developer to choose the most comfortable. The chain-capable BDD styles provide an expressive language & readable style, while the TDD assert style provides a more classical feel.

Should

```
chai.should();

foo.should.be.a('string');
foo.should.equal('bar');
foo.should.have.lengthOf(3);
tea.should.have.property('flavors')
  .with.lengthOf(3);
```

Visit Should Guide

Expect

```
var expect = chai.expect;

expect(foo).to.be.a('string');
expect(foo).to.equal('bar');
expect(foo).to.have.lengthOf(3);
expect(tea).to.have.property('flavors')
  .with.lengthOf(3);
```

Visit Expect Guide

Assert

```
var assert = chai.assert;

assert.typeOf(foo, 'string');
assert.equal(foo, 'bar');
assert.lengthOf(foo, 3)
assert.property(tea, 'flavors');
assert.lengthOf(tea.flavors, 3);
```

Visit Assert Guide

http://www.chai

```
npm install
      chai
      wdio-mocha-framework
```

```
npm install wdio-chromedriver-service
```

```
npm install
     allure
     allure-commandline
     wdio-allure-reporter
```

```
wdio config
```

```
=======================
WDIO Configuration Helper
=======================

? Where do you want to execute your tests? On my local machine
? Which framework do you want to use? mocha
? Shall I install the framework adapter for you? Yes
? Where are your test specs located? ./src/**/*_spec.js
? Which reporter do you want to use?  dot,  allure
? Shall I install the reporter library for you? Yes
? Do you want to add a service to your test setup?  Chromedriver
? Shall I install the services for you? Yes
? Level of logging verbosity error
? In which directory should screenshots gets saved if a command
fails? ./errorShots/
? What is the base url? http://localhost
```

Exercise 2: Workflow Test - Adding TODO spec

Install mocha, chai, allure

create wdio.conf.js

Navigate to todomvc.com

Using Angular implementation:

- add a ToDo
- marking it complete
- clear the list

Improve this doc

THE BROWSER OBJECT

If you use the wdio test runner you can access the webdriver instance through the global `browser` object. The session is initialized by the test runner so you don't need to call `init` command. The same goes for ending the session. This is also done by the test runner process.

Besides all commands from the api the browser object provides some more information you might be interested in during your test run:

Get desired capabilities

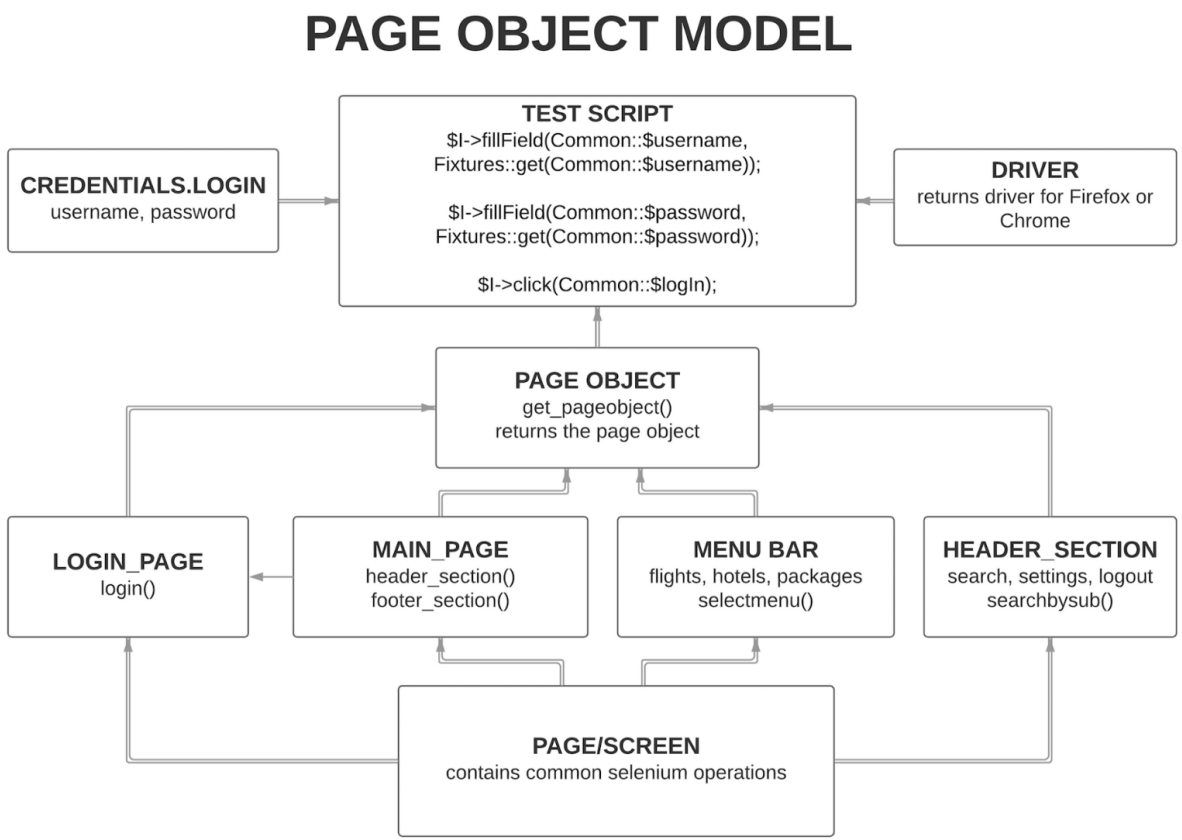
```
console.log(browser.desiredCapabilities);
/**
 * outputs:
 * {
       javascriptEnabled: true,
       locationContextEnabled: true,
       handlesAlerts: true,
       rotatable: true,
       browserName: 'chrome',
       loggingPrefs: { browser: 'ALL', driver: 'ALL' }
   }
 */
```

Get wdio config options

```
// wdio.conf.js
exports.config = {
    // ...
    foobar: true,
    // ...
}
```

Mohsin Follow
Software Test Development Engineer
May 24 · 4 min read

Page Object Model (POM) | Design Pattern



Page Object Model (POM)

GitHub, Inc. github.com/cloudinary/wdio-allure-ts

Features Business Explore Marketplace Pricing This repository Search Sign in or Sign up

cloudinary / wdio-allure-ts

Watch 5 Star 0 Fork 0

Code Issues 0 Pull requests 1 Projects 0 Insights

WebdriverIO, Allure reporter and TypeScript wrapper for UI E2E testing

23 commits 3 branches 0 releases 1 contributor MIT

Branch: master New pull request Find file Clone or download

FelixZilber validate element not visible if counter is 0 (#12) Latest commit 5060329 23 days ago

lib	validate element not visible if counter is 0 (#12)	23 days ago
src/commons	validate element not visible if counter is 0 (#12)	23 days ago
.travis.yml	update test script	3 months ago
CHANGELOG.md	initial commit	3 months ago
LICENSE	Initial commit	3 months ago
README.md	build status link fix	3 months ago
index.ts	initial commit	3 months ago
package-lock.json	Add failed test to reporter (#5)	a month ago
package.json	Add failed test to reporter (#5)	a month ago
tsconfig.json	initial commit	3 months ago
tslint.json	initial commit	3 months ago

README.md

WebdriverIO + Allure reporter + TypesScript

cloudinary.github.io/wdio-allure-ts-example/allure-report/index.html#suites/8

Allure

- Overview
- Categories
- Suites
- Graphs
- Timeline
- Behaviors
- Packages

En

Collapse

Suites

order | name | duration | status

Filter by status: 0 1 2 0 0

FailingTest 1 1

- #1 Check URL chrome 9s 482ms
- #2 Will Fail chrome 35s 423ms

TodoAngularTest 1

FailingTest.Will Fail

Broken Will Fail

Overview | History | Retries

```
Url not as expected.
                Expected: 'http://todomvc.com'
                Actual: 'https://cloudinary.com/'
```

Categories: Test defects

Severity: normal

Duration: 35s 423ms

Parameters

browser: chrome

Execution

Test body

- [STEP] - Navigate to https://cloudinary.com/ 1 attachment 1ms
- [STEP] - Validate url - expected to fail 1 attachment 1ms

Screenshot 660.5 KB

Browser console logs 650 B

Page HTML source 269.5 KB


```
Url not as expected.
                Expected: 'http://todomvc.com'
```

Exercise 3: POM spec

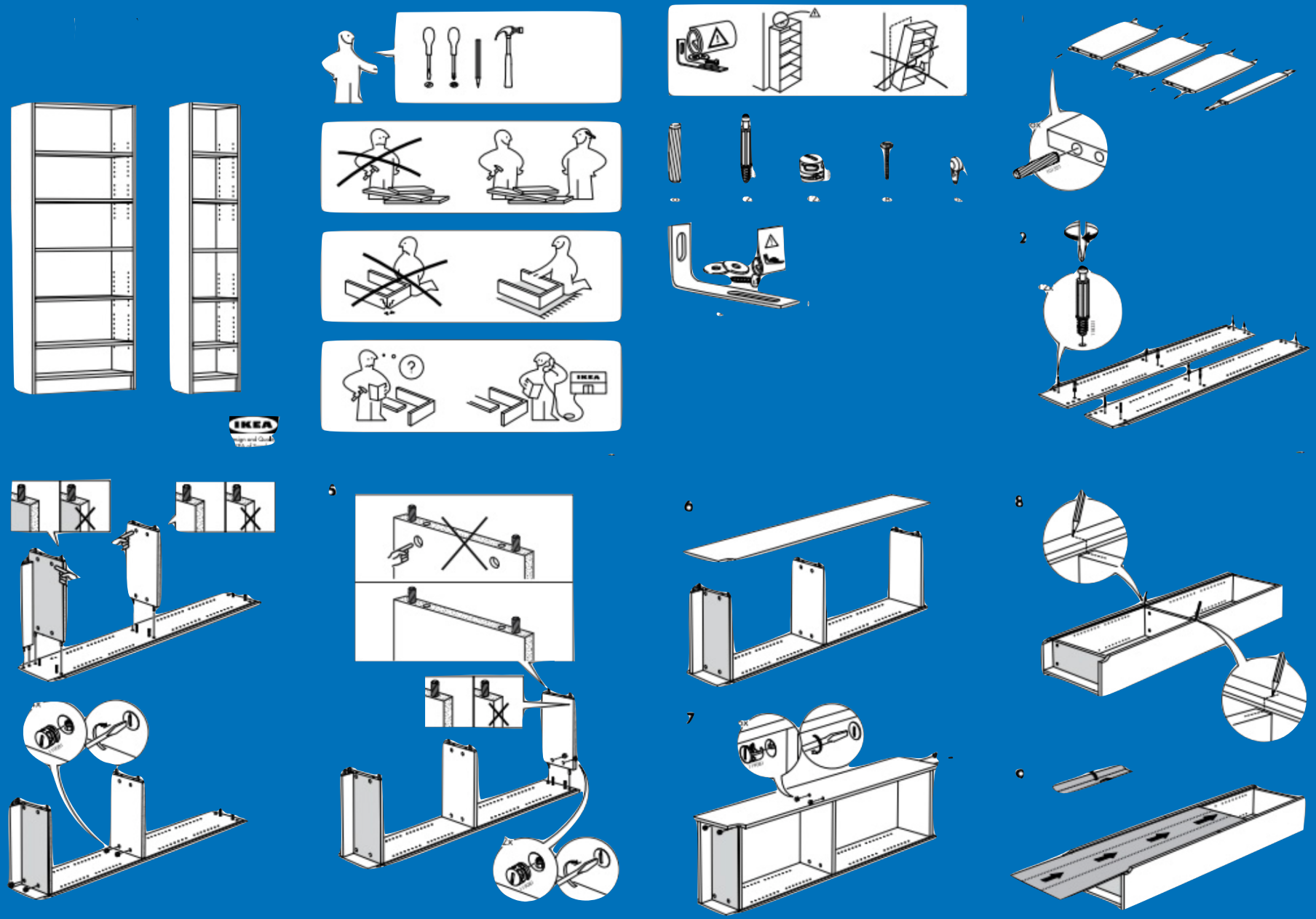
- Repeat Exercise 2
- Use Page Object Model
- Simplify Allure output with wdio-allure-ts

Take Action

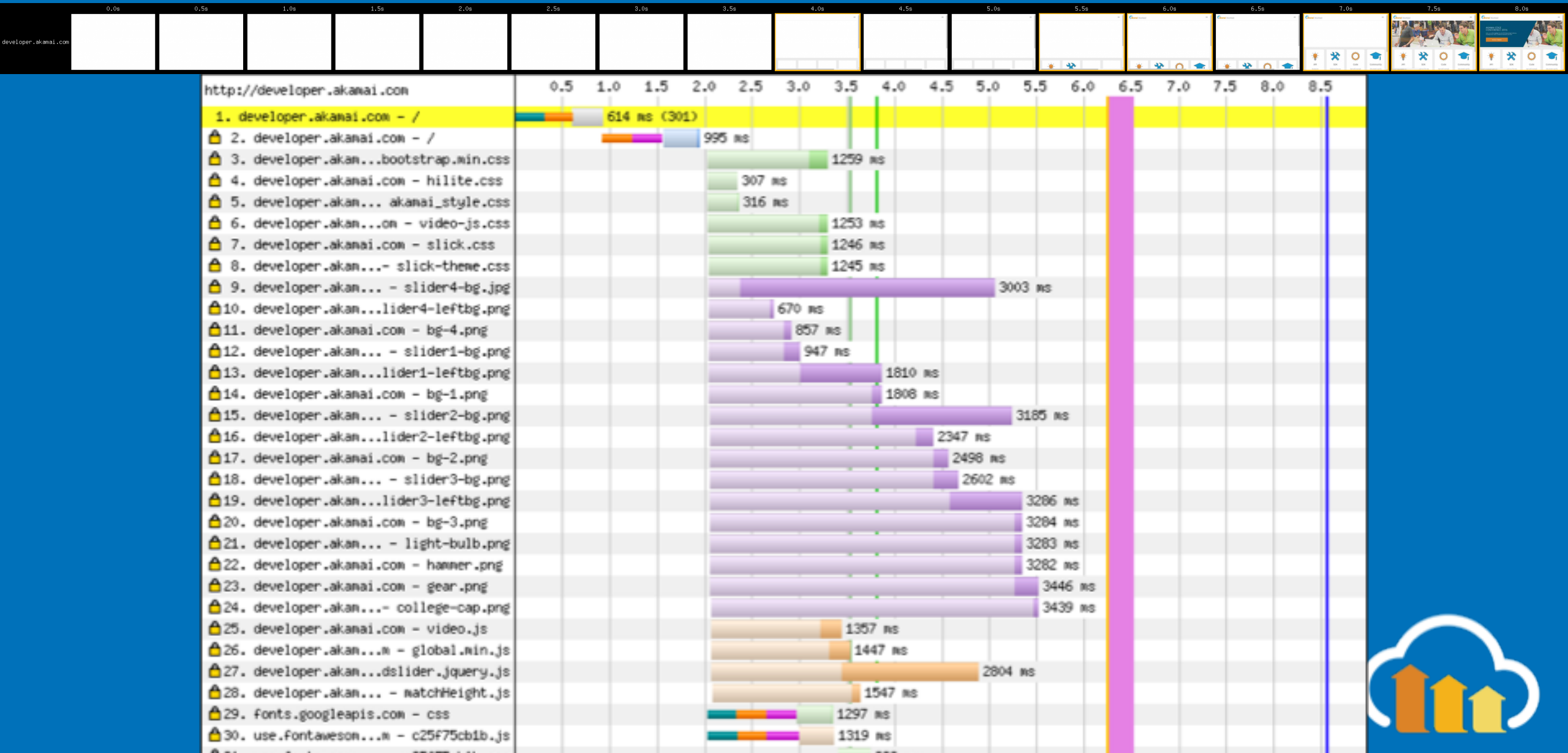
- Create workflows easily with mocha
- Use a Page-Object-Model to abstract selectors for reduced maintenance
- Create good & actionable outputs

