

A Baseline for Web Performance with PhantomJS

@wesleyhales

@ryanbridges



Wesley Hales

wesleyhales



ryan bridges

r3b



loadreport

A performance focused fork of confess.js

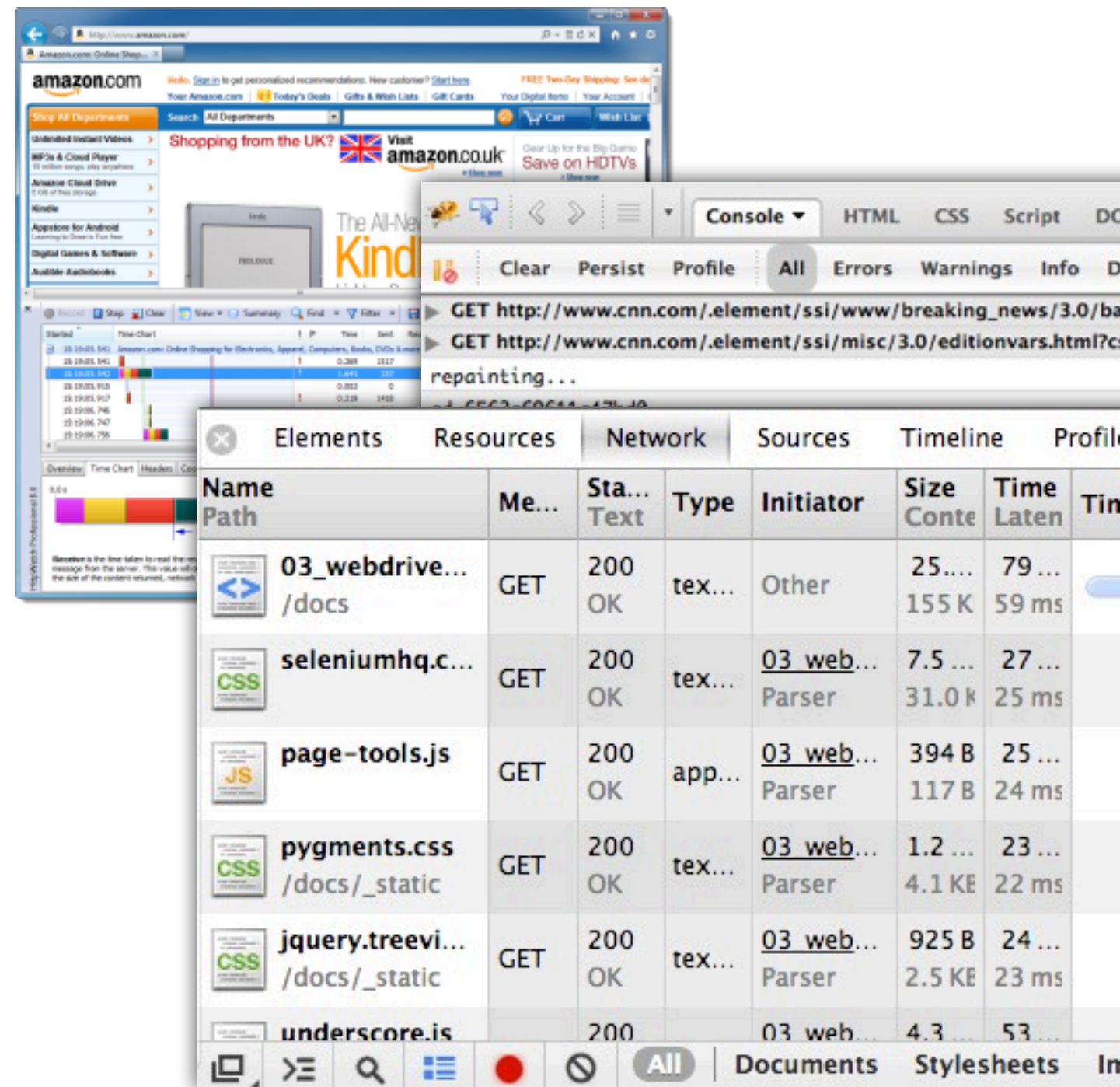
