

YSlow Crash Course

(or, I stole these rules from the Yahoo Exceptional Performance team)

Mark Stetzer
@stetzer

Who is this guy?

- Director of Engineering @ ChaCha
- History of developing web apps
- Try to spread the word of WPO
- Personal goal to make sure every app I work on has a YSlow score of over 90

Obligatory Quote

First and foremost, we believe that speed is more than a feature. Speed is the most important feature. If your application is slow, people won't use it.

-Fred Wilson

Introduction

- Yahoo identified a need to optimize the front end of websites in addition to the backend (can only have so many servers)
- Discovered 80% of the end-user response time is on the front-end
- Investigated how multiple browsers render pages, etc. and came up with a list of best practices to follow

YSlow Rules

- Yahoo's best practices list turned into a Firefox/Firebug plugin called YSlow (also available in Chrome, Opera, and mobile)
- Total list is currently 34 items in 7 categories
- YSlow plugin gives a numerical score and letter (A,B,C,D,E, etc.) and suggestions on how to improve

YSlow In Action

Grade **D** Overall performance score 65 Ruleset applied: YSlow(V2) URL: <http://www.cnn.com/>

ALL (23) FILTER BY: [CONTENT \(6\)](#) | [COOKIE \(2\)](#) | [CSS \(6\)](#) | [IMAGES \(2\)](#) | [JAVASCRIPT \(4\)](#) | [SERVER \(6\)](#)

F Make fewer HTTP requests

Grade F on Make fewer HTTP requests

This page has 27 external Javascript scripts. Try combining them into one.
This page has 7 external stylesheets. Try combining them into one.
This page has 53 external background images. Try combining them with CSS sprites.

Decreasing the number of components on a page reduces the number of HTTP requests required to render the page, resulting in faster page loads. Some ways to

F Use a Content Delivery Network (CDN)

A Avoid empty src or href

F Add Expires headers

F Compress components with gzip

Grade D 958.8K

Grade D 828.8K

Get to the Rules!



© Flickr user Honou

Minimize HTTP Requests

- Most end-user response time is spent downloading components in the page
- Reducing number of components reduces wait times
- HTTP 1.1 allows for two simultaneous downloads per host (modern browsers allow even more)
- Reduce # of components w/ combined files



© Flickr user Matthew Hine

Combined Files

- Combine all Javascript/CSS files into a single script/stylesheet - make this part of your release process
- YUI Compressor is one tool that can be used to accomplish this, and even automate
- CSS Sprites used to reduce image requests
- Image maps can be used as well, but usually awkward (since 1998 anyway)
- 40%-60% of daily visitors come in w/ empty cache



© Flickr user Sudhamshu Hebbar

Content Delivery Network (CDN)

- Physical proximity to your server plays a part in performance
- 80% of performance is downloading components; only worry about distributing components
- Use a CDN like Cloudfront to move content closer to end-users
- 20%+ response time improvement using CDN