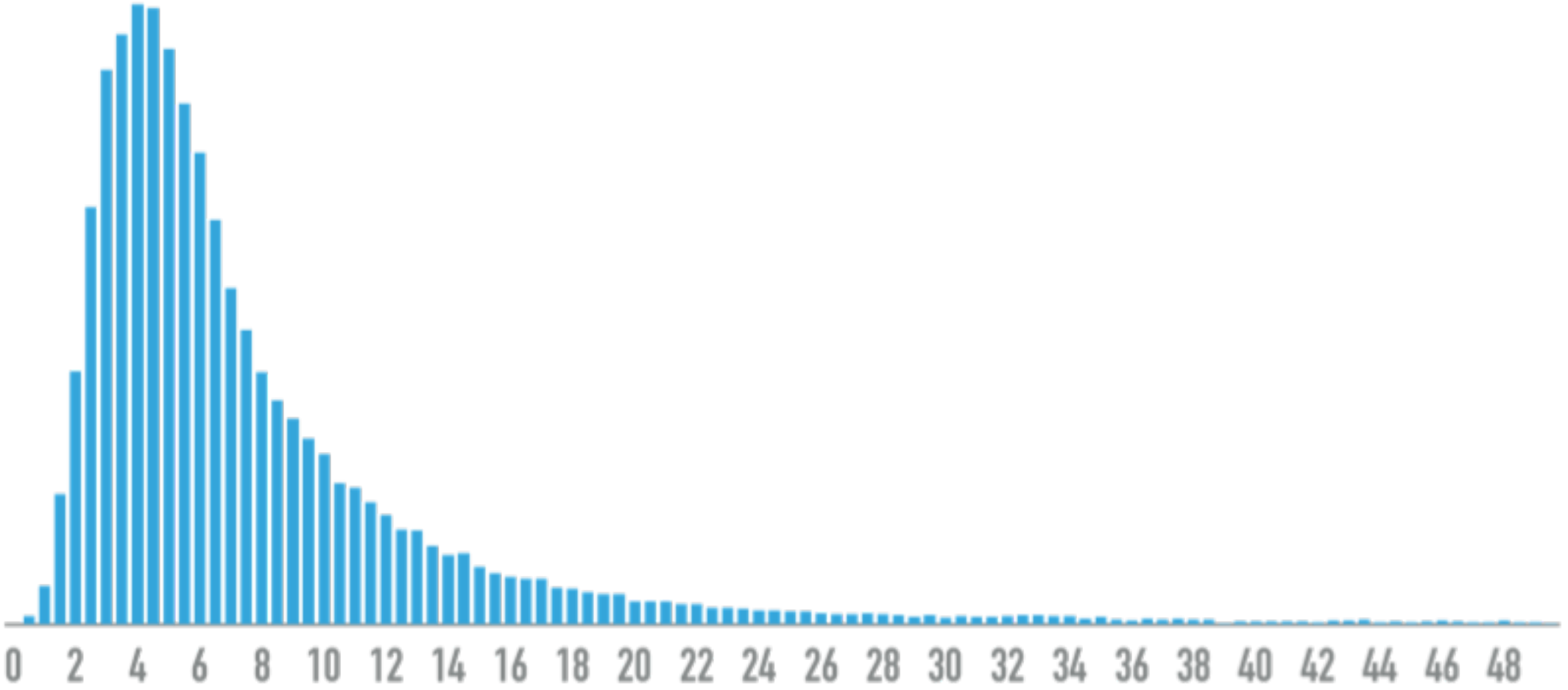
How Browsers Work

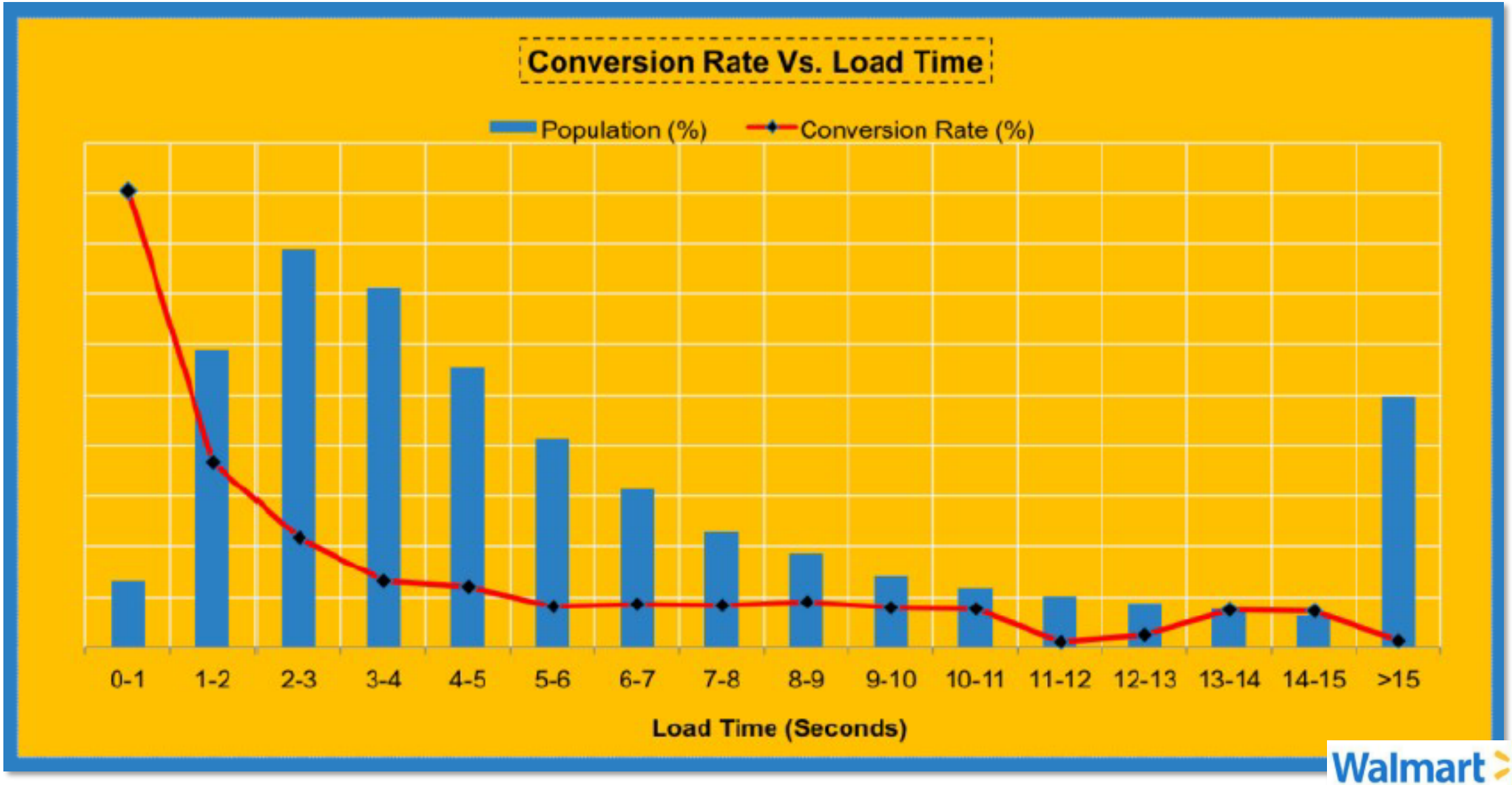
How Browsers Are Supposed to work



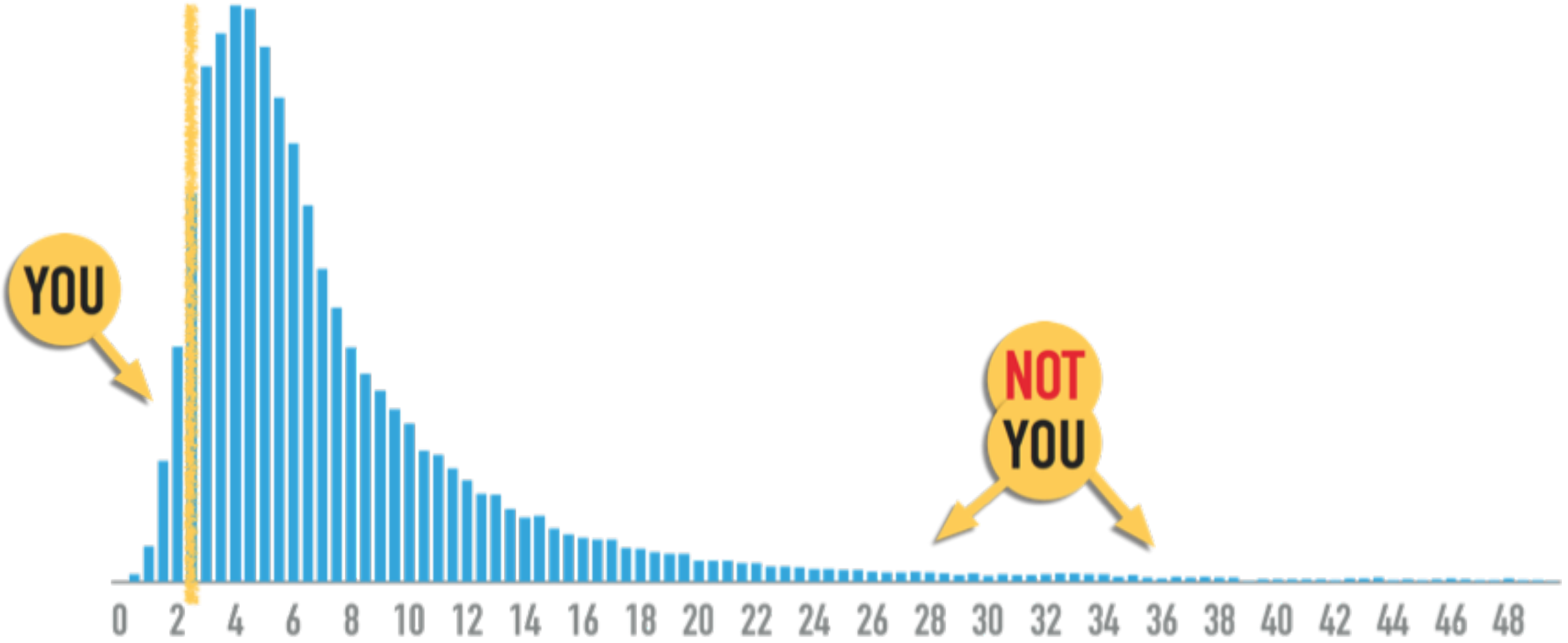

TELUS
22:33
34%
CNN
Home
Brothers are charged with multiple felonies
Black hole mystery unlocked?

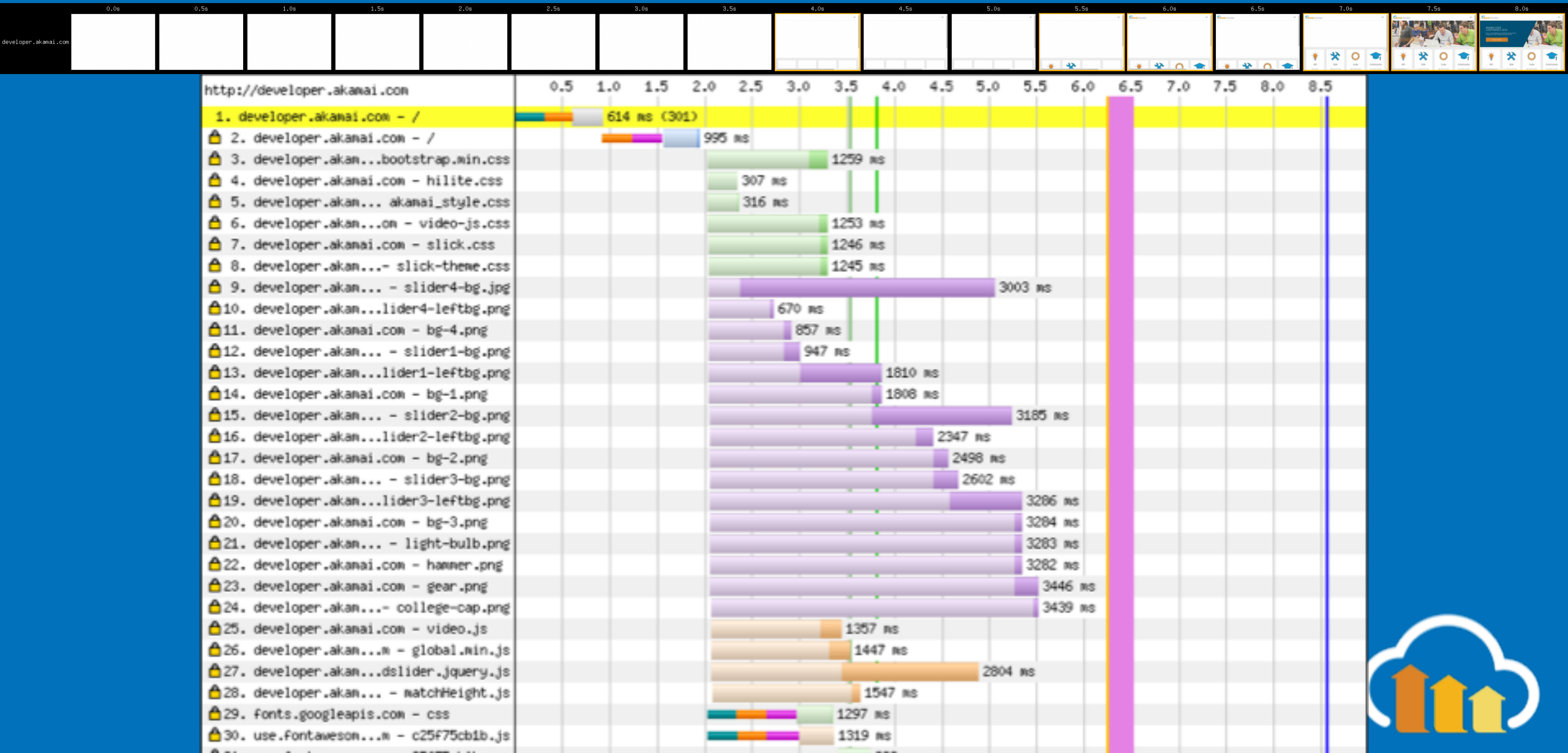

0.0s
0.5s
1.0s
1.5s
2.0s
2.5s
3.0s
3.5s
4.0s
4.5s
5.0s
5.5s
6.0s
6.5s
7.0s
7.5s
8.0s
developer.akamai.com
http://developer.akamai.com
0.5 1.0 1.5 2.0 2.5 3.0 3.5 4.0 4.5 5.0 5.5 6.0 6.5 7.0 7.5 8.0 8.5
1. developer.akamai.com - / 614 ms (301)
2. developer.akamai.com - / 995 ms
3. developer.akan...bootstrap.min.css 1259 ms
4. developer.akamai.com - hilite.css 307 ms
5. developer.akan... akamai_style.css 316 ms
6. developer.akan...om - video-js.css 1253 ms
7. developer.akamai.com - slick.css 1246 ms
8. developer.akan...- slick-theme.css 1245 ms
9. developer.akan... - slider4-bg.jpg 3003 ms
10. developer.akan...lider4-leftbg.png 670 ms
11. developer.akamai.com - bg-4.png 857 ms
12. developer.akan... - slider1-bg.png 947 ms
13. developer.akan...lider1-leftbg.png 1810 ms
14. developer.akamai.com - bg-1.png 1808 ms
15. developer.akan... - slider2-bg.png 3185 ms
16. developer.akan...lider2-leftbg.png 2347 ms
17. developer.akamai.com - bg-2.png 2498 ms
18. developer.akan... - slider3-bg.png 2602 ms
19. developer.akan...lider3-leftbg.png 3286 ms
20. developer.akamai.com - bg-3.png 3284 ms
21. developer.akan... - light-bulb.png 3283 ms
22. developer.akamai.com - hammer.png 3282 ms
23. developer.akamai.com - gear.png 3446 ms
24. developer.akan...- college-cap.png 3439 ms
25. developer.akamai.com - video.js 1357 ms
26. developer.akan...m - global.min.js 1447 ms
27. developer.akan...dslider.jquery.js 2804 ms
28. developer.akan... - matchHeight.js 1547 ms
29. fonts.googleapis.com - css 1297 ms
30. use.fontaweson...m - c25f75cb1b.js 1319 ms


0 2 4 6 8 10 12 14 16 18 20 22 24 26 28 30 32 34 36 38 40 42 44 46 48
Page Load Time: Seconds


Conversion Rate Vs. Load Time
Population (%)
Conversion Rate (%)
0-1
1-2
2-3
3-4
4-5
5-6
6-7
7-8
8-9
9-10
10-11
11-12
12-13
13-14
14-15
>15
Load Time (Seconds)