71 ★



more awesome pictures at THEMETAPICTURE.COM

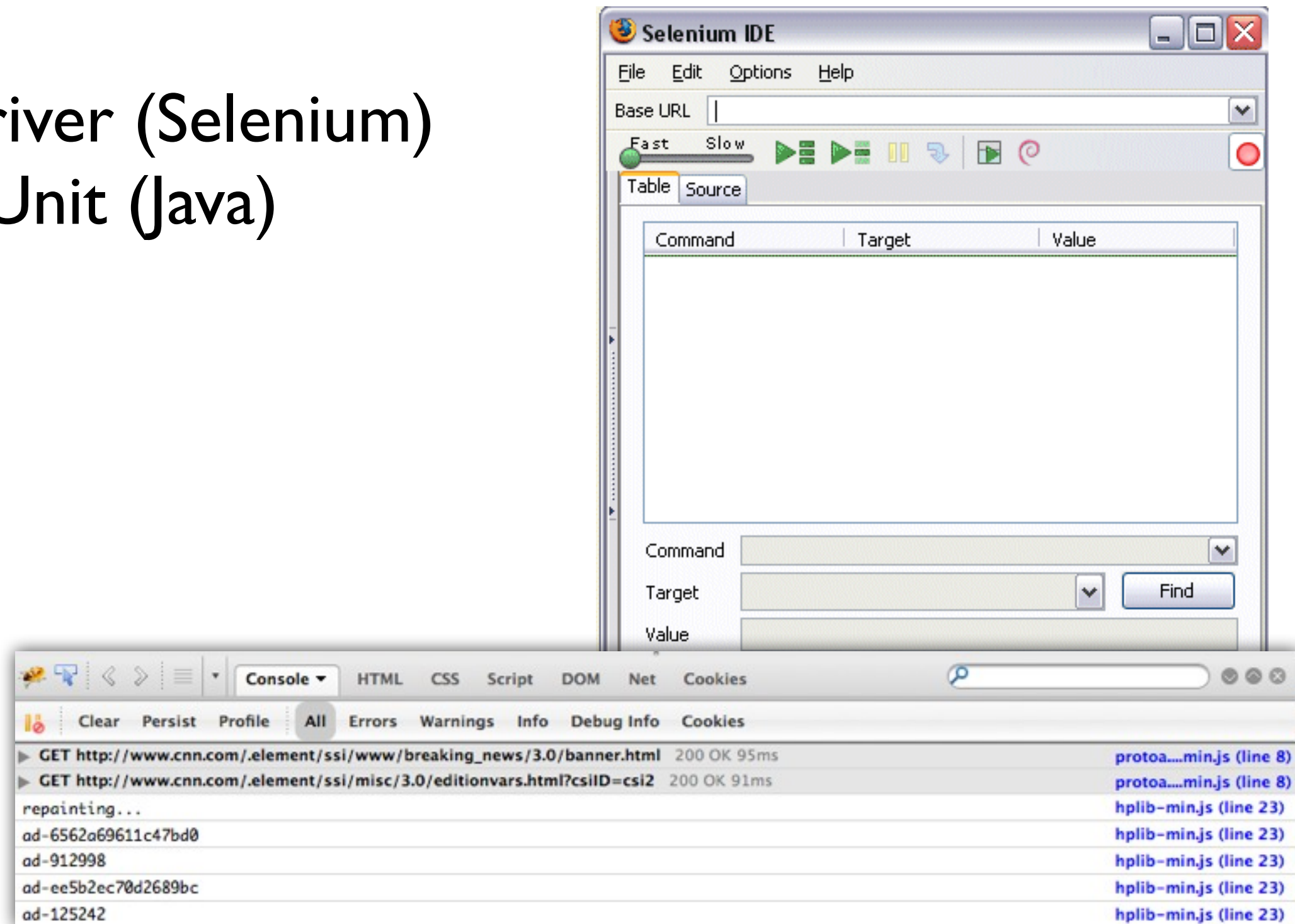
Page Load Testing (Manual)

- Dev Tools
- Firebug (Net panel)
- HTTPWatch
- HAR Viewers
- Charles/Proxies
- Etc...



