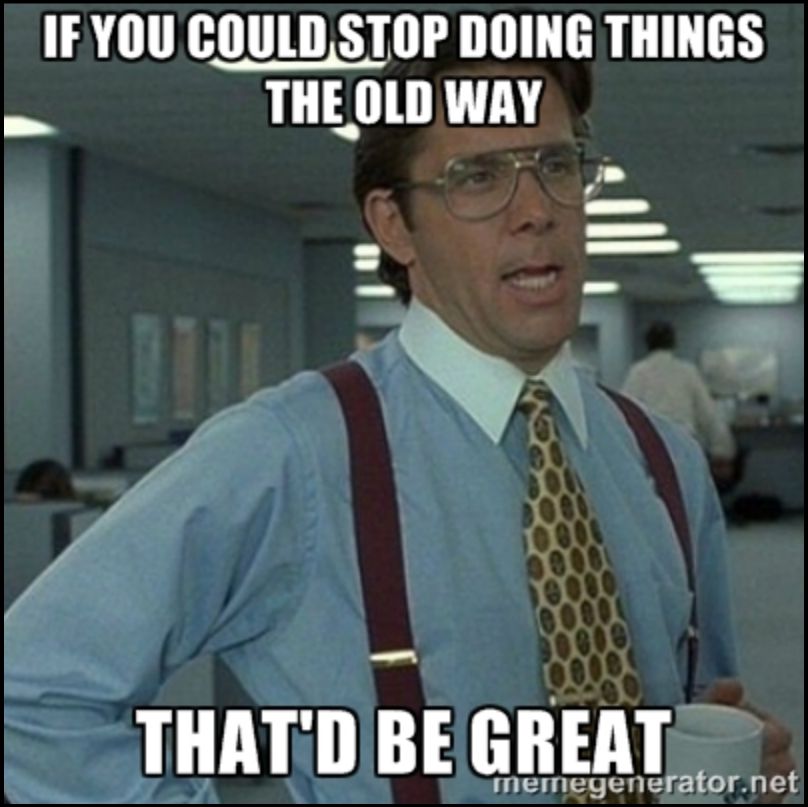
Automating Web Performance Testing with PhantomJS 2

Wesley Hales
Shape Security
@wesleyhales

Measuring Page Load Time



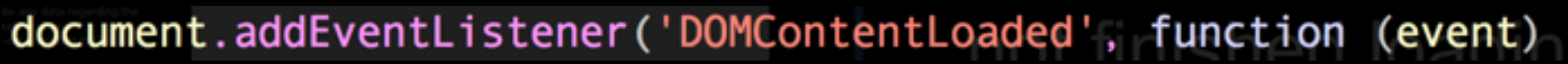
Date.now() || Date().getTime()

- When was the page loaded?
- Add an event listener
- Get the current time
- Profit
- Example: Simple.html

However...

- JavaScript time is notoriously inaccurate
- It is skewed by adjustments to the system clock
- it can’t provide any data regarding the server, network, and so on.
- Example: all-old.html (add HRT)

DOMContentLoaded

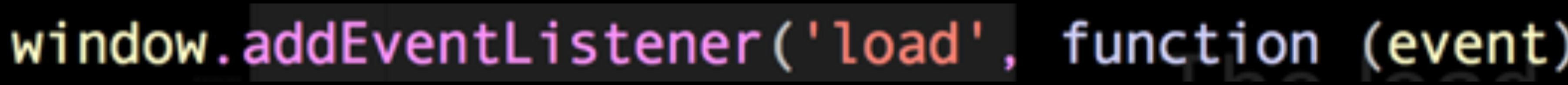


```
document.addEventListener('DOMContentLoaded', function (event)
```

- document has been completely loaded and parsed.
- stylesheets, images, and subframes have not finished loading

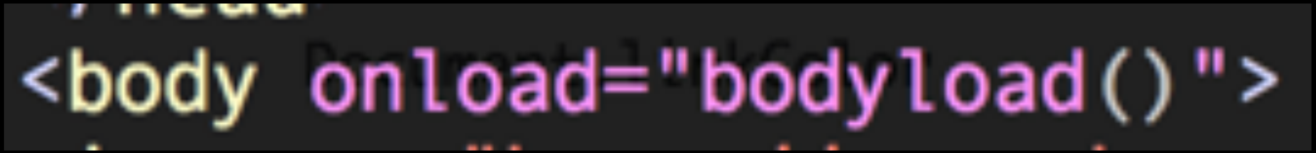
@wesleyhales

load || onload



```
window.addEventListener('load', function (event)
```

The load event is fired when a resource and its dependent resources have finished loading.

