

High Performance Web Sites

Tips for faster pages



Nate Koechley
natek@yahoo-inc.com
nate.koechley.com/blog
developer.yahoo.com/yui



Y! Performance Research Engineering Team

- Tenni Theurer, Nicole Sullivan,
Philip Tellis, Stoyan Stefanov,
Swapni Shinde, Wayne Shea,
Anand Kishore (Intern)

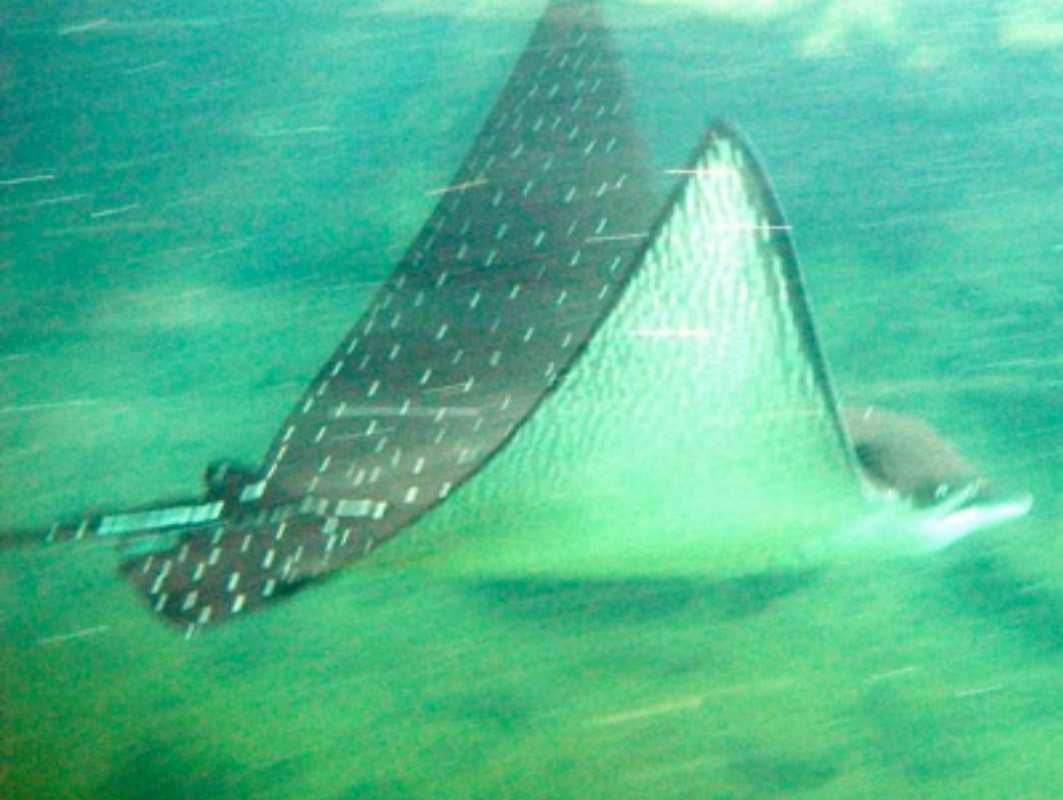
Why talk about
performance?

I: Because fast is better













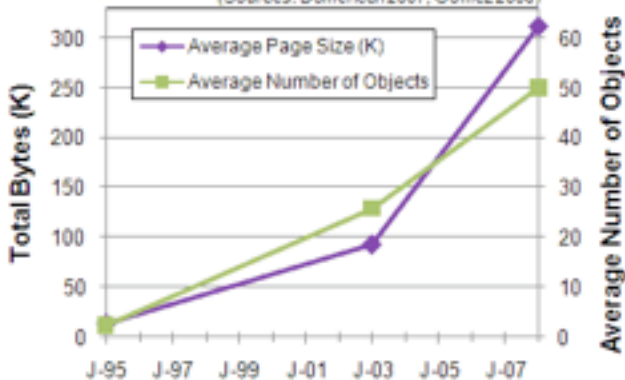


2: Because sites are bigger

Modern web sites &
applications have changed
architecturally.