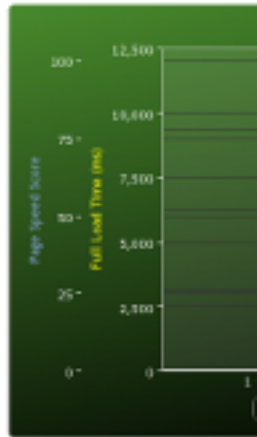
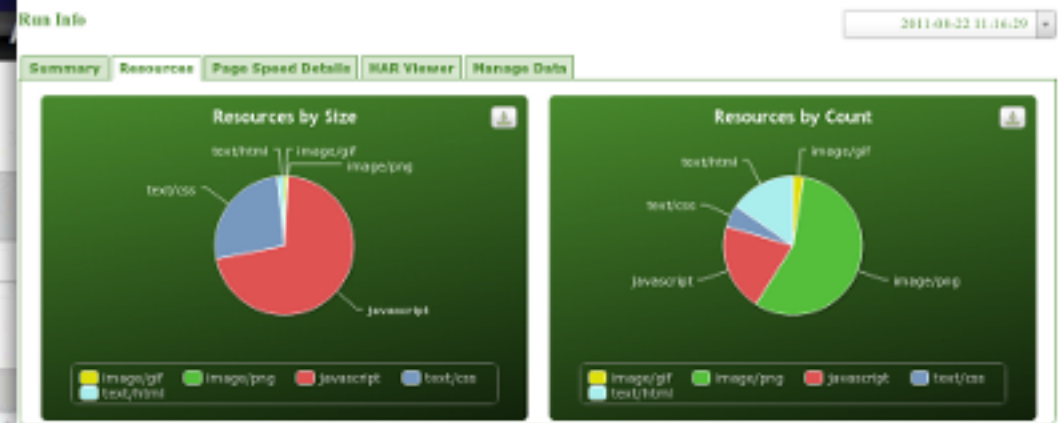
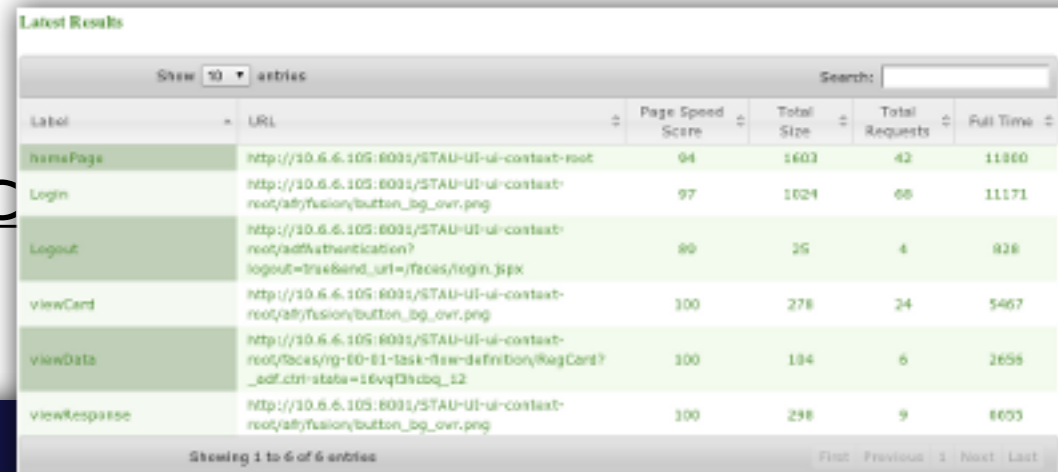
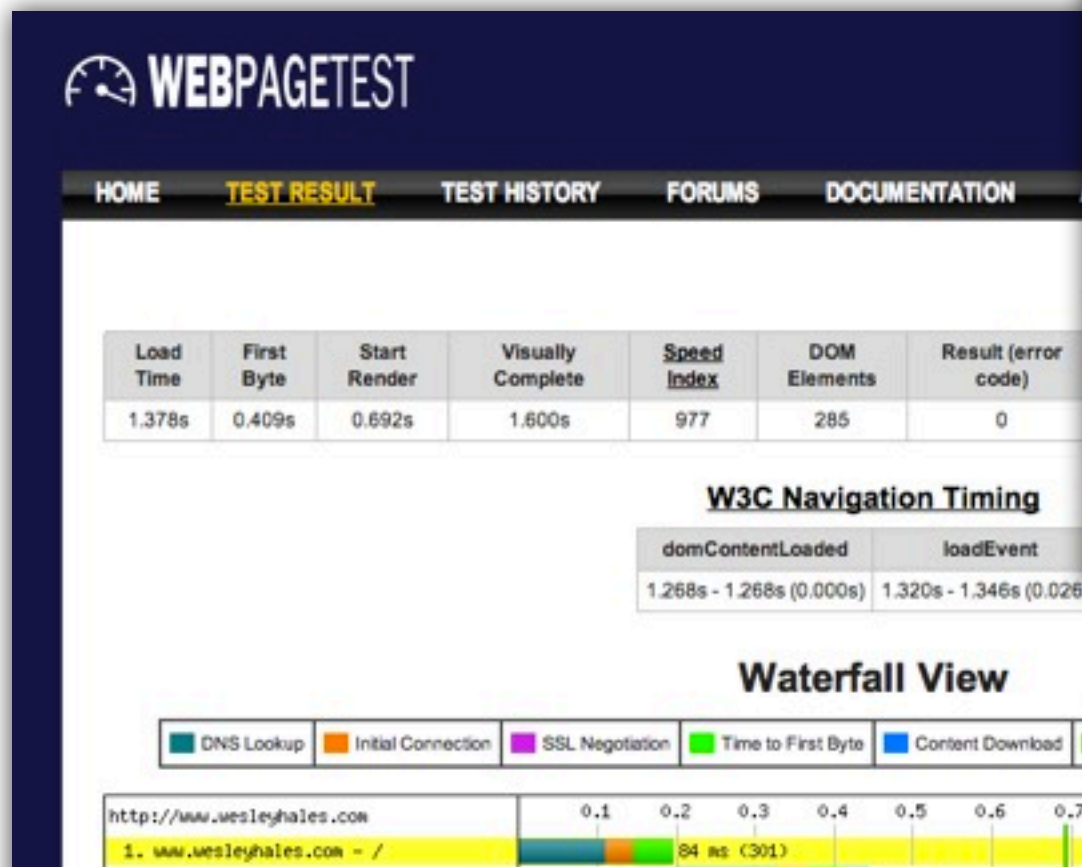
Page Load Testing (Automated)