2 Second Magic Number



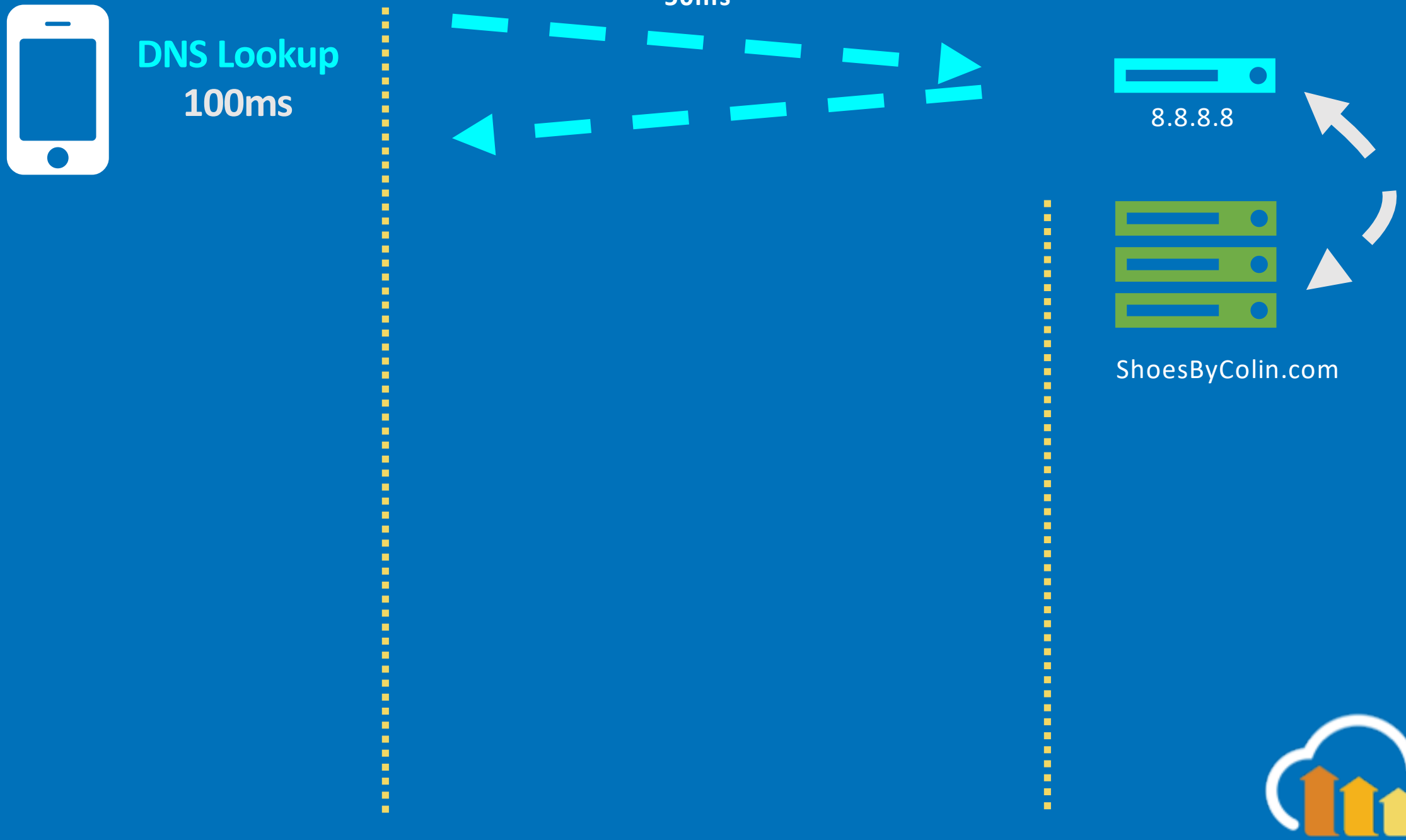

0.0s
0.5s
1.0s
1.5s
2.0s
2.5s
3.0s
3.5s
4.0s
4.5s
5.0s
5.5s
6.0s
6.5s
7.0s
7.5s
8.0s
developer.akamai.com
http://developer.akamai.com
0.5 1.0 1.5 2.0 2.5 3.0 3.5 4.0 4.5 5.0 5.5 6.0 6.5 7.0 7.5 8.0 8.5
1. developer.akamai.com - / 614 ms (301)
2. developer.akamai.com - / 995 ms
3. developer.akan...bootstrap.min.css 1259 ms
4. developer.akamai.com - hilite.css 307 ms
5. developer.akan... akamai_style.css 316 ms
6. developer.akan...om - video-js.css 1253 ms
7. developer.akamai.com - slick.css 1246 ms
8. developer.akan...- slick-theme.css 1245 ms
9. developer.akan... - slider4-bg.jpg 3003 ms
10. developer.akan...lider4-leftbg.png 670 ms
11. developer.akamai.com - bg-4.png 857 ms
12. developer.akan... - slider1-bg.png 947 ms
13. developer.akan...lider1-leftbg.png 1810 ms
14. developer.akamai.com - bg-1.png 1808 ms
15. developer.akan... - slider2-bg.png 3185 ms
16. developer.akan...lider2-leftbg.png 2347 ms
17. developer.akamai.com - bg-2.png 2498 ms
18. developer.akan... - slider3-bg.png 2602 ms
19. developer.akan...lider3-leftbg.png 3286 ms
20. developer.akamai.com - bg-3.png 3284 ms
21. developer.akan... - light-bulb.png 3283 ms
22. developer.akamai.com - hammer.png 3282 ms
23. developer.akamai.com - gear.png 3446 ms
24. developer.akan...- college-cap.png 3439 ms
25. developer.akamai.com - video.js 1357 ms
26. developer.akan...m - global.min.js 1447 ms
27. developer.akan...dslider.jquery.js 2804 ms
28. developer.akan... - matchHeight.js 1547 ms
29. fonts.googleapis.com - css 1297 ms
30. use.fontaweson...m - c25f75cb1b.js 1319 ms

DNS Lookup
100ms

50ms

8.8.8.8

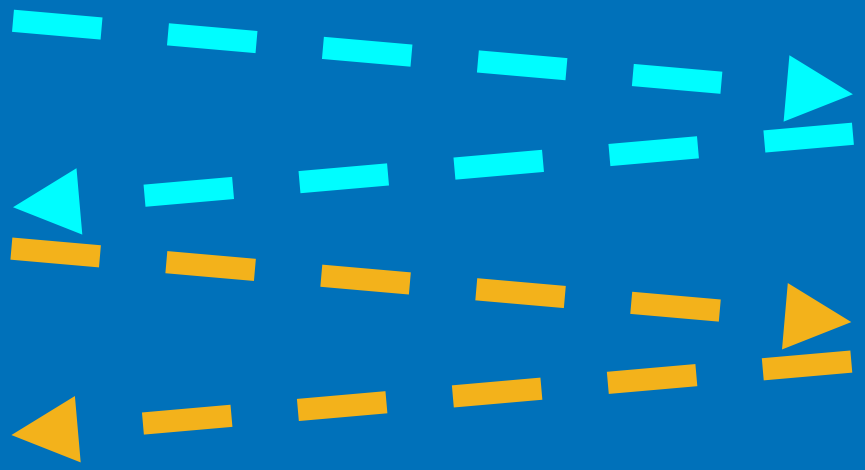
ShoesByColin.com


DNS Lookup
100ms
50ms
8.8.8.8
ShoesByColin.com

DNS Lookup
100ms

TCP Connect
+100ms

50ms



8.8.8.8

ShoesByColin.com

DNS Lookup
100ms

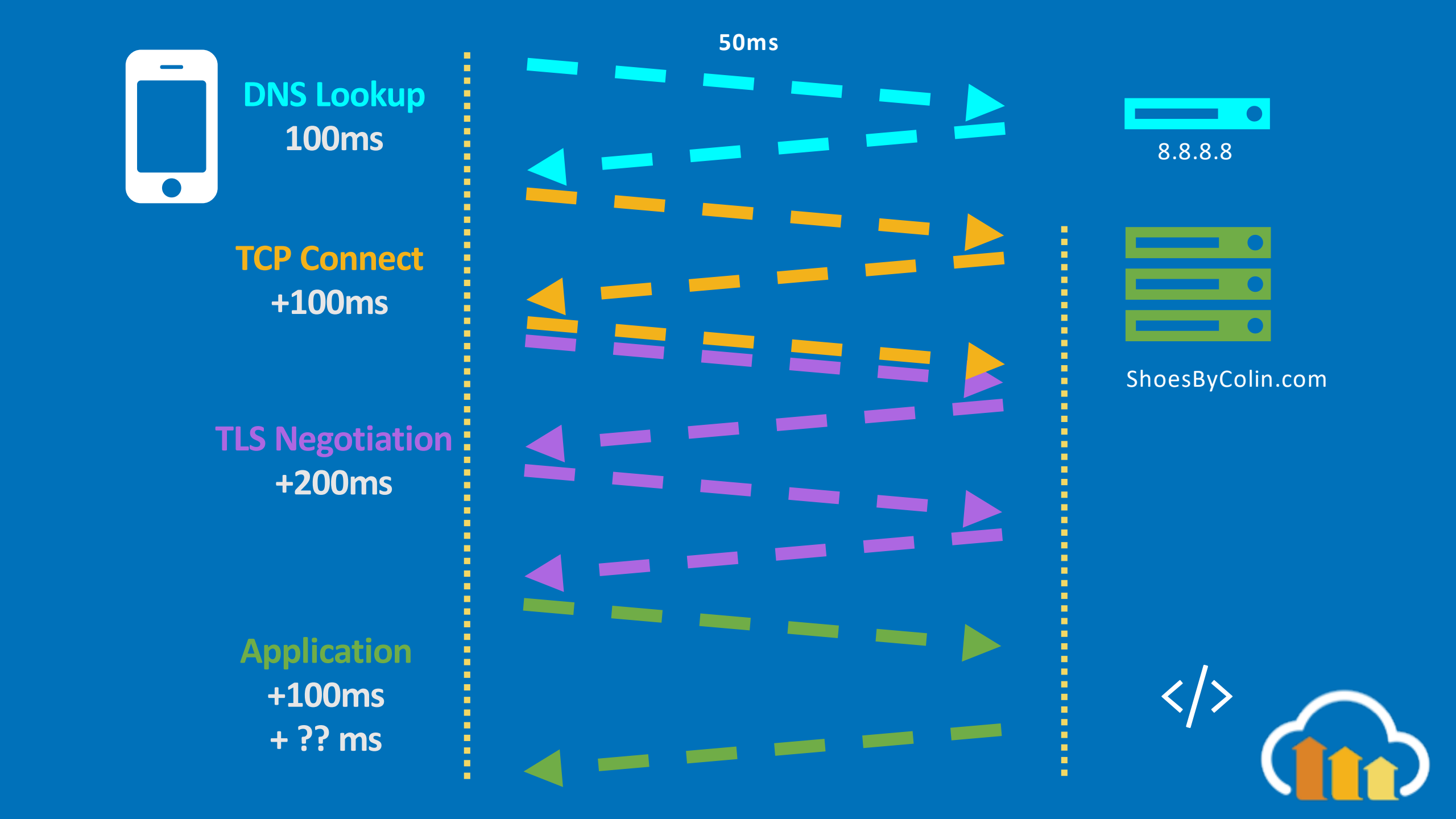
TCP Connect
+100ms

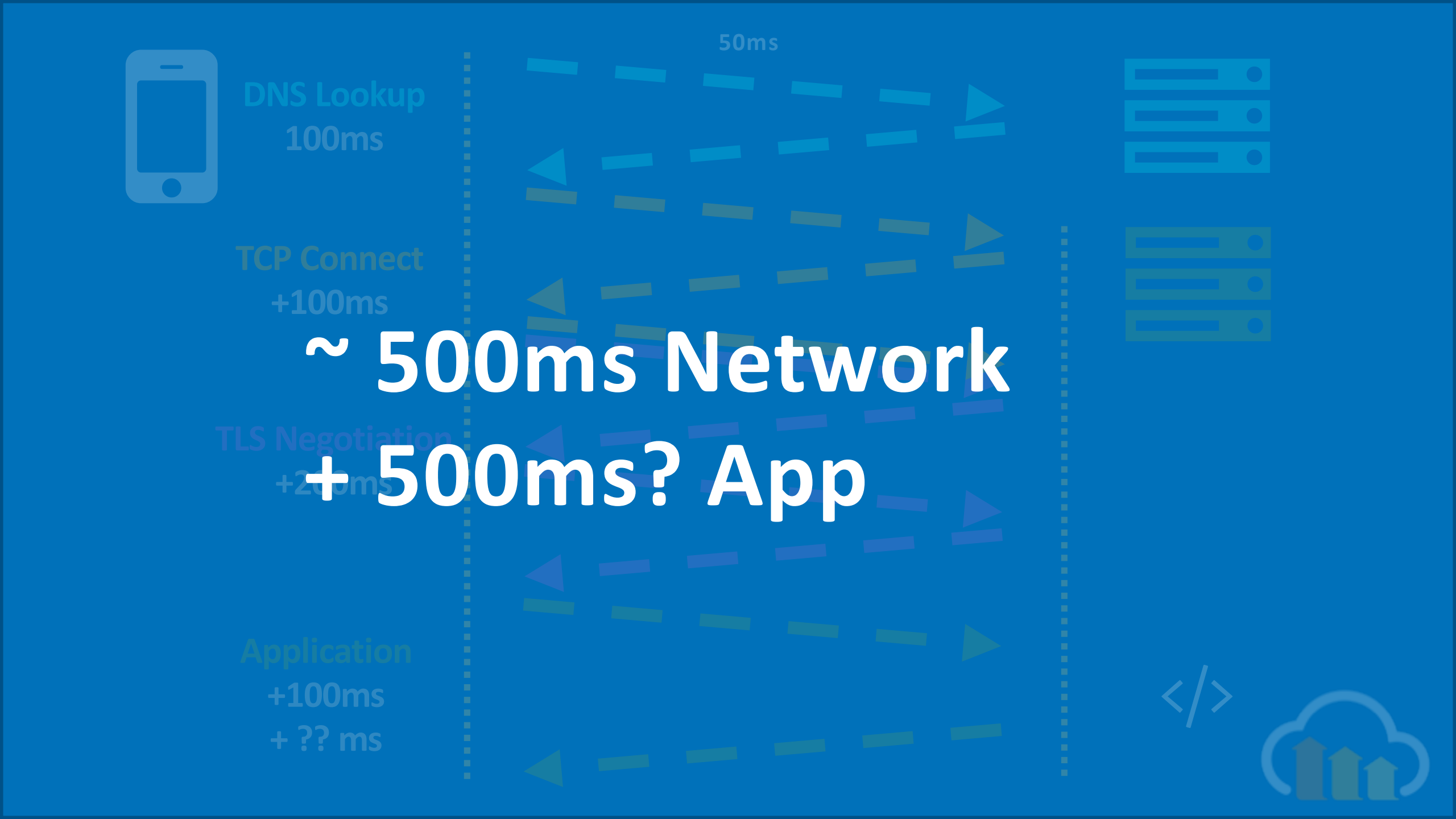
TLS Negotiation
+200ms

50ms

8.8.8.8

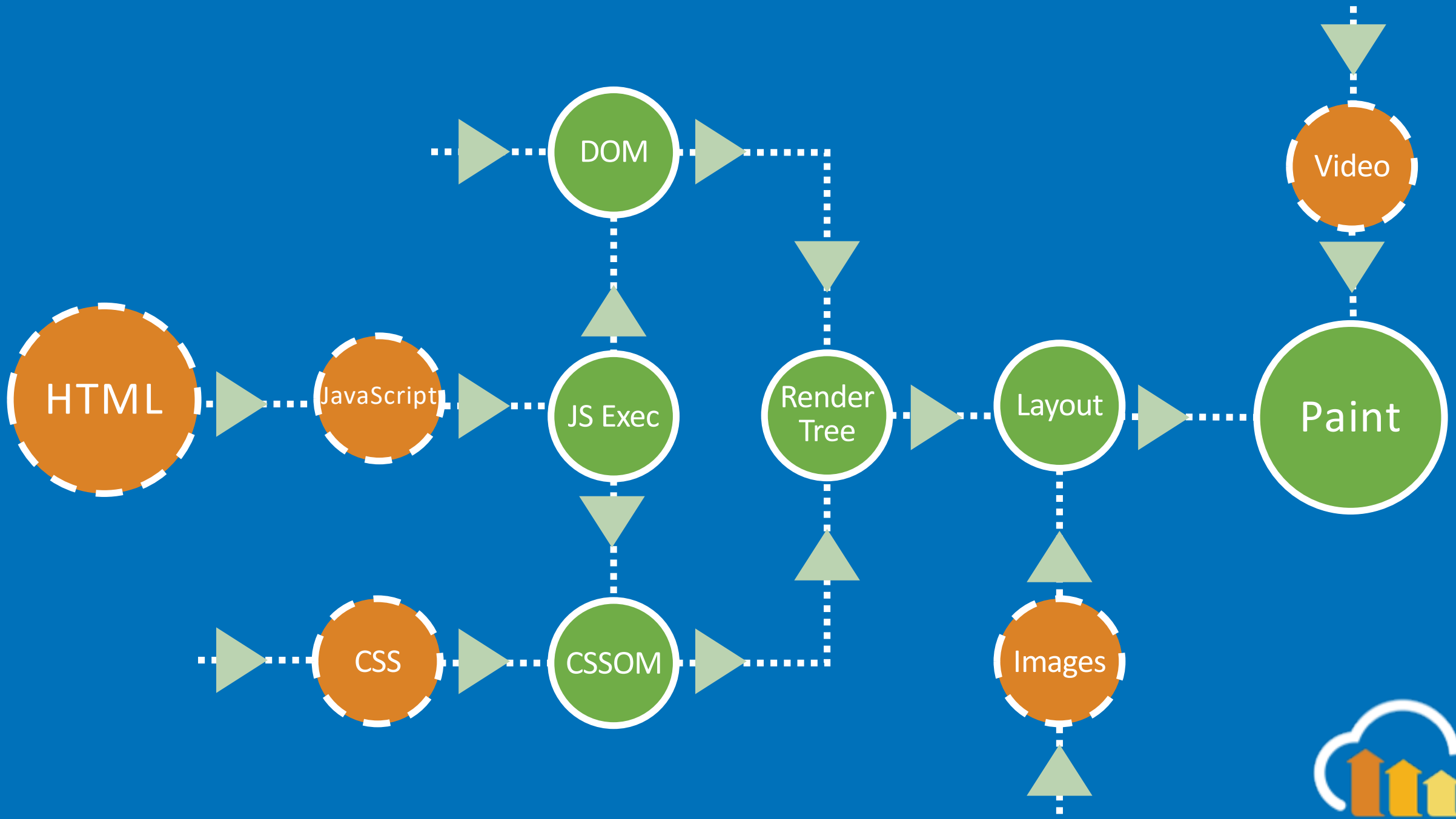
ShoesByColin.com


50ms
DNS Lookup
100ms
TCP Connect
+100ms
TLS Negotiation
+200ms
Application
+100ms
+ ?? ms
8.8.8.8
ShoesByColin.com



~ 500ms Network
+ 500ms? App


TELUS
22:33
34%
CNN
Home
Brothers are charged with multiple felonies
Black hole mystery unlocked?