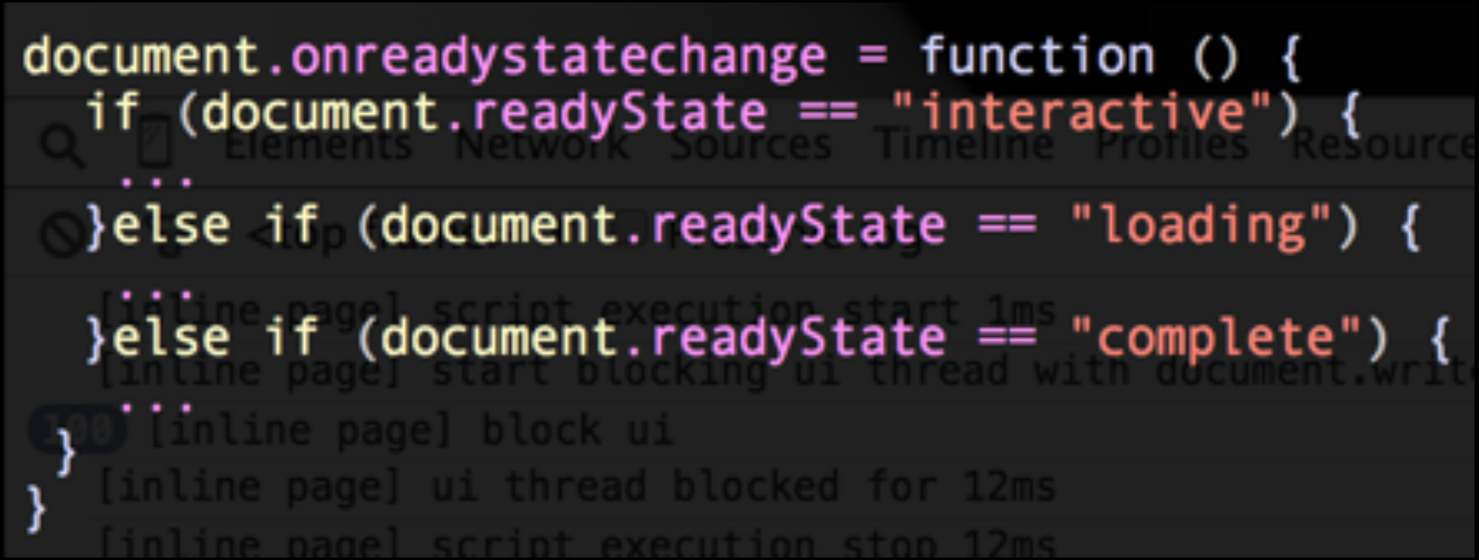


```
<body onload="bodyload()">
```

@wesleyhales

readyState

- "loading" while the document is loading
- "interactive" once it is finished parsing (but still loading sub-resources)
- "complete" once it has loaded.



```
document.onreadystatechange = function () {
  if (document.readyState == "interactive") {
    ...
  }else if (document.readyState == "loading") {
    ...
  }else if (document.readyState == "complete") {
    ...
  }
}
```

```
[inline page] script execution start 1ms
[inline page] start blocking ui thread with document.writ
[inline page] block ui
[inline page] ui thread blocked for 12ms
[inline page] script execution stop 12ms
```