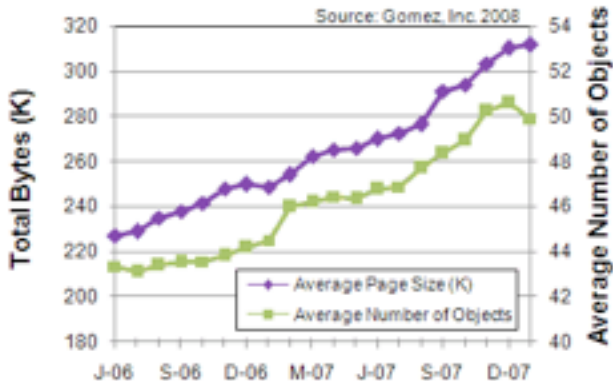
Growth of Average Web Page Size and Number of Objects

(Sources: Domenech 2007, Gomez 2008)



From 2003 to 2008:
97K to 312K.
25.7 to 49.9 objects.

Top 1000 Home Page Growth - Average Size and Number of Objects from June 2006 to January 2008



In past 12 months, Top 1000 sites:
From 250K to 310.4K.

WebSiteOptimization.com/speed/tweak/average-web-page

What *is* performance?

Two Flavors:

Response Time
&
System Efficiency



**Our focus is
the response
time of web
products**

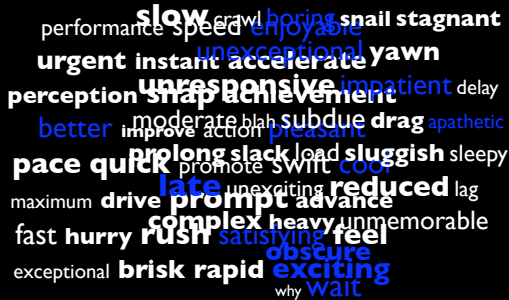


Foundational Research:

#1

Perception

perceived response time



what is the end user's experience?

It's in the eye of the beholder

- Perception and usability are important performance metrics.
- More relevant than actual unload-to-onload time.
- Definition of "user onload" is undefined or varies from one web page to the next.

#2

“80% of consequences come from
20% of causes”

—Vilfredo Pareto

The Importance of Front-End Performance

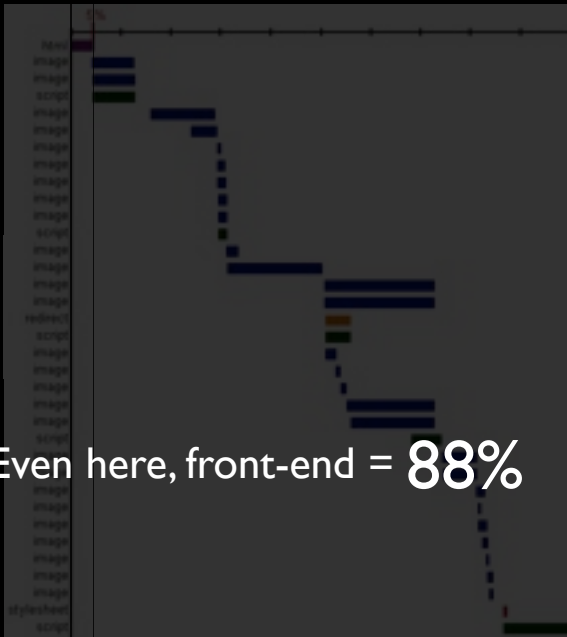
Back-end

= 5%

Front-end

= 95%

Even here, front-end = **88%**





Back-end vs. Front-end

	Empty Cache	Full Cache
amazon.com	82%	86%
aol.com	94%	86%
cnn.com	81%	92%
ebay.com	98%	92%
google.com	86%	64%
msn.com	97%	95%
myspace.com	96%	86%
wikipedia.org	80%	88%
yahoo.com	95%	88%
youtube.com	97%	95%

Until now our
optimization efforts
have targeted the tip
of the iceberg.