DOM
Video
HTML
JavaScript
JS Exec
Render Tree
Layout
Paint
CSS
CSSOM
Images

Browser APIs

Real User Monitoring: JavaScript Timers

Steve Souders

https://github.com/stevesouders/episodes

Philip Tellis

https://github.com/lognormal/boomerang

Web Performance Working Group Charter

The **mission** of the Web Performance Working Group, part of the Rich Web Client Activity, is to provide methods to observe and improve aspects of application performance of user agent features and APIs.

https://www.w3.org/wiki/Web_Performance/Publications

W3C Recommendation

Navigation Timing

W3C Recommendation 17 December 2012

This version:
http://www.w3.org/TR/2012/REC-navigation-timing-20121217/
Latest version:
http://www.w3.org/TR/navigation-timing/
Previous version:
http://www.w3.org/TR/2012/PR-navigation-timing-20120726/
Editor:
Zhiheng Wang (Google Inc.) <zhihengw@google.com>

Please refer to the **errata** for this document, which may include some normative corrections.

See also translations.

Abstract

This specification defines an interface for web applications to access timing information related to navigation and elements.

Status of this document

This section describes the status of this document at the time of its publication. Other documents may supersede this document. A list of current W3C publications and the latest revision of this technical report can be found in the W3C technical reports index at http://www.w3.org/TR/.

This is the W3C Recommendation for "Navigation Timing Specification". An implementation report, produced during the Candidate Recommendation phase in 2012, is available based on the Navigation Timing test suite.

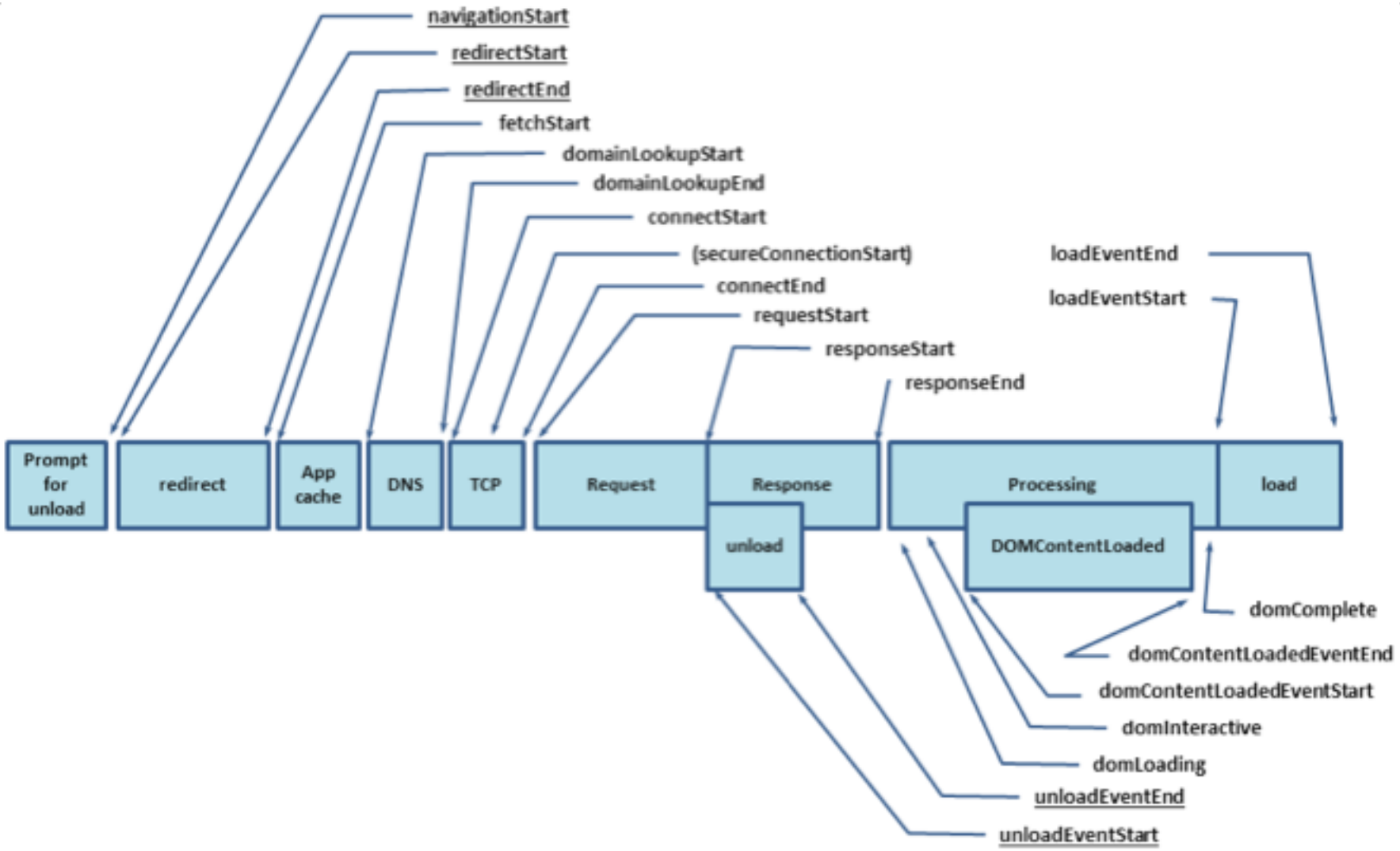
Please send comments to public-web-perf@w3.org (archived) with `[NavigationTiming]` at the start of the subject line.

This document is produced by the Web Performance Working Group. A diff document with the previous draft is available.

This document has been reviewed by W3C Members, by software developers, and by other W3C groups and interested parties, and is endorsed by the Director as a W3C Recommendation. It is a stable document and may be used as reference material or cited from another document. W3C's role in making the Recommendation is to draw attention to the specification and to promote its widespread deployment. This enhances the functionality and interoperability of the Web.

This document was produced by a group operating under the 5 February 2004 W3C Patent Policy. W3C maintains a public list of any patent disclosures made in connection with the deliverables of the group; that page also includes instructions for disclosing a patent. An individual who has actual knowledge of a patent which the individual believes contains Essential Claim(s) must disclose the information in accordance with section 6 of the W3C Patent Policy.

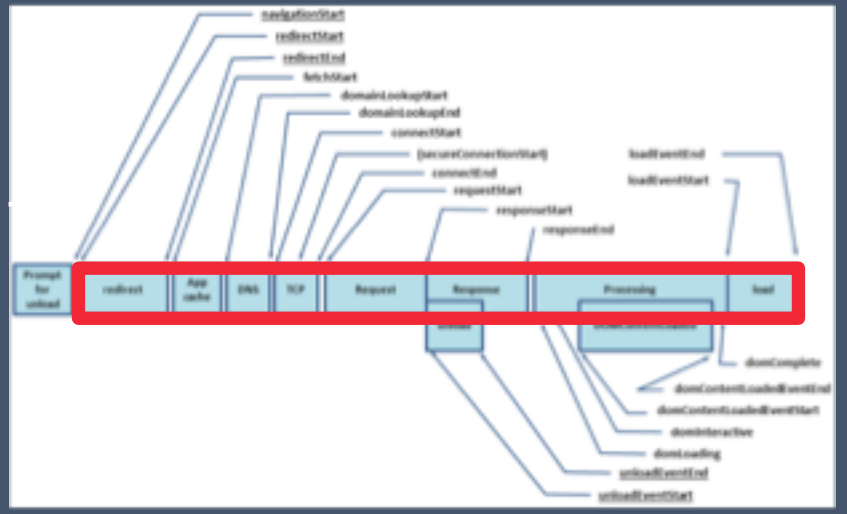
Navigation Timing API



```
window.performance
```

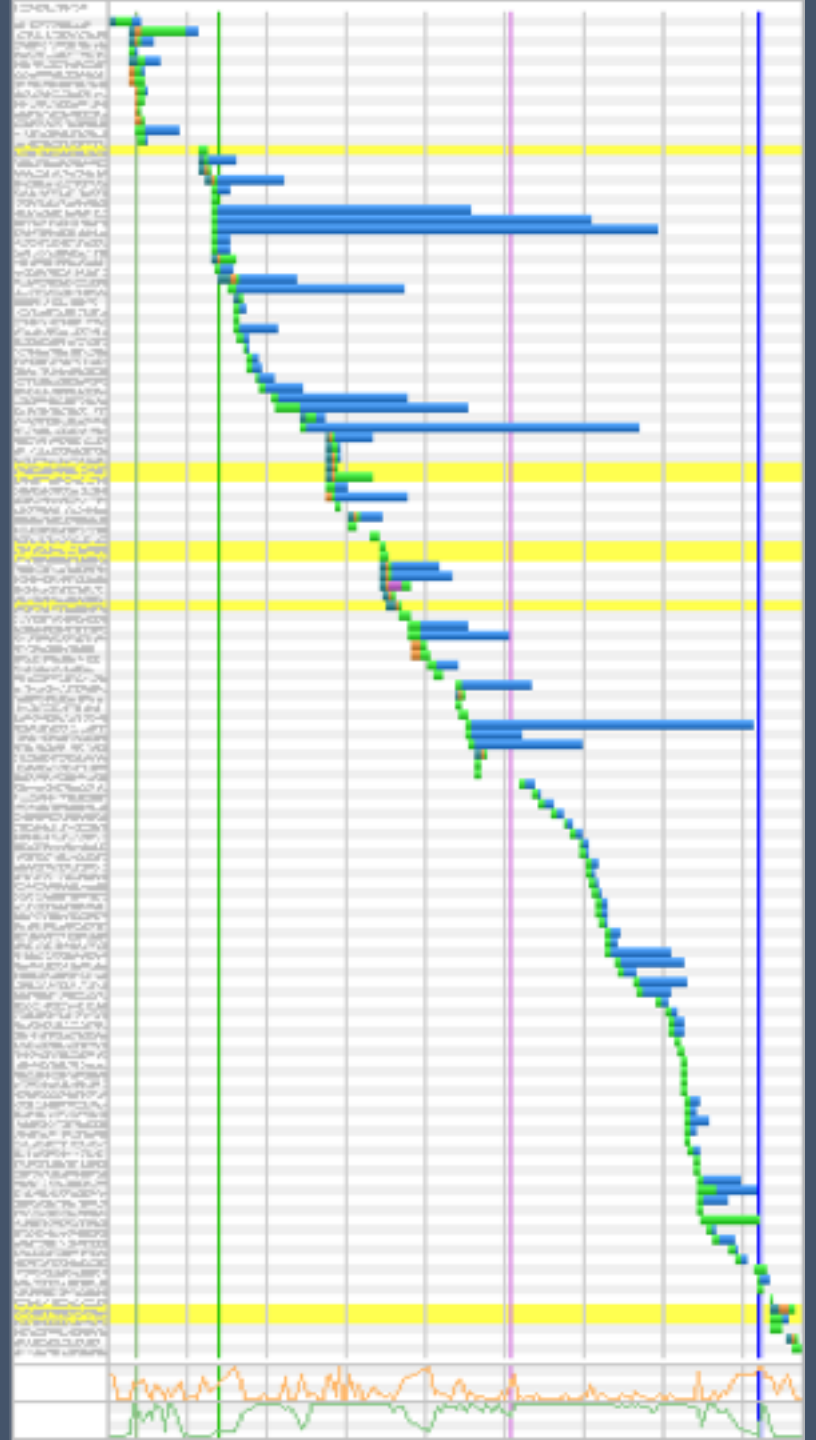
```
{
  "timeOrigin": 1528900579651.6672,
  "timing": {
    "navigationStart": 1528900579651,
    "unloadEventStart": 0,
    "unloadEventEnd": 0,
    "redirectStart": 0,
    "redirectEnd": 0,
    "fetchStart": 1528900583010,
    "domainLookupStart": 1528900579730,
    "domainLookupEnd": 1528900579730,
    "connectStart": 1528900579730,
    "connectEnd": 1528900579761,
    "secureConnectionStart": 1528900579733,
    "requestStart": 1528900579761,
    "responseStart": 1528900582922,
    "responseEnd": 1528900583010,
    "domLoading": 1528900583014,
    "domInteractive": 1528900583849,
    "domContentLoadedEventStart": 1528900583849,
    "domContentLoadedEventEnd": 1528900583876,
    "domComplete": 1528900585616,
    "loadEventStart": 1528900585617,
    "loadEventEnd": 1528900585856
  },
  "navigation": {"type": 0, "redirectCount": 0 }
}
```

```
function getPageLoadTime() {

   let timing = window.performance.timing

   return timing.loadEventStart
          - timing.navigationStart

}
```



What about …

Javascript | CSS | WebFonts | Images | Videos | XHR/Fetch



W3C Working Draft

TABLE OF CONTENTS

Resource Timing Level 2

W3C Working Draft 18 May 2018

This version:
https://www.w3.org/TR/2018/WD-resource-timing-2-20180518/

Latest published version:
https://www.w3.org/TR/resource-timing-2/

Latest editor's draft:
https://w3c.github.io/resource-timing/

Previous version:
https://www.w3.org/TR/2018/WD-resource-timing-2-20180517/

Editors:
Todd Reifsteck (Microsoft Corp.)
Ilya Grigorik (Google)
Arvind Jain (Google Inc.) (Until December 2014)
Jatinder Mann (Microsoft Corp.) (Until February 2014)
Zhiheng Wang (Google Inc.) (Until July 2012)
Anderson Quach (Microsoft Corp.) (Until March 2011)

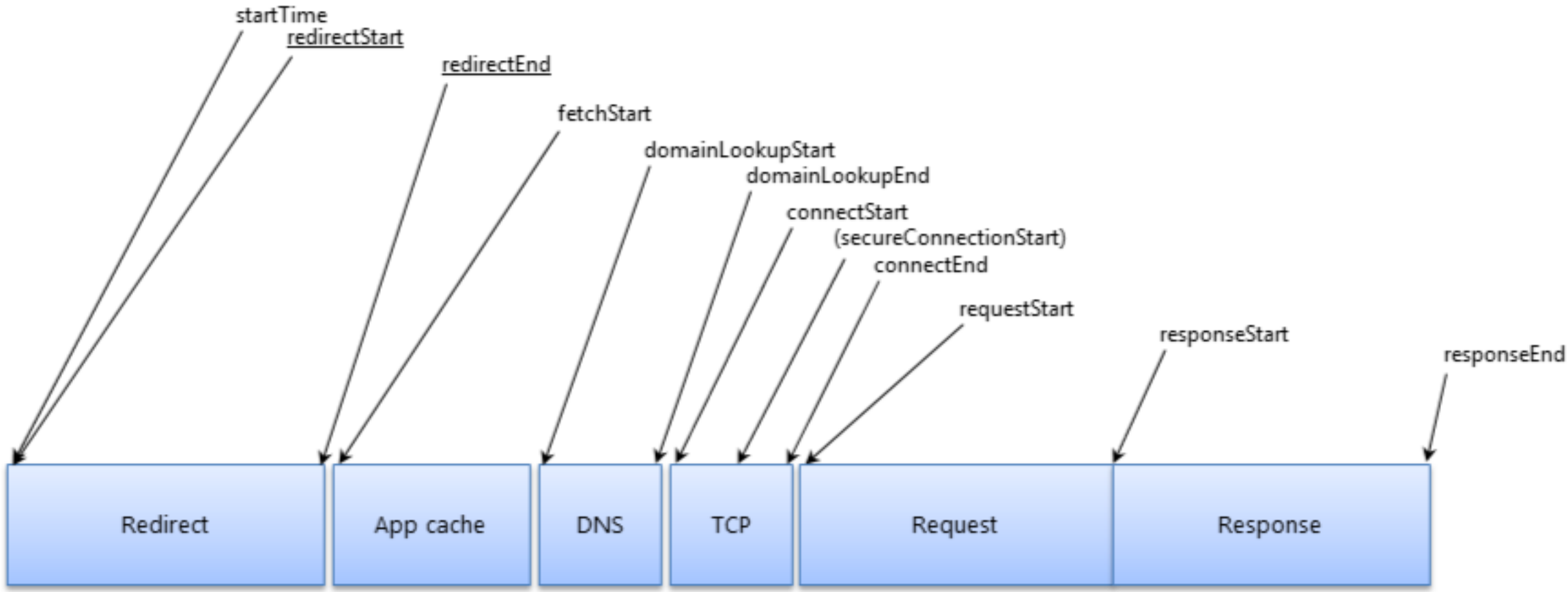
Participate:
GitHub w3c/resource-timing
File a bug
Commit history
Pull requests

Implementation:
Can I use Resource Timing?