@wesleyhales

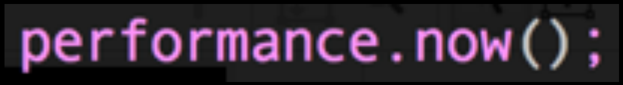
Demo

Basic loading and blocking

Enter Navigation Timing API

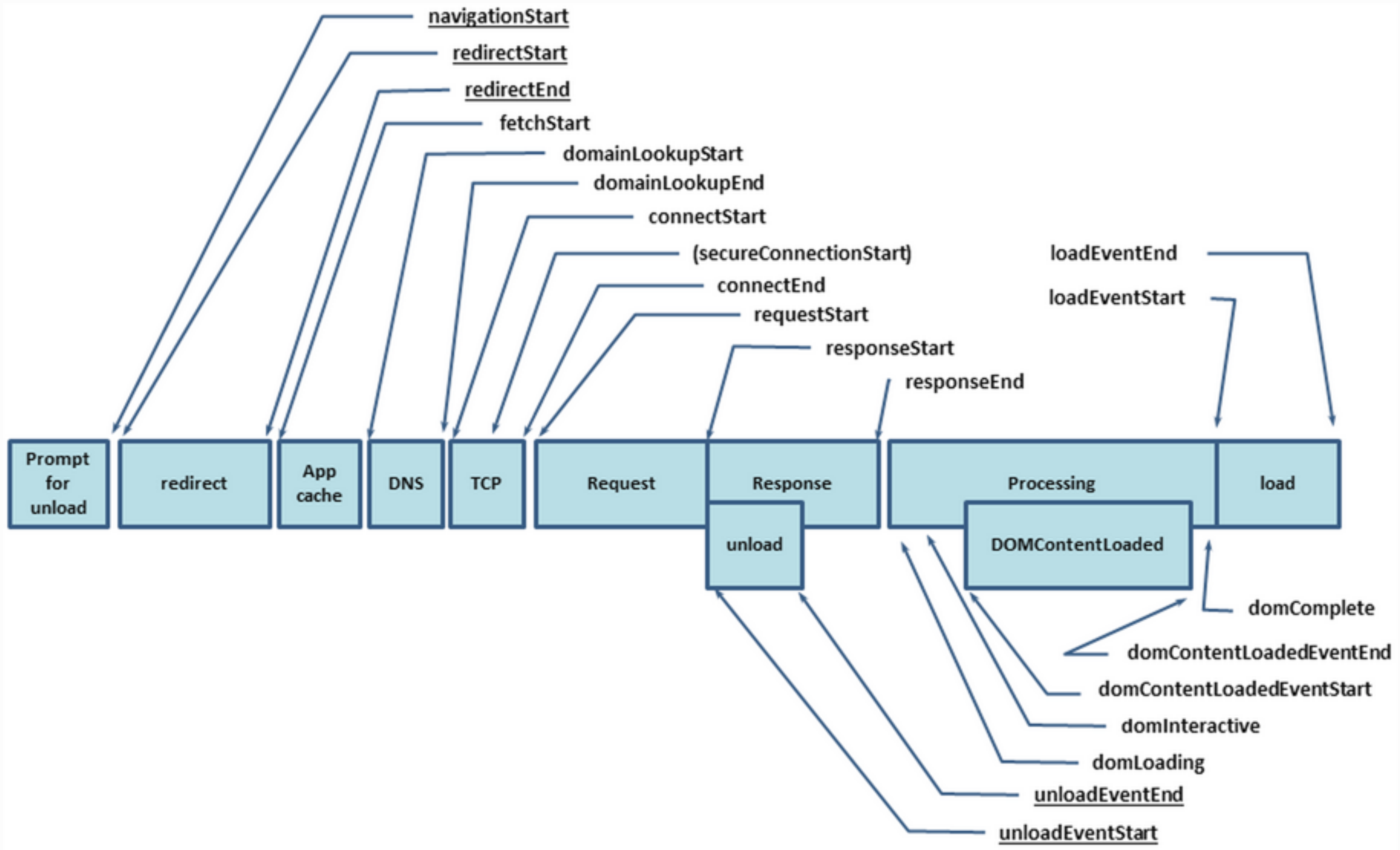
Navigation Timing is a JavaScript API for accurately measuring performance on the web.