#3

Cache

Empty vs. Full Cache



Empty vs. Full Cache

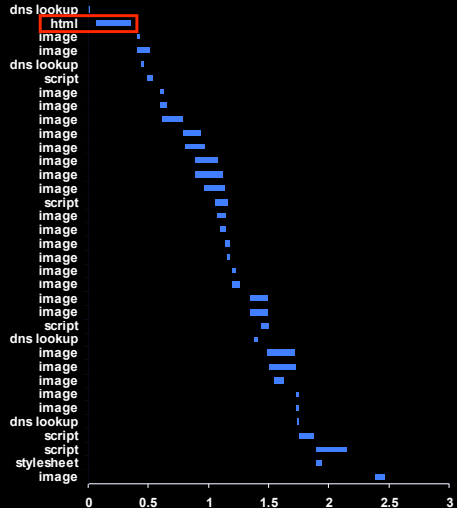
1



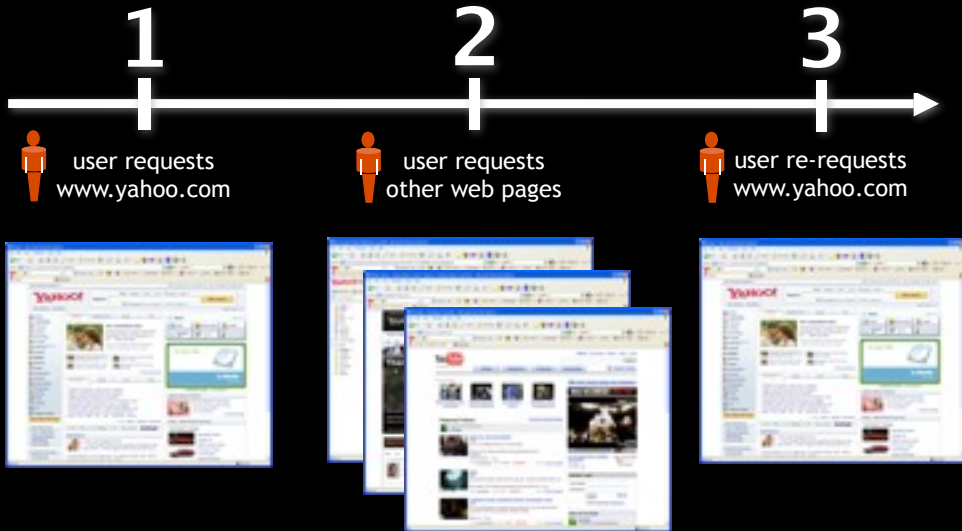
user requests
www.yahoo.com



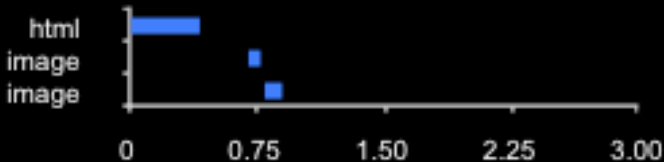
with an empty cache



Empty vs. Full Cache



Empty vs. Full Cache



Empty vs. Full Cache

Empty Cache	Full Cache
28.0K 1 HTML document	
1.9K 1 Style Sheet File	
59.5K 4 JavaScript Files	28.0K 1 HTML document
78.7K 24 Images	0.1K 2 Images
168.1K Total size	28.1K Total size
30 HTTP requests	3 HTTP requests
2.4s Response time*	0.9s Response time*

empty = 2.4 seconds

full = 0.9 second

full = 83% fewer bytes.

full = 90% fewer HTTP requests.