Abstract

This specification defines an interface for web applications to access the complete timing information for resources in a document.

Status of This Document


startTime
redirectStart
redirectEnd
fetchStart
domainLookupStart
domainLookupEnd
connectStart
(secureConnectionStart)
connectEnd
requestStart
responseStart
responseEnd
Redirect
App cache
DNS
TCP
Request
Response

```
window.performance
```

```
{
  "name": "https://ajax.googleapis.com/ajax/libs/jquery/1.11.2/jquery.min.js",
  "entryType": "resource",
  "startTime": 928.70000000039,
  "duration": 425.8000000008906,
  "initiatorType": "script",
  "nextHopProtocol": "h2",
  "workerStart": 0,
  "redirectStart": 0,
  "redirectEnd": 0,
  "fetchStart": 928.70000000039,
  "domainLookupStart": 937.4000000025262,
  "domainLookupEnd": 945.9000000024389,
  "connectStart": 945.9000000024389,
  "connectEnd": 967.5999999999476,
  "secureConnectionStart": 946.2000000021362,
  "requestStart": 949.5999999999185,
  "responseStart": 970.9000000002561,
  "responseEnd": 1354.5000000012806,
  "transferSize": 33798,
  "encodedBodySize": 33495,
  "decodedBodySize": 95931,
  "serverTiming": []
}
```

```
let url = “https://bendell.ca/logo.png”;
let me = performance.getEntriesByName(url)[0];

let timings = {
    loadTime: me.duration,
    dns: me.domainLookupEnd - me.domainLookupStart,
    tcp: me.connectEnd - me.connectStart,
    waiting: me.responseStart - me.requestStart,
    fetch: me.responseEnd - me.responseStart
}
```

```
200 OK HTTP/1.1
Timing-Allow-Origin: *
Content-Type: application/javascript
Content-Length: 1234
```

- Resource Timing API
- Other Uses

  - Slowest resources
  - Time to first image
  - Response time by domain
  - Time a group of assets
  - Response time by initiator type (element type)
  - Browser cache-hit ratio for resources

More: http://www.slideshare.net/bluesmoon/beyond-page-level-metrics

W3C Recommendation

User Timing

W3C Recommendation 12 December 2013

This version:
http://www.w3.org/TR/2013/REC-user-timing-20131212/
Latest version:
http://www.w3.org/TR/user-timing/
Previous version:
http://www.w3.org/TR/2013/PR-user-timing-20131031/
Editors:
Jatinder Mann, Microsoft Corp., jmann@microsoft.com
Zhiheng Wang, Google Inc. *(Until July 2013)*
Anderson Quach, Microsoft Corp. *(Until March 2011)*

Please refer to the **errata** for this document, which may include some normative corrections.

See also translations.

Abstract

This specification defines an interface to help web developers measure the performance of their applications by giving them access to high precision timestamps.

Status of this document

This section describes the status of this document at the time of its publication. Other documents may supersede this document. A list of current W3C publications and the latest revision of this technical report can be found in the W3C technical reports index at http://www.w3.org/TR/.

This is the W3C Recommendation of "User Timing Specification", produced by the Web Performance Working Group. An implementation report as of August 2013 is available.

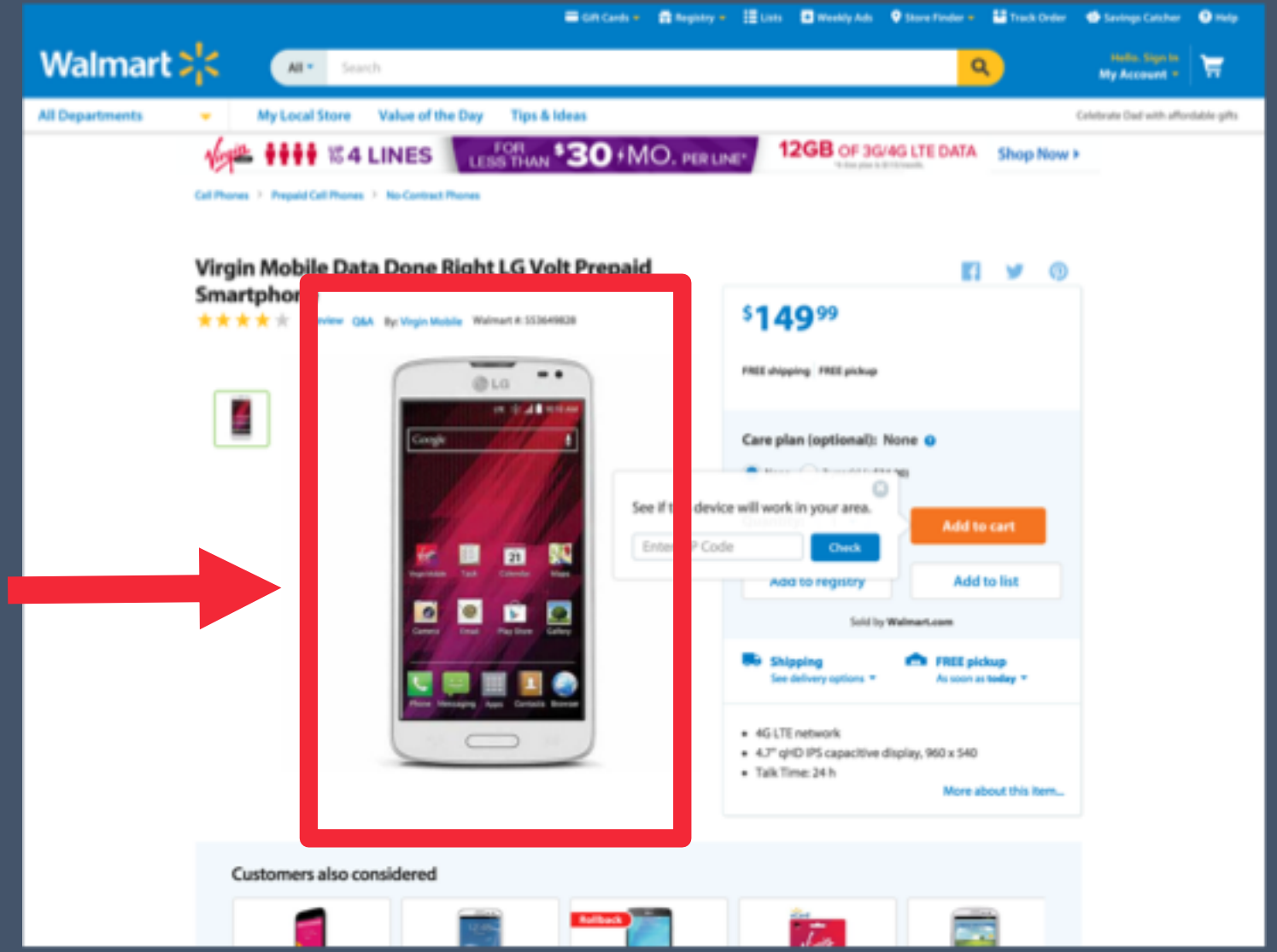
No changes were made since the previous publication.

By publishing this Recommendation, W3C expects that the functionality specified in this User Timing Recommendation will not be affected by changes to Web IDL as this specification proceed to Recommendation.

Please send comments to public-web-perf@w3.org (archived) with `[UserTiming]` at the start of the subject line.

This document has been reviewed by W3C Members, by software developers, and by other W3C groups and interested parties, and is endorsed by the Director as a W3C Recommendation. It is a stable document and may be used as reference material or cited from another document. W3C's role in making the Recommendation

How long does it take to display the main product image on my site?



```
function loadTemplate() {
  performance.mark(“startTask”);

  doTemplateStuff() //do stuff (assumes promise())
    .then(() => {
       performance.mark(“stopTask”);
       //Measure the duration between the two marks
       performance.measure(“taskDuration”,
                           “startTask”,“stopTask”);
    });
}
```

```
<img src="image1.gif"
      onload="performance.mark('hero_image')">
```

https://stevesouders.com/blog/2015/05/12/hero-image-custom-metrics/

W3C Working Draft

TABLE OF CONTENTS

Server Timing

W3C Working Draft 29 December 2017

This version:
https://www.w3.org/TR/2017/WD-server-timing-20171229/

Latest published version:
https://www.w3.org/TR/server-timing/

Latest editor's draft:
https://w3c.github.io/server-timing/

Previous version:
https://www.w3.org/TR/2017/WD-server-timing-20171221/

Editors:
Charles Vazac, Akamai, cvazac@gmail.com
Ilya Grigorik, Google, igrigorik@gmail.com

Participate:
GitHub w3c/server-timing
File a bug
Commit history

Mailing list:
public-web-perf@w3.org

Implementation:
Test Suite
Test Suite repository

Abstract

This specification enables a server to communicate performance metrics about the request-response cycle to the user agent. It also standardizes a JavaScript interface to enable applications to collect, process, and act on these metrics to optimize application delivery.

Status of This Document

```
GET /resource HTTP/1.1
Host: example.com

HTTP/1.1 200 OK
Server-Timing: cdn, dur="130"
Server-Timing: cloudinary, dur="10", desc="new trans"
```

```
{
  let url = 'https://example.com/resource.jpg’;
  let severEntries[] =
      window.performance.getEntriesByName(url)
       .filter((name) => name === "server");
}
```

W3C Editor's Draft

TABLE OF CONTENTS

Paint Timing 1

Editor's Draft, 9 January 2018

This version:
https://w3c.github.io/paint-timing/

Latest published version:
https://www.w3.org/TR/paint-timing/

Issue Tracking:
GitHub

Editor:
Shubhie Panicker (Google)

Abstract

This document defines an API that can be used to capture a series of key moments (First Paint, First Contentful Paint) during pageload which developers care about.

Status of this document

This section describes the status of this document at the time of its publication. Other documents may supersede this document. A list of current W3C publications and the latest revision of this technical report can be found in the W3C technical reports index at https://www.w3.org/TR/.

This document was published by the Web Performance Working Group as an Editors Draft. This document is intended to become a W3C Recommendation.

Feedback and comments on this specification are welcome, please send them to public-web-perf@w3.org (subscribe, archives) with `%5Bpaint-timing%5D` at the start of your email's subject.

Publication as an Editors Draft does not imply endorsement by the W3C Membership. This is a draft document and may be updated, replaced or obsoleted by other documents at any time. It is inappropriate to cite this document as other than work in progress.

This document was produced by a group operating under the 5 February 2004 W3C Patent Policy. W3C

W3C Editor's Draft

TABLE OF CONTENTS

Long Tasks API 1

Editor's Draft, 9 January 2018

This version:
https://w3c.github.io/longtasks/

Test Suite:
http://w3c-test.org/longtask-timing/

Issue Tracking:
GitHub

Editors:
Shubhie Panicker (Google)
Ilya Grigorik (Google)
Domenic Denicola (Google)

Abstract

This document defines an API that web page authors can use to detect presence of "long tasks" that monopolize the UI thread for extended periods of time and block other critical tasks from being executed - e.g. reacting to user input.

Status of this document

This section describes the status of this document at the time of its publication. Other documents may supersede this document. A list of current W3C publications and the latest revision of this technical report can be found in the W3C technical reports index at https://www.w3.org/TR/.

This document was published by the Web Performance Working Group as an Editors Draft. This document is intended to become a W3C Recommendation.

If you wish to make comments regarding this document, please send them to public-web-perf@w3.org (subscribe, archives) with `[LongTasks]` at the start of your email's subject.

Feedback and comments on this specification are welcome, please send them to public-web-perf@w3.org

§ 4.3.1. Interface `MutationObserver`

IDL

```
[Constructor(MutationCallback callback),
 Exposed=Window]
interface MutationObserver {
  void observe(Node target, optional MutationObserverInit options);
  void disconnect();
  sequence<MutationRecord> takeRecords();
};

callback MutationCallback = void (sequence<MutationRecord> mutations, MutationObserver observer);

dictionary MutationObserverInit {
  boolean childList = false;
  boolean attributes;
  boolean characterData;
  boolean subtree = false;
  boolean attributeOldValue;
  boolean characterDataOldValue;
  sequence<DOMString> attributeFilter;
};
```

A `MutationObserver` object can be used to observe mutations to the tree of nodes.

Each `MutationObserver` object has these associated concepts:

- A **callback** set on creation.
- A **node list** (a list of nodes), which is initially empty.
- A **record queue** (a queue of zero or more `MutationRecord` objects), which is initially empty.

For web developers (non-normative)

***observer* = new MutationObserver(callback)**

Constructs a `MutationObserver` object and sets its callback to *callback*. The *callback* is invoked with a list of `MutationRecord` objects as first argument and the constructed `MutationObserver` object as second argument. It is invoked after nodes registered with the `observe()` method, are mutated.

```
let config = { attributes: true, childList: true, subtree: true };

let targetNode = document.querySelector('body');

let callback = function(mutations) {
 mutations.forEach(function(mutation) {
    if (mutation.type === 'childList') {
      let values = [].slice.call(list.children)
                     .map(node => node.nodeName)
                     .filter(n => /img/i.test(n));

      console.log(list_values);
    }
  });
};

let observer = new MutationObserver(callback);

observer.observe(targetNode, config);
```

W3C Candidate Recommendation

TABLE OF CONTENTS

Beacon

W3C Candidate Recommendation 13 April 2017

This version:
https://www.w3.org/TR/2017/CR-beacon-20170413/

Latest published version:
https://www.w3.org/TR/beacon/

Latest editor's draft:
https://w3c.github.io/beacon/

Previous version:
https://www.w3.org/TR/2016/WD-beacon-20161028/

Editors:
Ilya Grigorik, Google, igrigorik@gmail.com
Alois Reitbauer, Compuware Corp., alois.reitbauer@compuware.com
Arvind Jain, Google Inc., arvind@google.com (Until January 2015)
Jatinder Mann, Microsoft Corp., jmann@microsoft.com (Until February 2014)

Repository:
We are on Github.
File a bug.
Commit history.

Mailing list:
public-web-perf@w3.org

Test Suite:
Test Suite

Implementation:
Implementation report
Can I use Beacon?

Abstract

This specification defines an interface that web developers can use to schedule asynchronous and non-blocking delivery of data that minimizes resource contention with other time-critical operations, while ensuring that such requests are still processed and delivered to destination.