© Flickr user Ethan Prater

Expires and Cache-Control Headers

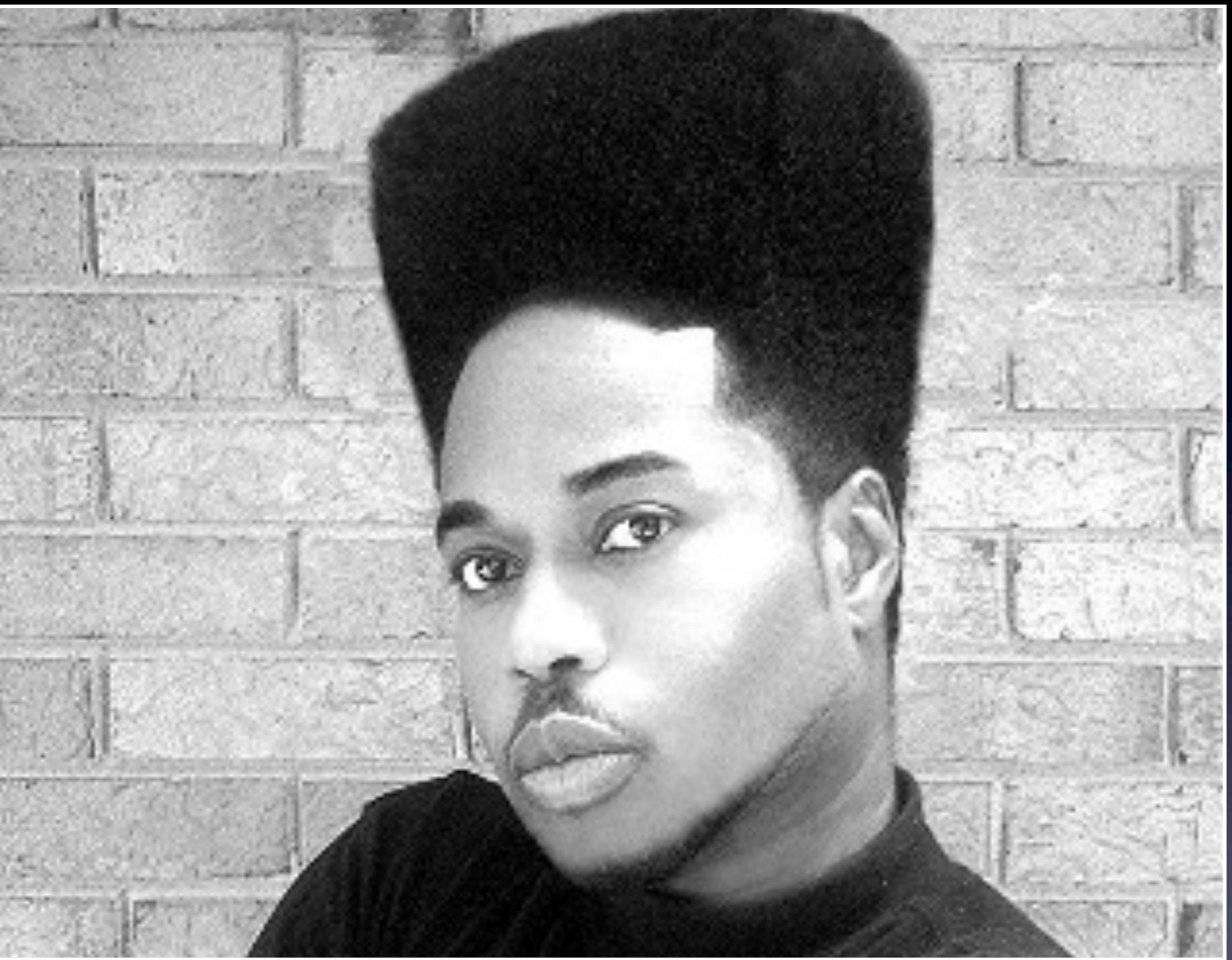
- Static components should get a far future Expires header
- This means if the content changes, the filename will need to change (can be part of release process)
- Dynamic components should have a Cache-Control header
- Number of page views w/ primed cache: 75%-85%



© Flickr user Identity Photogr@phy

Gzip Components

- Gzipping components generally reduces size by 70%
- Approx. 90% of Internet traffic goes through browsers that support gzip
- Gzip HTML, and any other text responses (including JS and CSS)
- Bad idea to gzip images, PDF files



© Flickr user ImagePros

Stylesheets in HEAD

- Putting stylesheets in doc HEAD makes pages appear to load faster (progressive rendering)
- Browsers like IE block rendering if styles are at the bottom (avoid redrawing) - means users stuck watching blank white page
- HTML spec even says to put <link> in HEAD



© Flickr user grongar

Scripts at the Bottom

- Scripts block parallel downloads
- Can serve two parallel files per host, but scripts block ALL other downloads
- Tools like Grails' UI Performance (now Resources) plugin can make placing scripts at bottom of document easy



© Flickr user MinimalistPhotography101.com

Avoid CSS Expressions

- Just don't use 'em
- They get evaluated A LOT (think thousands of times)
- Really, don't use them
- Are you still reading this slide?