#3b

(sadly, the cache
doesn't work as well
as we wish it did.)

How much does caching benefit our users?

Q1: What % of **unique users** view a page with an empty cache?

Q2: What % of **page views** are with an empty cache?

Browser Cache Experiment

Add a new image to your page

```

```



with the following response headers:

Expires: Thu, 15 Apr 2004 20:00:00 GMT

Last-Modified: Wed, 28 Sep 2006 23:49:57 GMT

Browser Cache Experiment

Two possible response codes:

200 – The browser does not have the image in its cache.

304 – The browser has the image in its cache, but needs to verify the last modified date.

Browser Cache Experiment

Q1: What % of **unique users** view with an empty cache?



$$\frac{\text{\# unique users with at least one 200 response}}{\text{total \# unique users}}$$

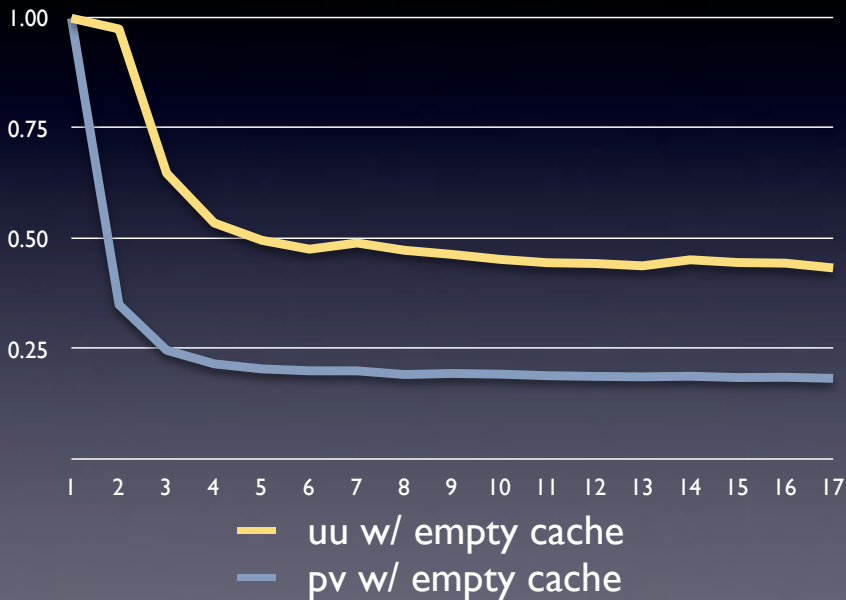
Q2: What % of **page views** are with an empty cache?



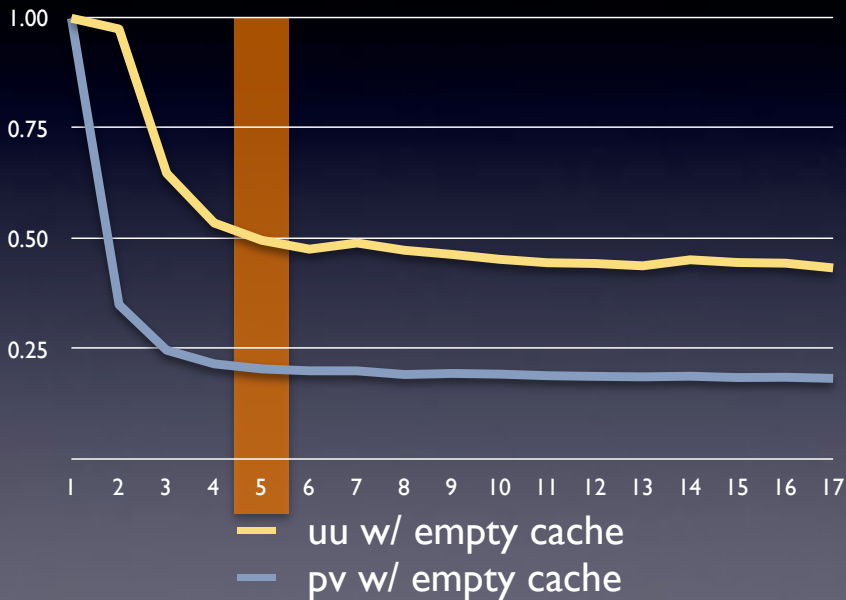
$$\frac{\text{total \# of 200 responses}}{\text{\# of 200 + \# of 304 responses}}$$