Exercise 4: performance budget

- Navigate to amazon.com
- Calculate the total bytes downloaded
- Ensure that the page load time < 2s and bytes < 2MiB
- Repeat with Disney.com

Take Action

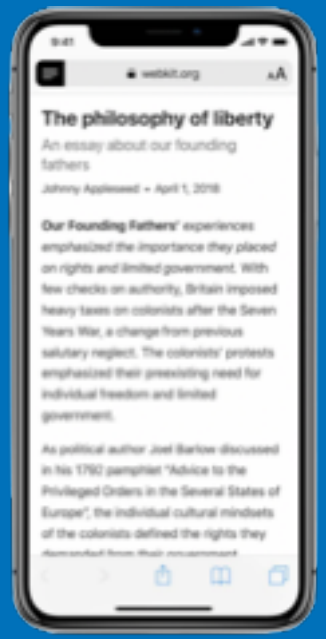
- Add performance.mark() to measure YOUR key metrics
  - Eg: Hero Image, Product Image, Sport Scores
- Keep performance.mark() in production code!
- Define Hard and Soft targets that synthetic tests should hit
- BHAG: 2s & 1MiB

Mobile

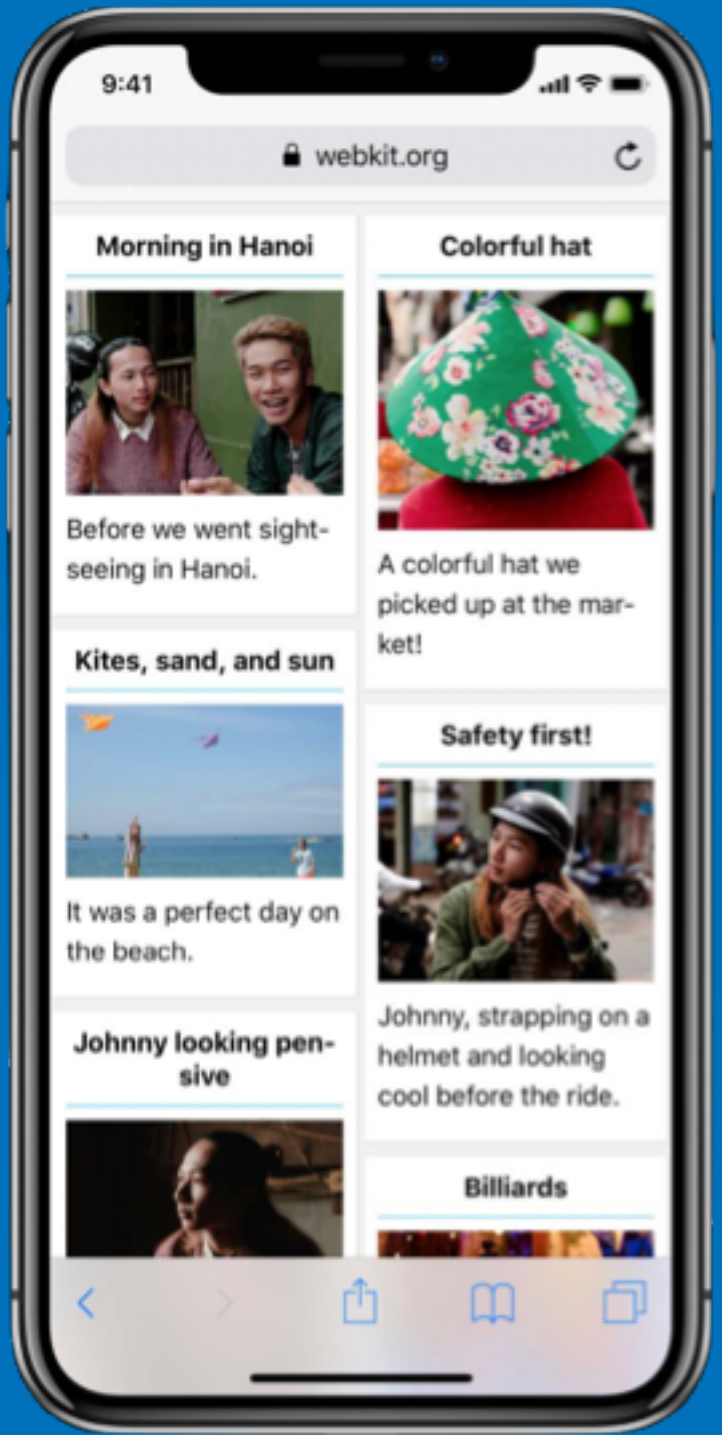
Display & Network Throttling

Tim Kadlec

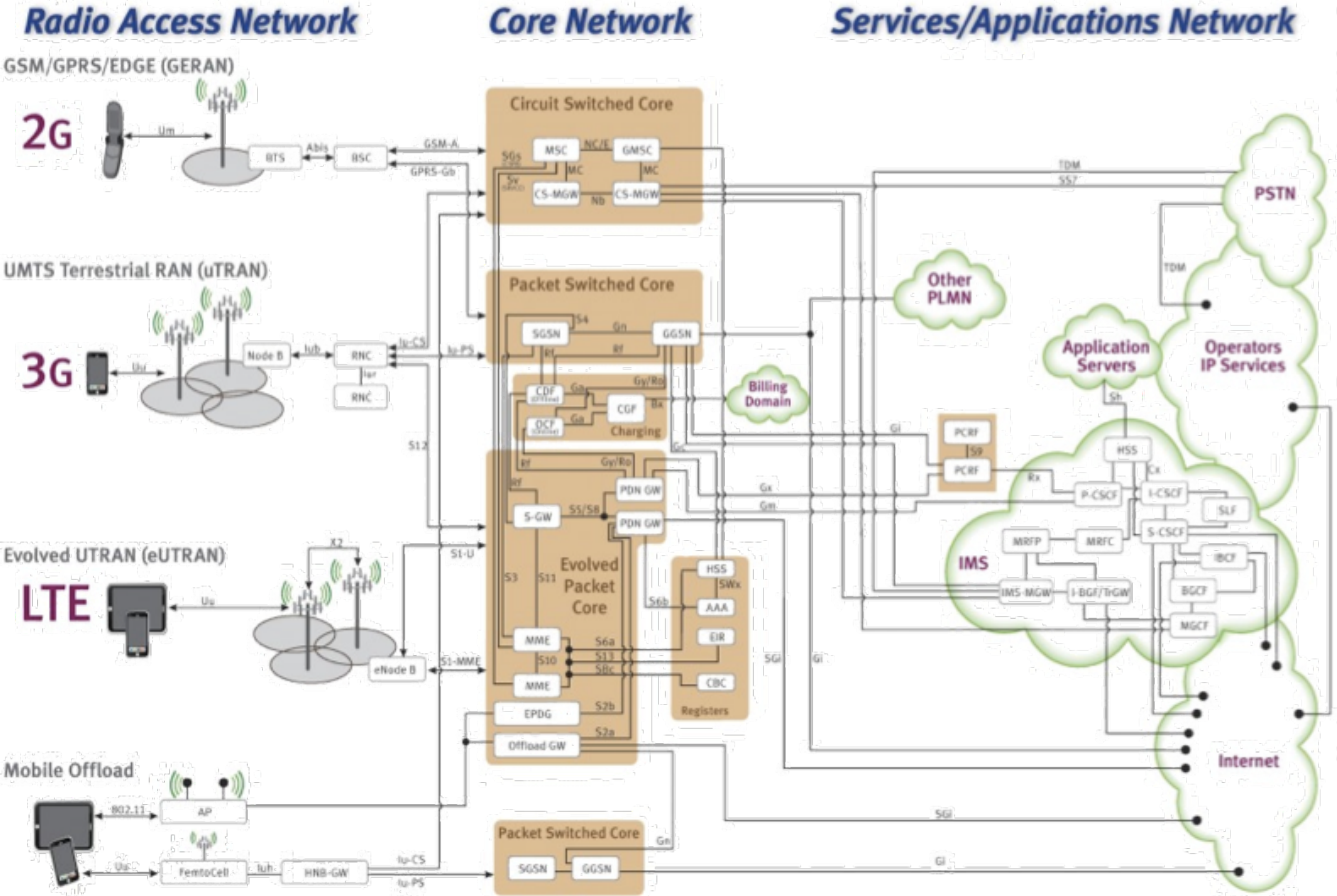

The philosophy of liberty
An essay about our founding fathers
Johnny Appleseed • April 1, 2018
Our Founding Fathers' experiences emphasized the importance they placed on rights and limited government. With few checks on authority, Britain imposed heavy taxes on colonists after the Seven Years War, a change from previous salutary neglect. The colonists' protests emphasized their preexisting need for individual freedom and limited government.
As political author Joel Barlow discussed in his 1792 pamphlet "Advice to the Privileged Orders in the Several States of Europe", the individual cultural mindsets of the colonists defined the rights they demanded from their government. Colonial experiences did not change the Founders' views as much as it legitimized their belief in individual rights already established during times of salutary neglect.
MacBook Pro

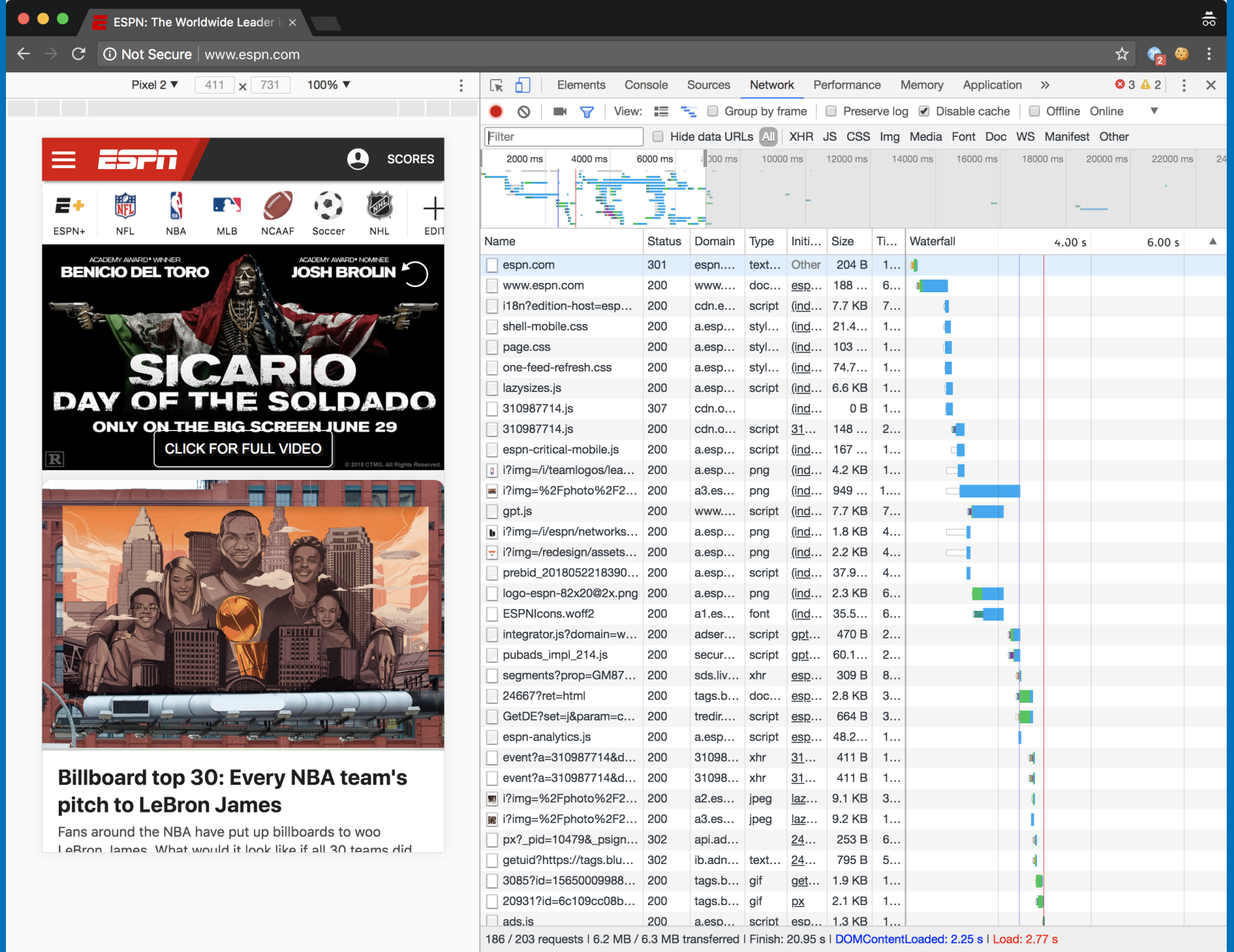

webkit.org
The philosophy of liberty
An essay about our founding fathers
Johnny Appleseed • April 1, 2018
Our Founding Fathers' experiences emphasized the importance they placed on rights and limited government. With few checks on authority, Britain imposed heavy taxes on colonists after the Seven Years War, a change from previous salutary neglect. The colonists' protests emphasized their preexisting need for individual freedom and limited government.
As political author Joel Barlow discussed in his 1792 pamphlet "Advice to the Privileged Orders in the Several States of Europe", the individual cultural mindsets of the colonists defined the rights they


Johnny
10:09
The philosophy of liberty
webkit.org


9:41
webkit.org
Morning in Hanoi
Before we went sight-seeing in Hanoi.
Kites, sand, and sun
It was a perfect day on the beach.
Johnny looking pen-sive
Colorful hat
A colorful hat we picked up at the mar-ket!
Safety first!
Johnny, strapping on a helmet and looking cool before the ride.
Billiards


Johnny
10:09
Johnny Appleseed's Blog
webkit.org


Radio Access Network
Core Network
Services/Applications Network
GSM/GPRS/EDGE (GERAN)
2G
Um
BTS
Abis
BSC
GSM-A
GPRS-Gb
Circuit Switched Core
MSC
NC/E
GMSC
MC
CS-MGW
Nb
SGs
Sv
UMTS Terrestrial RAN (uTRAN)
3G
Uu
Node B
Iub
RNC
Iur
Iu-CS
Iu-PS
Packet Switched Core
SGSN
S4
Gn
Rf
GGSN
CDF
Ga
Gy/Ro
CGF
Bx
OCF
Charging
S12
Evolved UTRAN (eUTRAN)
LTE
X2
S1-U
eNode B
S1-MME
S-GW
S5/S8
PDN GW
S3
S11
Evolved Packet Core
MME
S6a
S10
S13
SBc
EPDG
S2b
Offload GW
S2a
HSS
SWx
S6b
AAA
EIR
CBC
Registers
Billing Domain
Other PLMN
Gi
PCRF
S9
Gx
Gm
SGi
Mobile Offload
802.11
AP
FemtoCell
Iuh
HNB-GW
Iu-CS
Iu-PS
Packet Switched Core
SGSN
GGSN
TDM
SS7
PSTN
Application Servers
Operators IP Services
Sh
Rx
Cx
P-CSCF
I-CSCF
SLF
S-CSCF
MRFP
MRFC
IMS
IBCF
BGCF
IMS-MGW
I-BGF/TrGW
MGCF
Internet


ESPN: The Worldwide Leader
Not Secure | www.espn.com
Pixel 2 411 x 731 100%
ESPN
SCORES
ESPN+ NFL NBA MLB NCAAF Soccer NHL EDIT
ACADEMY AWARD® WINNER BENICIO DEL TORO
ACADEMY AWARD® NOMINEE JOSH BROLIN
SICARIO
DAY OF THE SOLDADO
ONLY ON THE BIG SCREEN JUNE 29
CLICK FOR FULL VIDEO
Billboard top 30: Every NBA team's pitch to LeBron James
Fans around the NBA have put up billboards to woo
Elements Console Sources Network Performance Memory Application
Filter Hide data URLs All XHR JS CSS Img Media Font Doc WS Manifest Other
Name Status Domain Type Initi... Size Ti... Waterfall 4.00 s 6.00 s
espn.com 301 espn.... text... Other 204 B 1...
www.espn.com 200 www.... doc... esp... 188 ... 6...
i18n?edition-host=esp... 200 cdn.e... script (ind... 7.7 KB 7...
shell-mobile.css 200 a.esp... styl... (ind... 21.4... 1...
page.css 200 a.esp... styl... (ind... 103 ... 1...
one-feed-refresh.css 200 a.esp... styl... (ind... 74.7... 1...
lazysizes.js 200 a.esp... script (ind... 6.6 KB 1...
310987714.js 307 cdn.o... (ind... 0 B 1...
310987714.js 200 cdn.o... script 31... 148 ... 2...
espn-critical-mobile.js 200 a.esp... script (ind... 167 ... 1...
i?img=/i/teamlogos/lea... 200 a.esp... png (ind... 4.2 KB 1...
i?img=%2Fphoto%2F2... 200 a3.es... png (ind... 949 ... 1....
gpt.js 200 www.... script (ind... 7.7 KB 7...
i?img=/i/espn/networks... 200 a.esp... png (ind... 1.8 KB 4...
i?img=/redesign/assets... 200 a.esp... png (ind... 2.2 KB 4...
prebid_2018052218390... 200 a.esp... script (ind... 37.9... 4...
logo-espn-82x20@2x.png 200 a.esp... png (ind... 2.3 KB 6...
ESPNIcons.woff2 200 a1.es... font (ind... 35.5... 6...
integrator.js?domain=w... 200 adser... script gpt... 470 B 2...
pubads_impl_214.js 200 secur... script gpt... 60.1... 2...
segments?prop=GM87... 200 sds.liv... xhr esp... 309 B 8...
24667?ret=html 200 tags.b... doc... esp... 2.8 KB 3...
GetDE?set=j&param=c... 200 tredir.... script esp... 664 B 3...
espn-analytics.js 200 a.esp... script esp... 48.2... 1...
event?a=310987714&d... 200 31098... xhr 31... 411 B 1...
event?a=310987714&d... 200 31098... xhr 31... 411 B 1...
i?img=%2Fphoto%2F2... 200 a2.es... jpeg laz... 9.1 KB 3...
i?img=%2Fphoto%2F2... 200 a3.es... jpeg laz... 9.2 KB 1...
px?_pid=10479&_psign... 302 api.ad... 24... 253 B 6...
getuid?https://tags.blu... 302 ib.adn... text... 24... 795 B 5...
3085?id=15650009988... 200 tags.b... gif get... 1.9 KB 1...
20931?id=6c109cc08b... 200 tags.b... gif px 2.1 KB 1...
ads.is 200 a.esp... script esp... 1.3 KB 1...
186 / 203 requests | 6.2 MB / 6.3 MB transferred | Finish: 20.95 s | DOMContentLoaded: 2.25 s | Load: 2.77 s

```
chromeOptions: {
    args:["--incognito", "--headless", "--disable-gpu"]
},
```

```
chromeOptions: {
    args:["--incognito", "--disable-gpu"],
    mobileEmulation: {
        deviceMetrics: {
            width: 360,
            height: 720,
            pixelRatio: 3},
        networkThrottle: 'Fast 3G'
    }
}
```

whatdoesmysitecost.com

What Does My Site Cost?