- WebDriver (Selenium)
- HTMLUnit (Java)



Page Load Testing (Services)

- harstorage.com (MongoDB and Pylons)
- harviewer (PHP + Javascript)
- webpagetest.org (PHP)
- pcapperf.appspot.com/ (C++)



Summary	Resources	Page Size
GET /css/3.jpg	20	
GET /CSS/Canvas_1_12_3.js	20	
GET /CSS/CanvasElements_norl_3_15	20	
GET /css-11.1.1.4.8-0205.js	20	
GET /css/FontFace-loading-ajpe	20	
GET /css-11.1.1.4.8-0205.js	20	
GET /Aad/translations/11.1.1.4.8	20	
GET /data/11.1.1.4.8-0205.js	20	
GET /img/11.1.1.4.8-0205.js	20	
GET /css-11.1.1.4.8-0205.js	20	
GET /page_background_whiles	20	
GET /info.jpg	20	
GET /waring.png	20	
GET /error.png	20	
GET /logoWithOutLine.png	20	
GET /confirmation.png	20	
GET /blank.html	20	
GET /css/1.jpg	20	
GET /css/1.jpg	20	
GET /diag-3-header-middle.png	20	
GET /diag-3-footer-middle.png	20	
GET /css/1.jpg	20	
GET /css/1.jpg	20	

Front-end performance basics

- Caching (headers, localStorage, appcache)
- When to gzip (don't gzip stuff that's natively compressed)
- When to pull from CDN, local, or break apart (single HTTP reqs aren't always a win)

CODING

[CSS](#)[HTML](#)[JavaScript](#)[Techniques](#)

DESIGN

[Web Design](#)[Typography](#)[Inspiration](#)[Business](#)

MOBILE

[Responsive](#)[iPhone & iPad](#)[Android](#)[Design Patterns](#)

GRAPHICS

[Photoshop](#)[Illustrator](#)

Front-End Ops

By [Alex Sexton](#) June 11th, 2013 [Web Design, Workflow](#) [25 Comments](#)

When a team builds a complex application, there is often a common breakdown of roles. Specifically on the back end, there are database engineers, application engineers and operations engineers, or something close to this. In recent years, more and more **application logic is being deferred to the client side**. For some reason, though, operations folks aren't going with it.

I recently wrote an article on "[Deploying JavaScript Applications](#)." It was largely well received, and I was happy with the content, but one negative comment stuck out to me. I probably didn't have the reaction that the commenter was intending, but it pointed out something to me nonetheless.

"With all due respect, may I ask if you actually enjoy your job? I am a dev, and I do enjoy using tech to do stuff to a point. If your role is to squeeze every last second of performance out of your app, then yea, all this stuff must be cool. BUT if you are a coder doing something else and then come back to all of this as well, then wow, I don't know how you haven't gone mad already. I'd be sick to the stomach if I had to do all of this, in addition to my usual work."