Surprising Results



Surprising Results



40-60% UUs w/ empty cache

20% PVs w/ empty cache

Experiment Takeaways

1. The empty cache user experience is more prevalent than you think!
2. Therefore, optimize for both full cache and empty cache experience.

© 2007 Yahoo! Inc. All rights reserved. | [Privacy](#) | [Terms of Use](#) | [Feedback](#) | [Help](#) | [Site Map](#) | [About Us](#) | [Contact Us](#) | [Careers](#) | [Partners](#) | [Press](#) | [Legal](#) | [Site Feedback](#) | [Accessibility](#) | [Yahoo! Japan](#) | [Yahoo! Korea](#) | [Yahoo! Taiwan](#) | [Yahoo! India](#) | [Yahoo! Singapore](#) | [Yahoo! Hong Kong](#) | [Yahoo! Philippines](#) | [Yahoo! Indonesia](#) | [Yahoo! Thailand](#) | [Yahoo! Vietnam](#) | [Yahoo! Malaysia](#) | [Yahoo! Australia](#) | [Yahoo! New Zealand](#) | [Yahoo! South Africa](#) | [Yahoo! Brazil](#) | [Yahoo! Mexico](#) | [Yahoo! Chile](#) | [Yahoo! Colombia](#) | [Yahoo! Peru](#) | [Yahoo! Argentina](#) | [Yahoo! Venezuela](#) | [Yahoo! Ecuador](#) | [Yahoo! Costa Rica](#) | [Yahoo! Panama](#) | [Yahoo! Uruguay](#) | [Yahoo! Paraguay](#) | [Yahoo! Bolivia](#) | [Yahoo! Cuba](#) | [Yahoo! Haiti](#) | [Yahoo! Dominican Republic](#) | [Yahoo! Puerto Rico](#) | [Yahoo! Trinidad and Tobago](#) | [Yahoo! Barbados](#) | [Yahoo! Guyana](#) | [Yahoo! Suriname](#) | [Yahoo! Guadeloupe](#) | [Yahoo! Martinique](#) | [Yahoo! French Guiana](#) | [Yahoo! Cayman Islands](#) | [Yahoo! Bermuda](#) | [Yahoo! Montserrat](#) | [Yahoo! Antigua and Barbuda](#) | [Yahoo! Saint Kitts and Nevis](#) | [Yahoo! Saint Lucia](#) | [Yahoo! Saint Vincent and the Grenadines](#) | [Yahoo! Grenada](#) | [Yahoo! Barbados](#) | [Yahoo! Guyana](#) | [Yahoo! Suriname](#) | [Yahoo! Guadeloupe](#) | [Yahoo! Martinique](#) | [Yahoo! French Guiana](#) | [Yahoo! Cayman Islands](#) | [Yahoo! Bermuda](#) | [Yahoo! Montserrat](#) | [Yahoo! Antigua and Barbuda](#) | [Yahoo! Saint Kitts and Nevis](#) | [Yahoo! Saint Lucia](#) | [Yahoo! Saint Vincent and the Grenadines](#) | [Yahoo! Grenada](#)