HRT

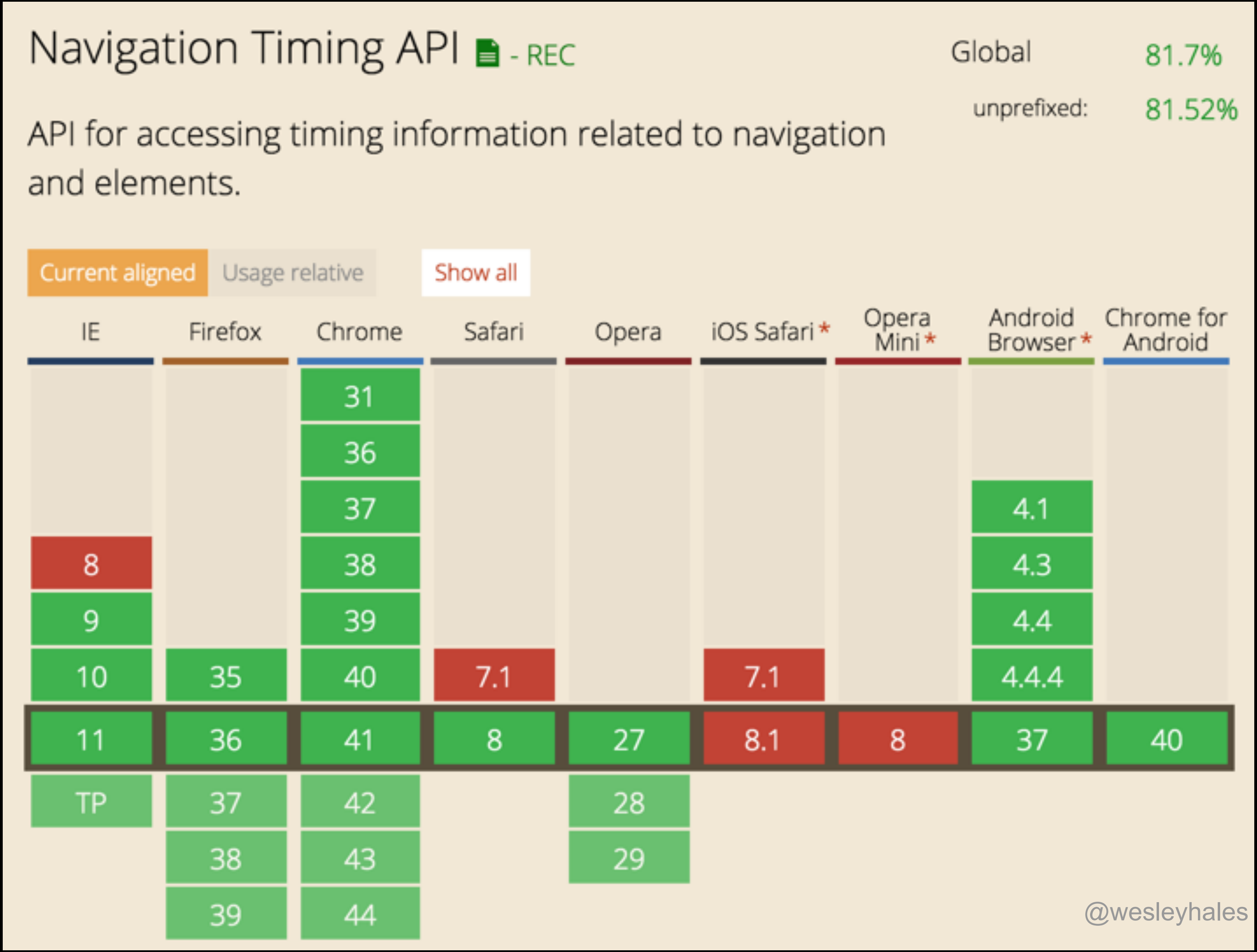


```
performance.now();
```

- The timestamps returned by Performance. now() are up to microsecond precision.
- Example: perf.now.html



@wesleyhales



Navigation Timing API - REC

Global 81.7%

unprefixed: 81.52%

API for accessing timing information related to navigation and elements.

Current aligned | Usage relative | Show all

IE	Firefox	Chrome	Safari	Opera	iOS Safari *	Opera Mini *	Android Browser *	Chrome for Android
		🟩 31						
		🟩 36						
		🟩 37					🟩 4.1	
🟥 8		🟩 38					🟩 4.3	
🟩 9		🟩 39					🟩 4.4	
🟩 10	🟩 35	🟩 40	🟥 7.1		🟥 7.1		🟩 4.4.4	
🟩 11	🟩 36	🟩 41	🟩 8	🟩 27	🟥 8.1	🟥 8	🟩 37	🟩 40
🟩 TP	🟩 37	🟩 42		🟩 28				
	🟩 38	🟩 43		🟩 29				
	🟩 39	🟩 44						