© Flickr user Jiri Brozovsky

External Javascript and CSS

- External content can be separately cached by the browser; inline content cannot
- External files mean more HTTP requests
- Common content between pages generally make external; content on common user homepages (~one pageview per session) could go either way
- Could inline on homepage, then download external after page is loaded



© Flickr user bookfinch

Reduce DNS Lookups

- Average DNS lookup can take 20-120 ms
- Less unique hostnames means less parallel downloads though
- Yahoo rule of thumb is 2-4 hosts



© Flickr user Chris Williamson

Minify JS and CSS

- Removes unnecessary code/formatting
- Tools like YUI Compressor can do this
- Obfuscation is an option (for JS), but is more error-prone, plus typically yields ~ same size savings



© Flickr user Myki Roventine

Avoid Redirects

- Redirect via HTTP 301/302 obviously results in another page load
- Browsers will not cache permanent redirects (not without Expires or Cache-Control anyway)
- URL w/o trailing slash results in redirect to trailing slash (e.g. http://www.a.com/b vs http://www.a.com/b/)



© Flickr user James Brooks

Configure ETags

- ETags are unique identifiers (strings) for components
- If browser's ETag matches server's, no need to download
- Typically constructed using server-specific attributes - this is bad in most cases
- If you don't hand-roll ETags, better to just remove them & use something like Last-Modified



teh watr. it no has a flavr.

n00b human finely
lernd 2 flush.

© Flickr user anomalous4

Flush Buffer Early

- If you have access to the buffer, and you are able to, flushing early can help with fetching components
- A good place to flush might be at the end of the <head> tag; usually pretty minimal risk



© Flickr user Les Chatfield

Split Components Across Domains

- Each domain allows 2 parallel downloads
- Remember DNS rule of thumb: 2-4 hosts
- Good fit might be HTML/dynamic content on www.chacha.com; static JS, CSS, etc. on static.chacha.com



© Flickr user Brian Richardson

Cookie-Free Component Domains

- Requests for static content still have to send cookies; waste of bandwidth
- Keeping with previous suggestions, static.chacha.com for static content (no cookies), www.chacha.com for dynamic content (use cookies)

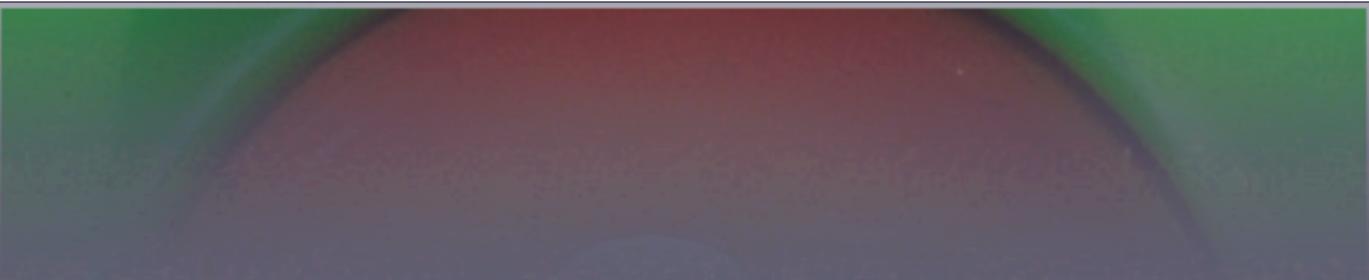
Crazy iPhone Rule

- iPhone won't cache anything over 25k
- The 25k limit applies to the uncompressed size - minification will be a big deal here

...and many more!

- Too many rules to mention here; tried to pick the biggest offenders and most interesting
- Check out <http://developer.yahoo.com/performance/rules.html> for a full list
- Download YSlow at <http://developer.yahoo.com/yslow/>
- Steve Souders' books are [High Performance Web Sites](#) and [Even Faster Web Sites](#)

Questions?



Demo Time