Find out how much it costs for someone to use your site on mobile networks around the world.

WebPageTest Test ID

GRAB THE RESULTS

OR

Site URL

TEST MY SITE

According to the May 15, 2018 run of HTTP Archive, the median site now weighs 1720kb. Here's the approximate cost around the world:

Cost in USD | Cost in USD, PPP | Cost as % of GNI, PPP

Postpaid | Prepaid

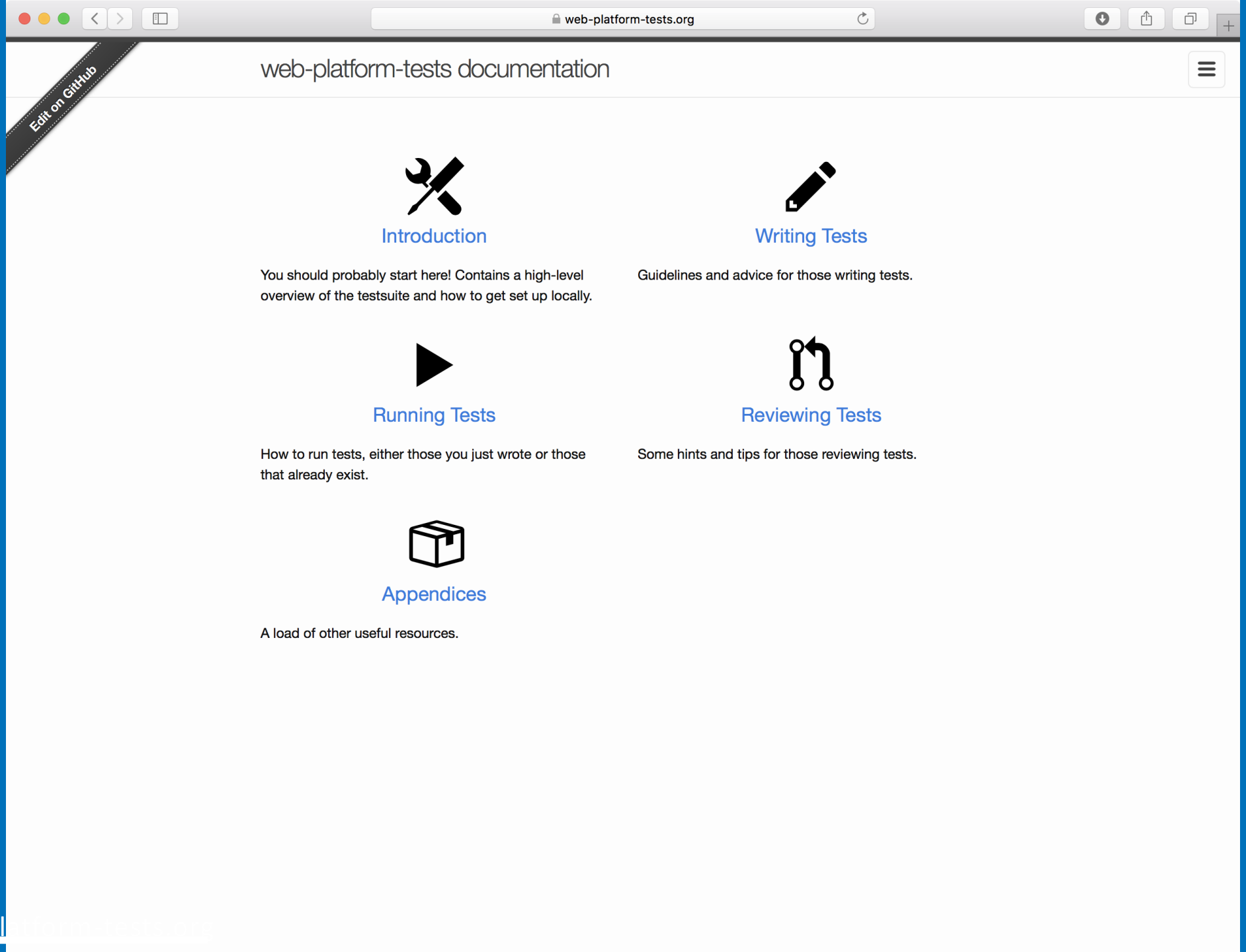
Cost in USD (POSTPAID DATA)

This is the cost of the site based on data from the ITU, without any adjustment for purchasing power or relative affordability. Prices were collected from the operator with the largest marketshare in the country, using the least expensive plan with a (minimum) data allowance of 500 MB over (a minimum of) 30 days. Prices include taxes. Because these numbers are based on the least expensive plan, they are **best case scenarios**.

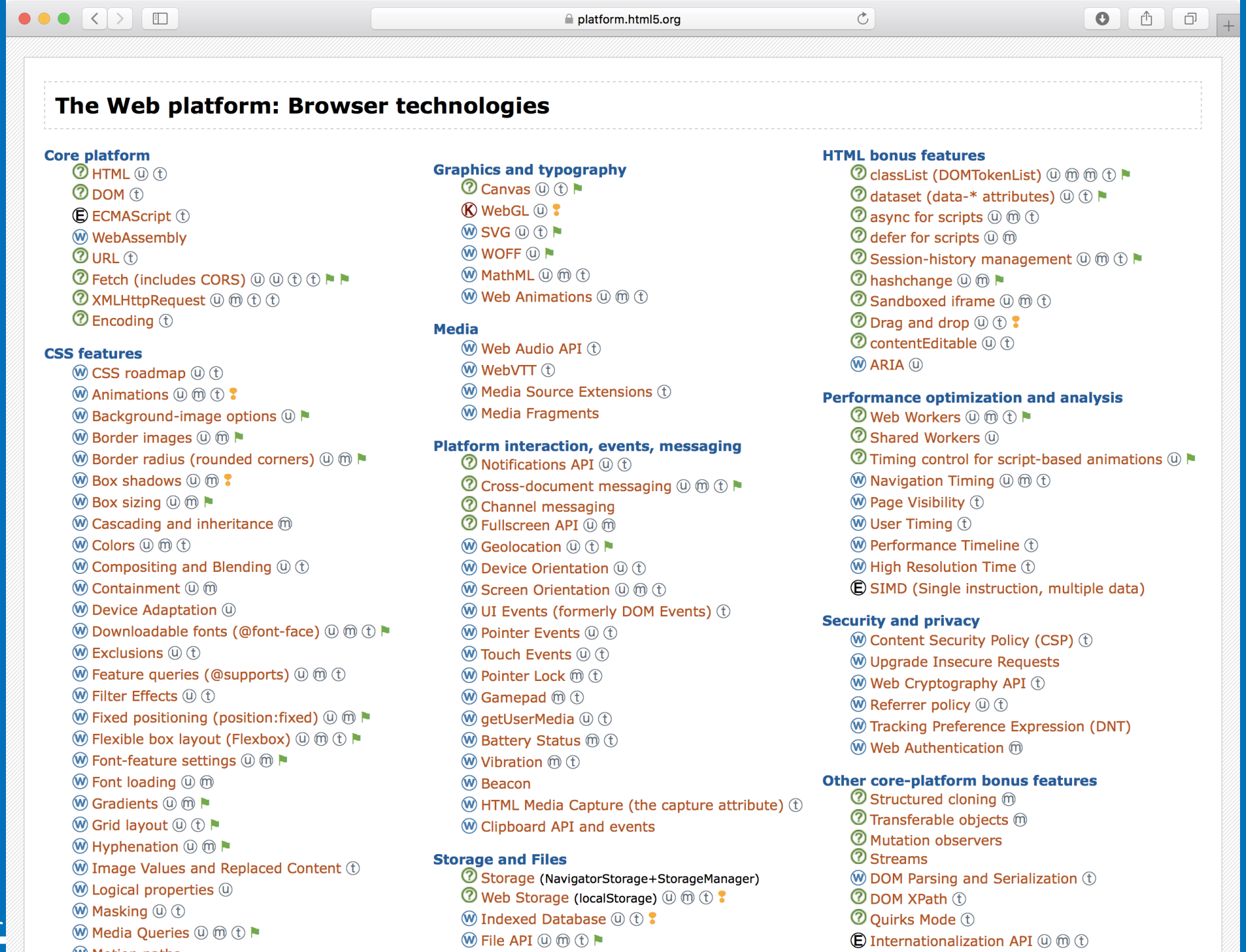
Take Action

- Run Tests in different viewport configurations to test RWD
- Experiment with different Network Conditions
- Similarly with different CPU throttling

Web Platform Test


web-platform-tests.org
Edit on GitHub
web-platform-tests documentation
Introduction
You should probably start here! Contains a high-level overview of the testsuite and how to get set up locally.
Writing Tests
Guidelines and advice for those writing tests.
Running Tests
How to run tests, either those you just wrote or those that already exist.
Reviewing Tests
Some hints and tips for those reviewing tests.
Appendices
A load of other useful resources.

https://web-pl



https://platfor

they may be using internally.

If you wish the HTML page into which your tests are being run to render results in a nice and convenient table, you should include an HTML element with ID `log` where you would like those results to appear. See the bottom of this document for an example.

Basic Testing

The most basic usage relies on the `test()` function, which takes a function with the code to run, a name of the test, and optionally a literal object providing some additional options.

```
test(function () {
    assert_true(true);
}, "True really is true");
```

The given function *must* include at least one assertion; conversely, assertions can only appear in the context of a test function. A single test may contain multiple assertions, in which case it is considered to be atomic. That is to say that a single failed assertion will fail the test, whereas all are required to pass for the whole test to pass. A document can contain as many tests as you wish it to.

The example here contains two assertions, both of which pass.

```
test(function () {
    assert_true(true);
    assert_false(false);
}, "Truth is what you believe it to be");
```

But in this example one passes, while the other fails. This causes the entire test to be reported as a failure.

```
test(function () {
    assert_true(true);
    assert_false(true);
}, "All opinions are equally valid.");
```

In addition to a function and a name, `test()` can also accept a third parameter being a dictionary of options. Most of those options are documented in the Including Metadata section below.

The only general-purpose option that you can use is `timeout`. It takes a number of milliseconds (defaulting to 1000, which the brightest at maths amongst you will have recognised as being equivalent to one second). If the content of the test takes longer than `timeout` to run then the test is aborted and counted as a fail. Since some processing can take longer than one second to run, especially if you are performing a complex test in a low-end environment (e.g. a basic mobile phone) it can at times be useful to increase this limit as exemplified here.

```
test(function () {
    /* do something long and slow here */
    assert_true(true);
}, "Long operation is successful", { timeout: 5000 });
```

Why Inside v. Outside Browser Testing?

- Hawthorne Effect
- Platform Widgets & Controls
- APIs

Exercise 5: WPT

Implement Responsive Images and Test the naturalWidth

A11y

with AXE

Léonie Watson

Marcy Sutton

Katie Sylor-Miller

ACCESSIBILITY (a11y)

"Websites, tools, and technologies are designed and developed so that people with disabilities can use them."

- W3C Web Accessibility Initiative

THE AMERICANS WITH DISABILITIES ACT (ADA)

"No individual shall be discriminated against on the basis of disability in the full and equal enjoyment of the goods, services, facilities, privileges, advantages, or accommodations of any place of *public accommodation*"

- ADA Sec. 12182 (a)

57 M
19%

AMERICANS WITH A DISABILITY

UNITED STATES CENSUS REPORT, JULY 2012

The Basics

- Web Content Accessibility Guidelines (WCAG 2.0)
- Developed by the W3C Web Accessibility Three levels of compliance - A, AA, AAA

www.axe-core.org

deque

THE ACCESSIBILITY ENGINE axe

- Playground
- Integrations
- Documentation
- About
- Future tools
- Community

Fork me on GitHub

Easy Accessibility Testing with aXe

How do I get started?

The quickest way to pick up the aXe, is by using the extensions, available for Chrome and Firefox. For developers, we recommend installing axe-core directly and using it in your projects.

aXe Browser Extensions

aXe for Chrome

aXe for Firefox 54+

Using Firefox ~53, or NVDA with Firefox? Install our legacy addon.

How easy is aXe to use?

To download the module, make sure you have Node.js installed and are sitting in the root directory of your project, and execute the following command:

```
npm install axe-core --save-dev
```

dequelabs / axe-core

Watch 110 Star 1,565 Fork 219

Code Issues 125 Pull requests 6 Projects 3 Insights

Accessibility engine for automated Web UI testing https://www.axe-core.org/

2,898 commits 22 branches 41 releases 71 contributors MPL-2.0

Branch: develop New pull request Find file Clone or download

stephenmathieson and WilcoFiers fix(axe.run): allow returning a Promise in jsdom (#943) ... Latest commit 3858a1f 7 days ago

.github	chore: Further markdown linting (#848)	a month ago
build	fix: attempt to fix ci build failures. (#917)	17 days ago
doc	fix: Update tags for frame-title rule (#935)	10 days ago
lib	fix(axe.run): allow returning a Promise in jsdom (#943)	7 days ago
locales	feat: Translated rules help and checks fail into German (#877)	28 days ago
test	fix: Right trim URLs before outputting them in getSelector (#924)	10 days ago
typings/axe-core	fix(typescript): Update ElementContext type (#822)	2 months ago
.babelrc	Babelify Axe	2 years ago
.editorconfig	chore: Further markdown linting (#848)	a month ago
.eslintrc	chore: Switch to eslint from jshint	3 months ago
.gitattributes	add .gitattributes to master	a year ago
.gitignore	fix: attempt to fix ci build failures. (#917)	17 days ago
.jshintrc	update to master	2 years ago
.markdownlint.json	chore: Further markdown linting (#848)	a month ago
.npmignore	reduce .npmignore to what is required to build the develop branch	a year ago
.retireignore.json	chore: add retire dependency to retireignore	2 months ago
CHANGELOG.md	chore(release): 3.0.3	10 days ago

Exercise 6: A11y

- Add axe-core npm
- Load 3 news websites
- Compare the violations
- Create a test case to fail on a11y violations

Take Action

- Good Usability === Good Accessibility
- Use Semantic HTML (also great for SEO)
- Add `alt` and `description` tags to media and anchors
- Test your site (axe-core.com)
- Try the Funkity-Disability-Simulator Chrome Extension https://chrome.google.com/webstore/detail/funkify-disability-simula/ojcijjdchelkddboickefhnbdpeajdjg?hl=en
- Be an ally