@wesleyhales

Demo

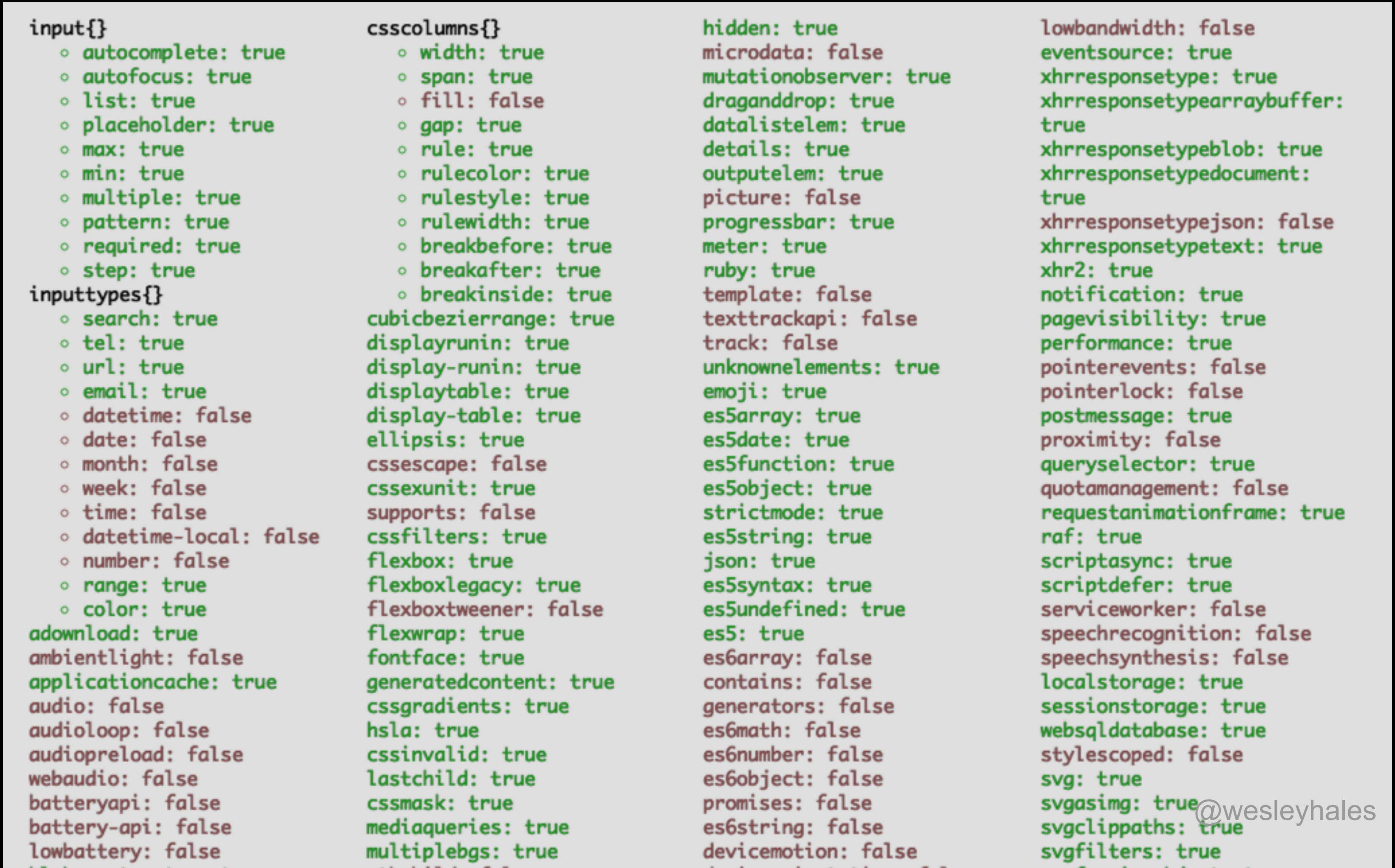
window.performance
simple-new.html

PhantomJS 2

- Released January 2015
- Headless Web browser
- Based on QTWebkit

@wesleyhales

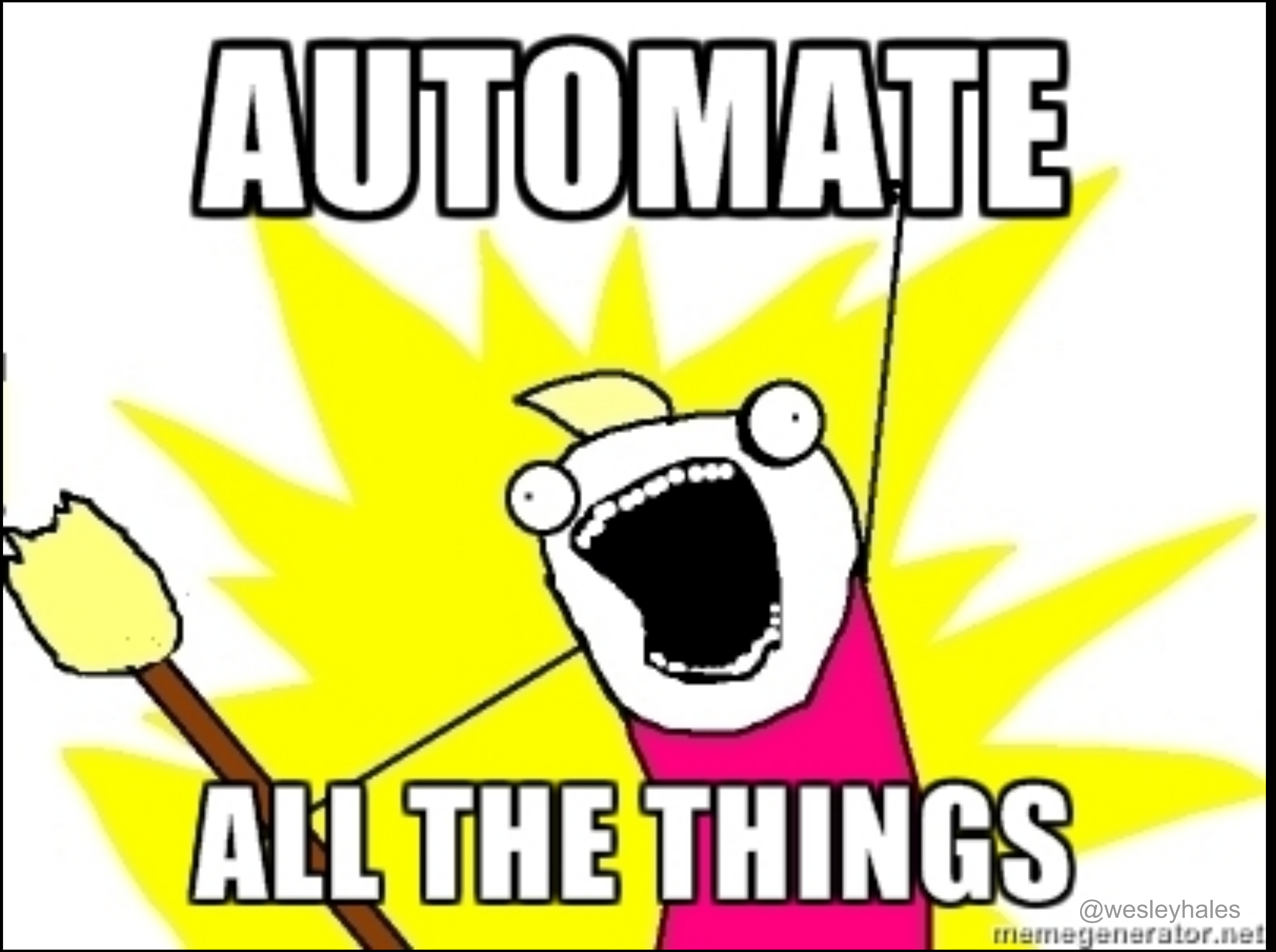
PhantomJS 2 Feature Detect



```
input{}
  ◦ autocomplete: true
  ◦ autofocus: true
  ◦ list: true
  ◦ placeholder: true
  ◦ max: true
  ◦ min: true
  ◦ multiple: true
  ◦ pattern: true
  ◦ required: true
  ◦ step: true
inputtypes{}
  ◦ search: true
  ◦ tel: true
  ◦ url: true
  ◦ email: true
  ◦ datetime: false
  ◦ date: false
  ◦ month: false
  ◦ week: false
  ◦ time: false
  ◦ datetime-local: false
  ◦ number: false
  ◦ range: true
  ◦ color: true
adownload: true
ambientlight: false
applicationcache: true
audio: false
audioloop: false
audiopreload: false
webaudio: false
batteryapi: false
battery-api: false
lowbattery: false
csscolumns{}
  ◦ width: true
  ◦ span: true
  ◦ fill: false
  ◦ gap: true
  ◦ rule: true
  ◦ rulecolor: true
  ◦ rulestyle: true
  ◦ rulewidth: true
  ◦ breakbefore: true
  ◦ breakafter: true
  ◦ breakinside: true
cubicbezierrange: true
displayrunin: true
display-runin: true
displaytable: true
display-table: true
ellipsis: true
cssescape: false
cssexunit: true
supports: false
cssfilters: true
flexbox: true
flexboxlegacy: true
flexboxtweener: false
flexwrap: true
fontface: true
generatedcontent: true