“...If your role is to squeeze every last second of performance out of your app, then yea, all this stuff must be cool. BUT if you are a coder doing something else and then come back to all of this as well, then wow, ...I’d be sick to the stomach if I had to do all of this, in addition to my usual work.”

Front-End Ops

- Constant eye on front-end performance
- Manager of builds/dependencies (grunt, bower, etc..)
- Expert on delivering content from server to browser.
- Critical of new HTTP requests, file sizes and load times.

Justifying Front End Perf

- HARs are indispensable and are driven by a specification (currently 1.2), but you need more
- Adding a front-end operations position to the team, or just getting time approved to work on front-end performance requires...

...better analytics

Project Under Test: CNNCMS3-WWWHOMEPAGE

Total Resource Time in Milliseconds (the total time it takes for all resources to load on the page)



Highcharts.com

Total Resource Size in Kilobytes (includes homepage itself and any other files that are being loaded)



Highcharts.com

Elapsed Load Time in Milliseconds



phantom cache enabled	task name	document.readyState=interactive	window.onload	document.re
Performance with default phantom cache enabled				
yes	performance	860	1077	
yes	performance	719	956	
yes	performance	873	1012	
yes	performance	816	1422	
yes	performance	723	1133	
		798.2	1120	
Performance with default phantom cache disabled				
no	performance	1103	1460	
no	performance	1058	1385	
no	performance	1403	2184	
no	performance	1040	1379	
no	performance	1048	1519	
		1130.4	1585.4	
Performance with default phantom cache disabled and performancecache enabled				
no	performancecache	602	1861	
no	performancecache	671	937	
no	performancecache	636	872	
no	performancecache	808	1050	
no	performancecache	659	928	
		675.2	1129.6	
Report from 8/2				
phantom cache enabled	task name	document.readyState=interactive	window.onload	document.re
Performance with default phantom cache enabled				
yes	performance	775	1096	
yes	performance	776	929	
yes	performance	695	932	

Metrics must be
 measured consistently
 before and after each
 build.
 (especially when
 proving that you
 actually did something
 good)

After breaking
jQuery out of the
“I http request”
concatenation and
far-future caching
it, we greatly
improved site
performance



- think about millions of users/day
- jQuery weight around 32 KB (minified and gzipped)
- deployment every 2 weeks with new concatenated JS file for users to download

Holding it together

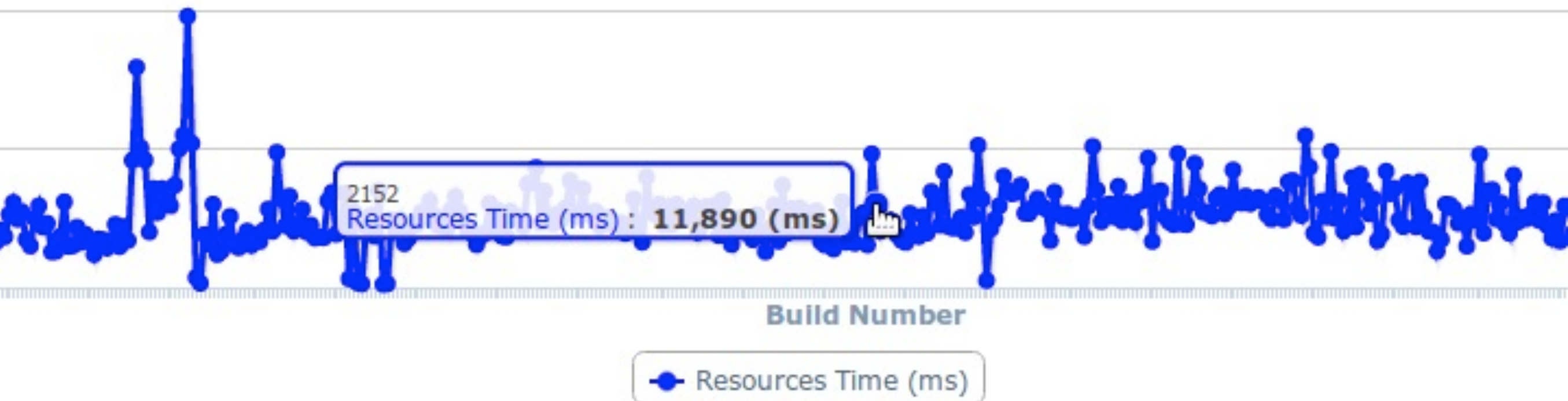


Getting Started

- Build and deployment
- Front-end operations for CNN.com
- Continuous Integration with PhantomJS

CNN.com home page

(per build analysis)



PhantomJS

- Gives us access to
 - Cookie and Local Storage related APIs
 - SSL support
 - DOM readystate
 - Screen capture
 - HAR generation
 - A ton more...