DNS

Take Action

Parallization

Take Action

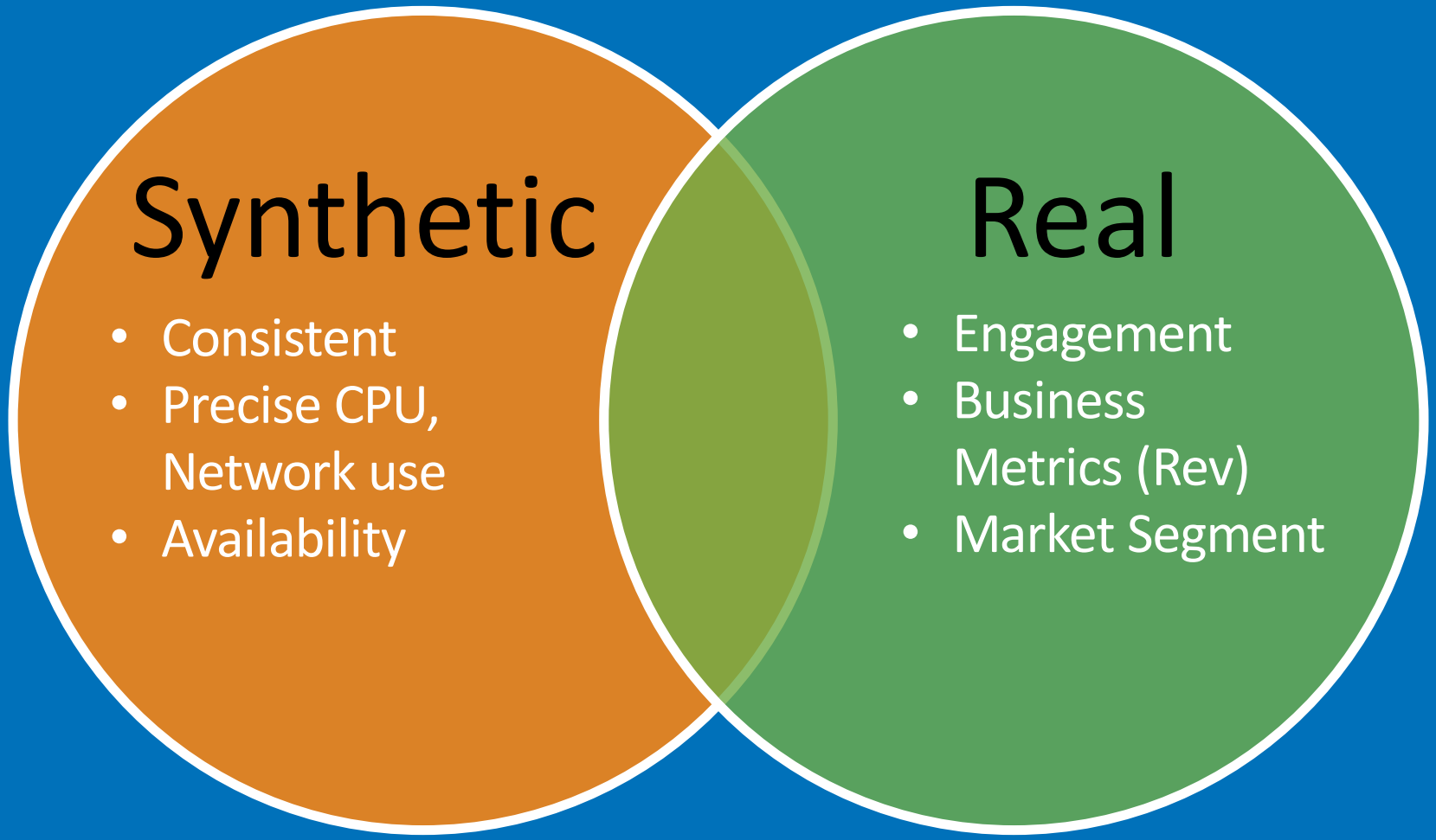
CI/CD/CB

with Jenkins

Take Action

Real User Monitoring

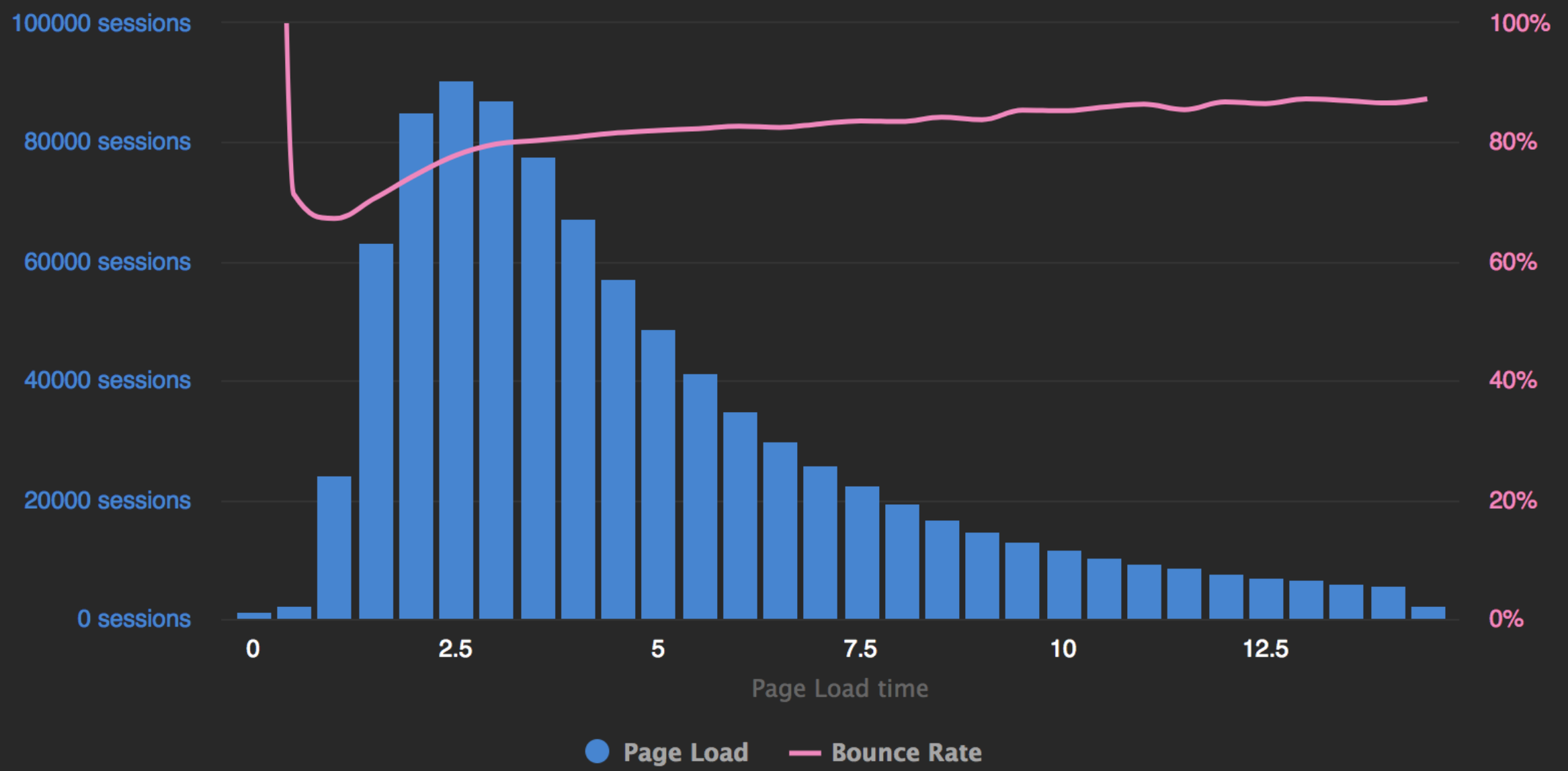
with Lux



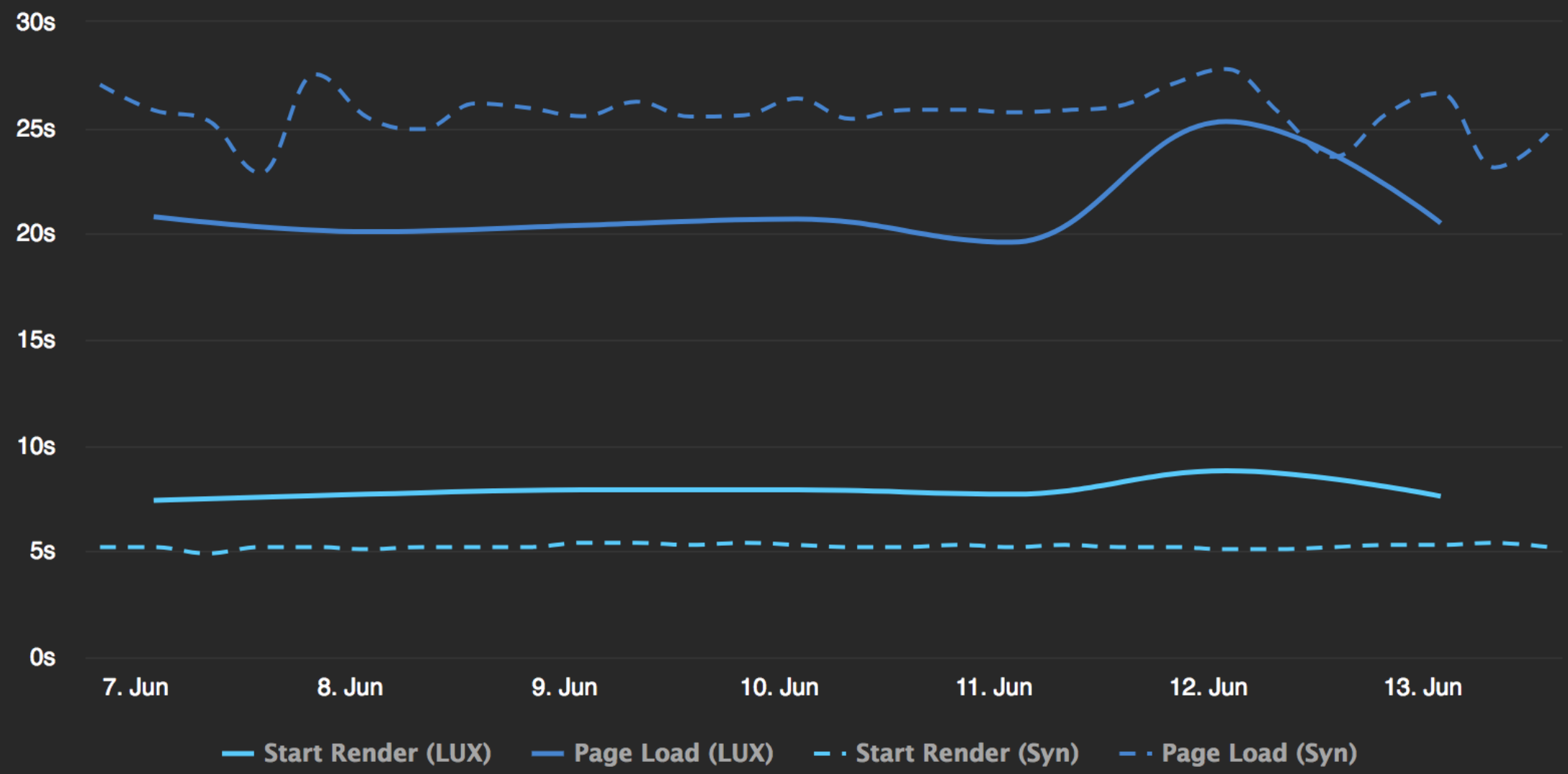
- Layout
- Design

LOAD TIME VS BOUNCE RATE

Notes

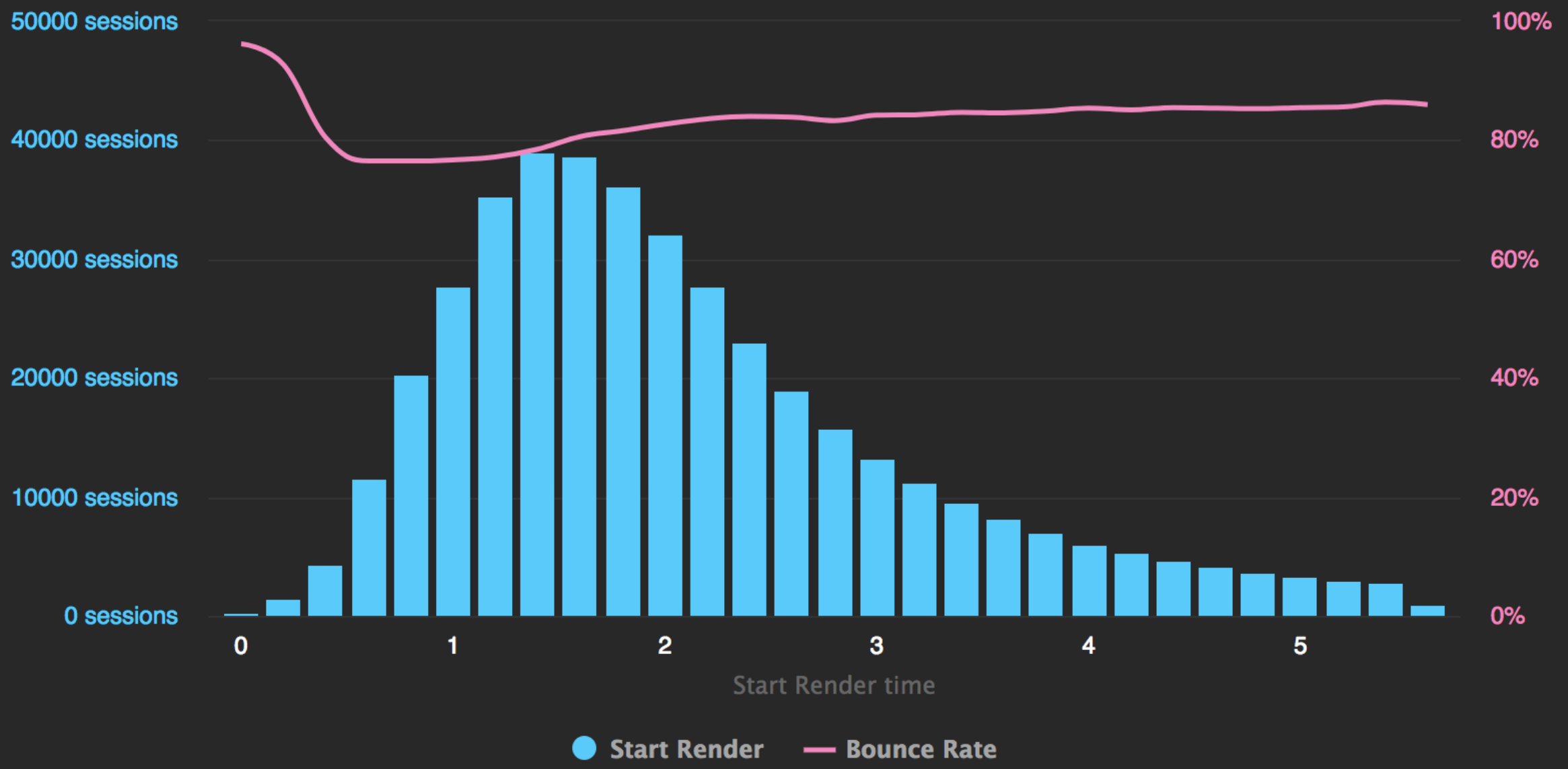


Start Render (LUX)	Page Load (LUX)	Start Render (Syn)	Page Load (Syn)
7.7s	20.5s	5.2s	25.74s



START RENDER VS BOUNCE RATE

Notes



Take Action

- Collect Real-User browser exerpiences
- Audit user experience (load time, downloaded content)
- Periodically run tests in the wild
- Add window.addEventListener('error') to capture console errors
- Use Beacon API to ensure metrics are returned

Summary

- Browser testing is easy!
- Synthetic testing with:
  - WebDriver.io for easy functional testing
  - WebPageTest.org to augment performance testing
- Use WebPlatformTest for direct platform tests
- Don’t just test workflows: Design, Performance, and Accessibility
- Use Real-User feedback to compare tests in the wild


@ColinBendell

Further Reading

- http://webdriver.io/api.html
- https://github.com/cloudinary/wdio-allure-ts
- https://github.com/cloudinary/wdio-allure-ts-example
- https://www.npmjs.com/package/webdriver-manager
- http://blog.kevinlamping.com/testing-your-login-an-in-depth-webdriverio-tutorial/
- https://www.browserstack.com/automate/webdriverio
- https://medium.com/@boriscoder/setting-up-selenium-tests-with-webdriver-io-cc7fc3c86629
- https://w3c.github.io/webdriver/
- https://www.slideshare.net/LinkMeSrl/webdriverio?qid=8cdf045d-9868-4ca8-a8ad-8aefb5c22289&v=&b=&from_search=1
- https://peter.sh/experiments/chromium-command-line-switches/#load-extension
- https://www.ghacks.net/2013/10/06/list-useful-google-chrome-command-line-switches/
- https://deliciousbrains.com/how-were-automating-acceptance-testing/
- https://medium.com/tech-tajawal/page-object-model-pom-design-pattern-f9588630800b

Agenda

- Test Methodologies
- WebDriver.io
- Mocha tests
- How Browsers Work & Browser APIs
- Web Platform Test
- Mobile & Device Emulation
- Accessibility
- Operational: DNS, Paralyzing, CI
- Real User Testing

9:00 AM Welcome | **10:30 AM** Break | **12:00 AM** Lunch | **2:30 PM** Break | **4:00 PM** Wrap-up & Q&A

Break

10:30 - 10:45 AM

Lunch

12:00 PM – 1:00PM

Break

2:30 – 2:45 PM