cssgradients: true
hsla: true
cssinvalid: true
lastchild: true
cssmask: true
mediaqueries: true
multiplebgs: true
hidden: true
microdata: false
mutationobserver: true
draganddrop: true
datalistelem: true
details: true
outputelem: true
picture: false
progressbar: true
meter: true
ruby: true
template: false
texttrackapi: false
track: false
unknownelements: true
emoji: true
es5array: true
es5date: true
es5function: true
es5object: true
strictmode: true
es5string: true
json: true
es5syntax: true
es5undefined: true
es5: true
es6array: false
contains: false
generators: false
es6math: false
es6number: false
es6object: false
promises: false
es6string: false
devicemotion: false
lowbandwidth: false
eventsource: true
xhrresponsetype: true
xhrresponsetypearraybuffer:
true
xhrresponsetypeblob: true
xhrresponsetypedocument:
true
xhrresponsetypejson: false
xhrresponsetypetext: true
xhr2: true
notification: true
pagevisibility: true
performance: true
pointerevents: false
pointerlock: false
postmessage: true
proximity: false
queryselector: true
quotamanagement: false
requestanimationframe: true
raf: true
scriptasync: true
scriptdefer: true
serviceworker: false
speechrecognition: false
speechsynthesis: false
localstorage: true
sessionstorage: true
websqldatabase: true
stylescoped: false
svg: true
svgasimg: true
svgclippaths: true
svgfilters: true
```