Loadreport.js

- Goals
 - Consistent measurements for baseline
 - Historical tracking
 - Understanding page load time

Loadreport.js

- Demo
 - Basic Report
 - Filmstrip
 - Detailed charts
 - Service
(<http://loadreport.wesleyhailes.com>)

The hard parts

- Performance issues (on the front-end) require more thought
- `advertisers`, `document.write`, too much concatenation

The hard parts

- Server (browser) location
- Will you be running tests on your LAN, Cloud (multiple instances), or elsewhere?

The hard parts

- Browser types
- DIY options are limited

The hard parts

- CPU activity
- Affects all tests. Need as few (or consistent) processes running - that won't cause a spike during test.

CI Setup

Jenkins


Jenkins


yslow.org


#63


Test Results


ENABLE AUTO REFRESH


[Back to Project](#)


[Status](#)


[Changes](#)

[Console Output](#)

[Edit Build Information](#)


[History](#)

[Test Result](#)

[Previous Build](#)

Test Result

4 failures (±0) , 1 skipped (±0)

24 tests (±0)
Took 0 ms.
 [add description](#)

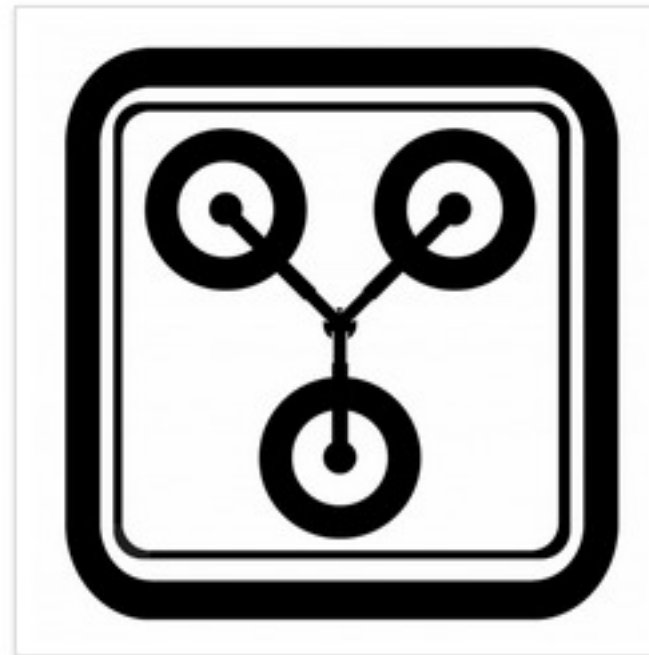
All Failed Tests

Test Name	Duration	Age
<<< YSlow.ynumreq: Make fewer HTTP requests Error Details This page has 7 external Javascript scripts. Try combining them into one. This page has 5 external stylesheets. Try combining them into one.	0 ms	3
<<< YSlow.ycdn: Use a Content Delivery Network (CDN) Error Details There are 3 static components that are not on CDN. <p>You can specify CDN hostnames in your preferences. See YSlow FAQ for details.</p>	0 ms	3

CI Setup

From the enterprise side, we can run the scripts internally with no special setup.

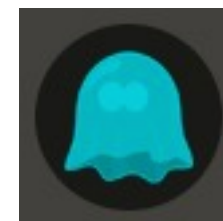
- Bamboo
- Jenkins
- Barebone Linux system without X11 is not a problem for PhantomJS.
- Use Vagrant to create build and test servers on demand, with predictable server- and client-side resources



**UI Performance
Monitoring**
fluxui



[@fluxui](https://github.com/fluxui)