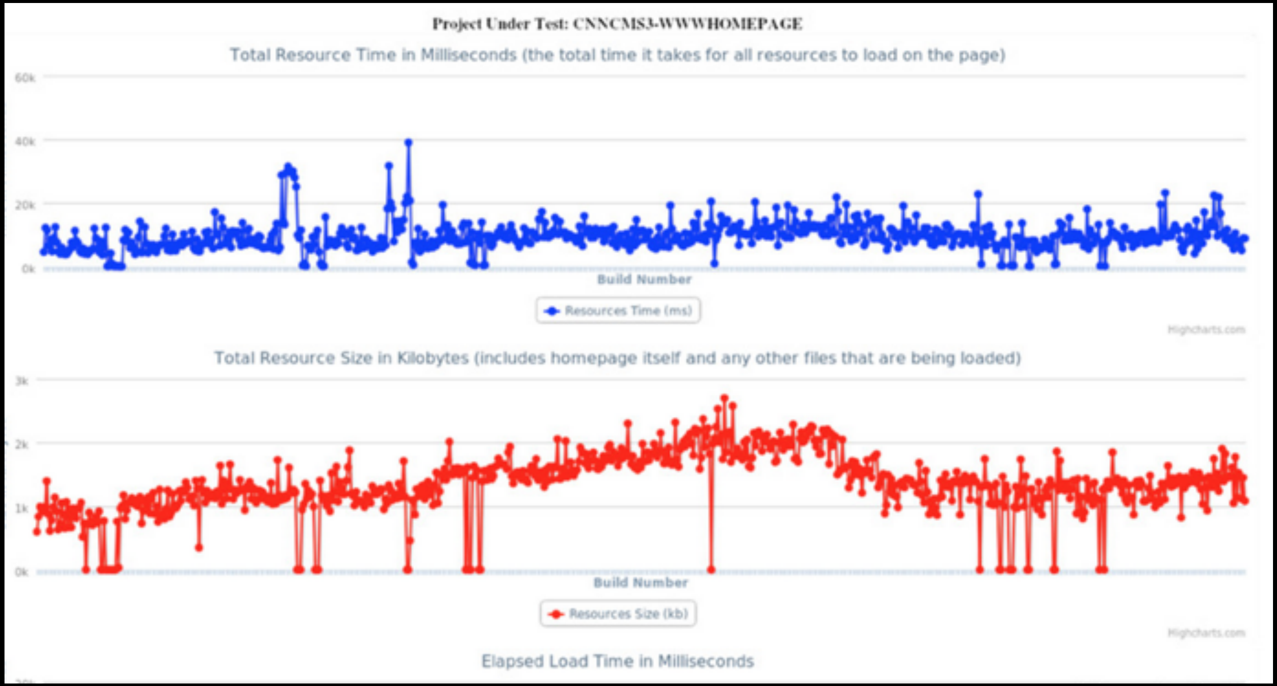
Demo

Basic PhantomJS && http://phantomjs.org/examples/



@wesleyhales

Loadreport.js (2012-2015)