CI

Continuous Integration Server scripts

Last updated 12 hours ago

Shell

★ 0

🔗 0



FluxScripts

Scripts required to run reports

Last updated 13 hours ago

JavaScript

★ 0

🔗 1



FluxServer-node

Last updated 19 hours ago

JavaScript

★ 0

🔗 0



FluxServer-vagrant-node

Last updated 19 hours ago

Ruby

★ 0

🔗 0



cookbooks

Last updated 19 hours ago

Ruby

★ 0

🔗 0



FluxServer-Java

Last updated 3 days ago

JavaScript

★ 0

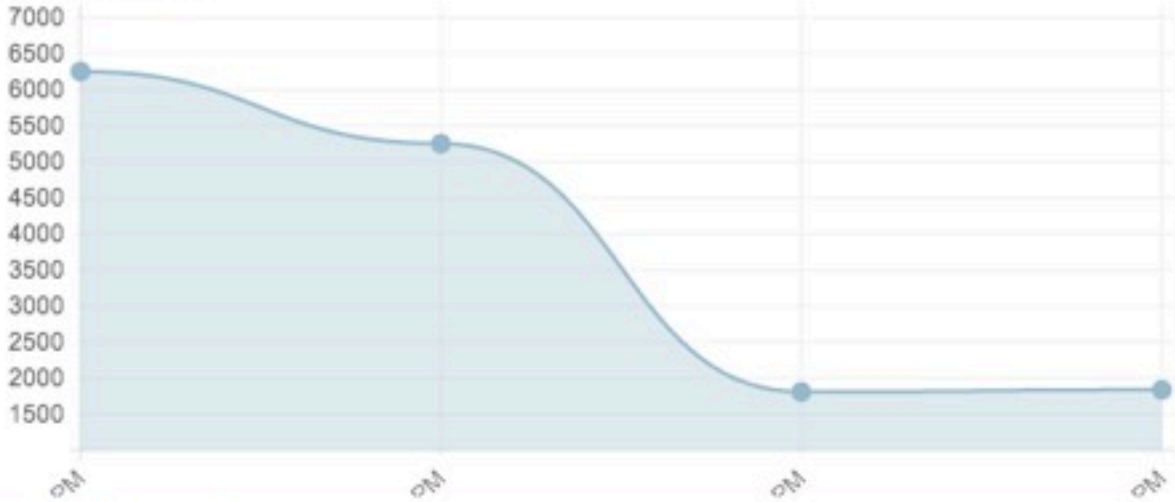
🔗 0

TravisCI Demo

From an opensource perspective, we can use Travis CI

- github OAuth token
- encryption with travis
- pushing your site with an automated travisci build
- <http://bit.ly/14xDZMO>

Elapsed Load Time



Number of Resources

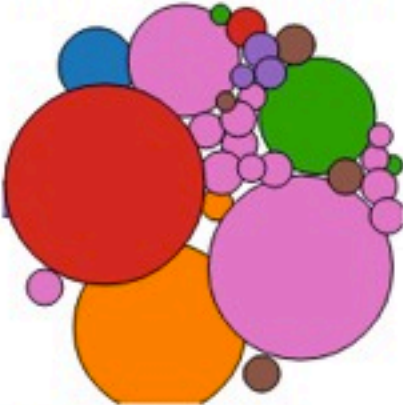


Total Resource Time

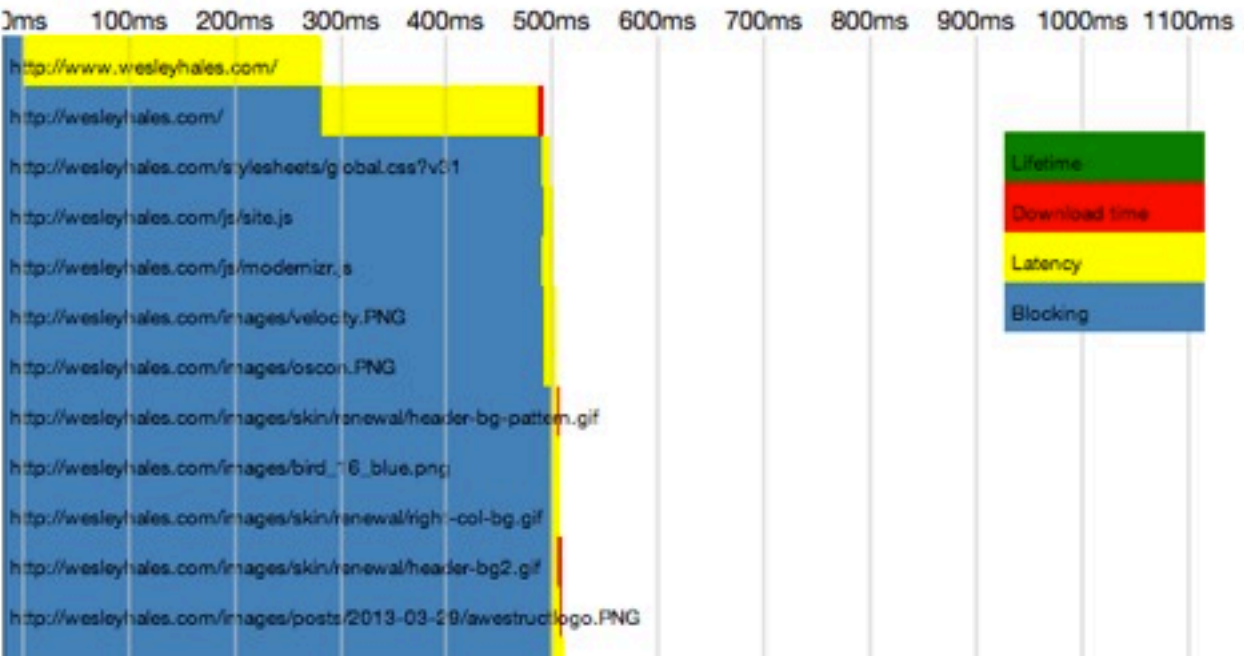


Author: wesleyhales
Commit message: update build scripts
Github link:
<https://github.com/wesleyhales/wesleyhales.com/commit/908e1959fe7f0a35cb9f3ac861a9200489712b80>

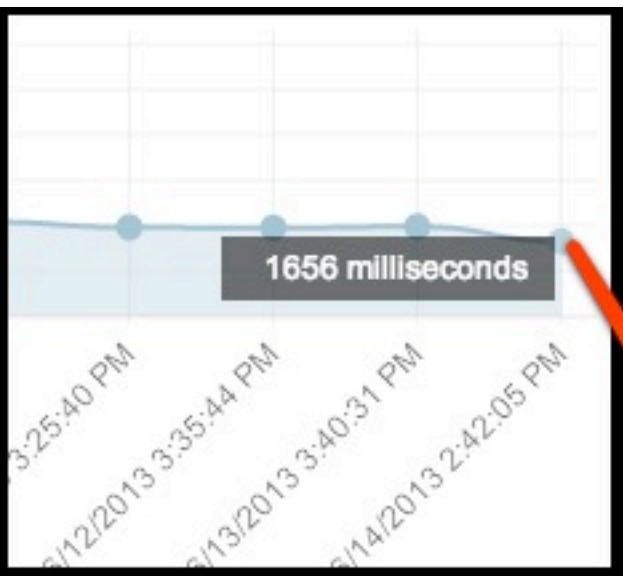
Size and content type



Request lifecycle



<http://bit.ly/Xp9G87>



Search or type a command ?

Explore Gist Blog Help

wesleyhailes

Pull Request Unwatch Unstar 1 Fork 0

Work Pull Requests 0 Issues 0 Wiki Graphs Settings

Commits Branches 2 Tags

add defer back to scripts

master

wesleyhailes authored 3 days ago

1 parent 4edd215 commit e749d14bd5e789b0f162e4650a46163a796979c8

Showing 1 changed file with 2 additions and 2 deletions.

Show Diff Stats

4 _layouts/base.html.haml View file @ e749d14

@@ -65,6	+65,6	_layouts/base.html.haml
65	65	~ content
66	66	%br{:class=>'clear'}
67	67	
68	-	%script{:src => "http://twitter.com/javascripts/blogger.js", :type => "text/javascript"}
69	-	%script{:src => "/js/site.js", :type => "text/javascript"}
	68	+ %script{:src => "http://twitter.com/javascripts/blogger.js", :type => "text/javascript", :defer => "defer"}
	69	+ %script{:src => "/js/site.js", :type => "text/javascript", :defer => "defer"}
70	70	= google_analytics

<http://bit.ly/Xp9G87>

What about Flash?

Summary

- For simplicity, use phantomJS and build a script that works for you.
- For more advanced, host your own instance of FluxUI or webpagetest.org

FluxUI Roadmap

- Better UI and analytics over time for page speed and resource consumption.
- Better integration with CI systems.
- SlimerJS integration (Gecko)
- Better storage of results.

Thanks!

@wesleyhales

@ryanbridges