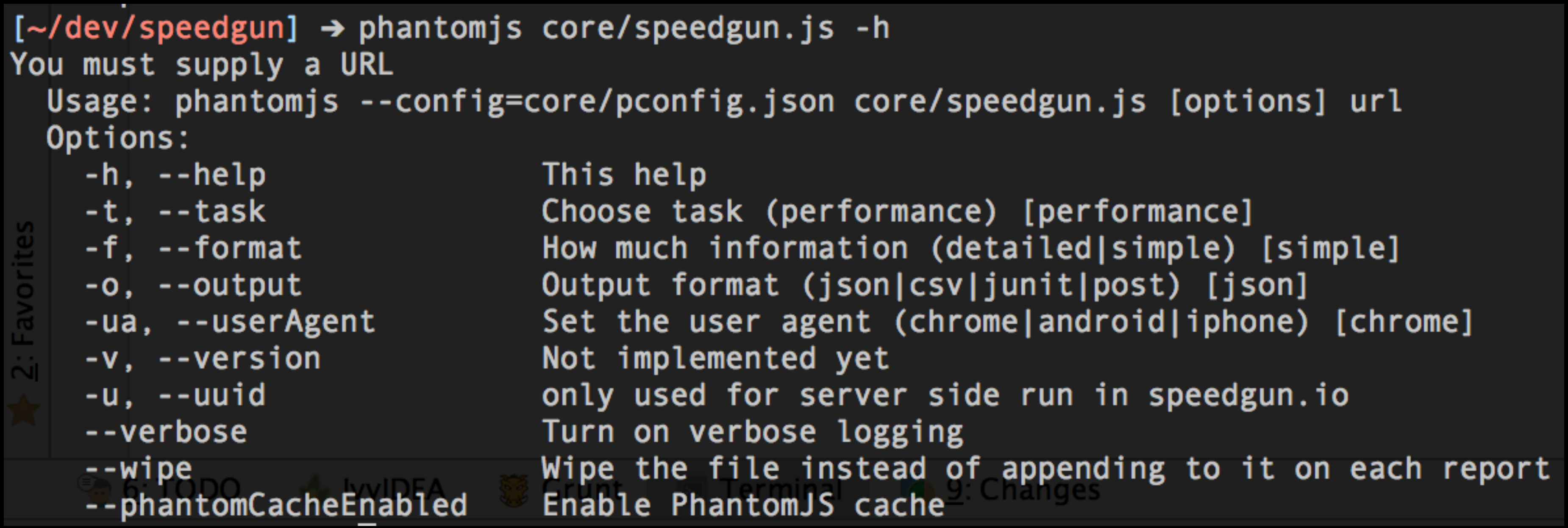


@wesleyhales

Speedgun.js

- Rewrite of loadreport.js
- Leverages all implemented PhantomJS 2 Navigation Timing APIs
- (shims resource timing)

Speedgun.js



```
[~/dev/speedgun] ➔ phantomjs core/speedgun.js -h
You must supply a URL
  Usage: phantomjs --config=core/pconfig.json core/speedgun.js [options] url
  Options:
    -h, --help                  This help
    -t, --task                  Choose task (performance) [performance]
    -f, --format                How much information (detailed|simple) [simple]
    -o, --output                Output format (json|csv|junit|post) [json]
    -ua, --userAgent            Set the user agent (chrome|android|iphone) [chrome]
    -v, --version               Not implemented yet
    -u, --uuid                  only used for server side run in speedgun.io
    --verbose                   Turn on verbose logging
    --wipe                      Wipe the file instead of appending to it on each report
    --phantomCacheEnabled       Enable PhantomJS cache
```

@wesleyhales

Demo

Speedgun.js

@wesleyhales

Speedgun.io

- Allows Speedgun.js to run as a service
- Dockerized
- New term… Synthetic RUM

Demo

Speedgun.io

RUM

Real user monitoring (RUM) is a passive monitoring technology that records all user interaction with a website or client interacting with a server or cloud-based application.

Synthetic RUM?

- Use Speedgun.io as centralized server
- All docker nodes send beacon with:
  - Current container CPU and memory usage
- Yes, another demo...

Thanks!!

- speedgun.io (github)
- Navigation Timing API
- Navigation Timing 2
- Resource Timing API