Yahoo! User Interface Blog - New 2007-01-04 10:00:00

Yahoo! User Interface Blog - New 2007-01-04 10:00:00 | [Privacy](#) | [Terms of Use](#) | [Feedback](#) | [Help](#) | [Site Map](#) | [About Us](#) | [Contact Us](#) | [Careers](#) | [Partners](#) | [Press](#) | [Legal](#) | [Site Feedback](#) | [Accessibility](#) | [Yahoo! Japan](#) | [Yahoo! Korea](#) | [Yahoo! Taiwan](#) | [Yahoo! India](#) | [Yahoo! Singapore](#) | [Yahoo! Hong Kong](#) | [Yahoo! Philippines](#) | [Yahoo! Indonesia](#) | [Yahoo! Thailand](#) | [Yahoo! Vietnam](#) | [Yahoo! Malaysia](#) | [Yahoo! Australia](#) | [Yahoo! New Zealand](#) | [Yahoo! South Africa](#) | [Yahoo! Brazil](#) | [Yahoo! Mexico](#) | [Yahoo! Chile](#) | [Yahoo! Colombia](#) | [Yahoo! Peru](#) | [Yahoo! Argentina](#) | [Yahoo! Venezuela](#) | [Yahoo! Ecuador](#) | [Yahoo! Costa Rica](#) | [Yahoo! Panama](#) | [Yahoo! Uruguay](#) | [Yahoo! Paraguay](#) | [Yahoo! Bolivia](#) | [Yahoo! Cuba](#) | [Yahoo! Haiti](#) | [Yahoo! Dominican Republic](#) | [Yahoo! Puerto Rico](#) | [Yahoo! Trinidad and Tobago](#) | [Yahoo! Barbados](#) | [Yahoo! Guyana](#) | [Yahoo! Suriname](#) | [Yahoo! Guadeloupe](#) | [Yahoo! Martinique](#) | [Yahoo! French Guiana](#) | [Yahoo! Cayman Islands](#) | [Yahoo! Bermuda](#) | [Yahoo! Montserrat](#) | [Yahoo! Antigua and Barbuda](#) | [Yahoo! Saint Kitts and Nevis](#) | [Yahoo! Saint Lucia](#) | [Yahoo! Saint Vincent and the Grenadines](#) | [Yahoo! Grenada](#)

YAHOO! USER INTERFACE BLOG

News and Articles about Designing and Developing with Yahoo! Libraries

Blog

About

search in blog

Performance Research, Part 2: Browser Cache Usage - Exposed!

January 4, 2007 at 12:24 pm by Tami Theurer | In Development |

This is the second in a series of articles describing experiments conducted to learn more about optimizing web page performance. You may be wondering why you're reading a performance article on the YUI Blog. It turns out that most of web page performance is affected by front-end engineering, that is, the user interface design and development.

In an earlier post, I described [What the 80/20 Rule Tells Us about Reducing HTTP Requests](#). Since browsers spend 80% of the time fetching external components including scripts, stylesheets and images, reducing the number of HTTP requests has the biggest impact on reducing response time. But shouldn't everything be saved in the browser's cache anyway?

Why does cache matter?

It's important to differentiate between end user experiences for an empty versus a full cache page view. An "empty cache" means the browser bypasses the disk cache and has to request all the components to load the page. A "full cache" means all (or at least most) of the components are found in the disk cache and the corresponding HTTP requests are avoided.

The main reason for an empty cache page view is because the user is visiting the page for the first time and the browser has to download all the components to load the page. Other reasons include:

- The user visited the page previously but cleared the browser cache.
- The browser cache was automatically cleared based on the browser's settings.

SYNDICATE

All Stories:

[+ MY YOUTUBE](#) [RSS](#)

All Comments:

[+ MY YOUTUBE](#) [RSS](#)

RECENT POSTS

Welcoming Mike Lee and Der Glass to the YUI Team

YUI Theater - Doug Geoffrey: "From the Mouth of a Screenreader"

YUI Implementation Focus: Dustin Diaz's OED/Chain

Happy Birthday to the Pattern Library, Tool

YUI Theater - "Browser Wars Episode II: Attack of the DGMs"

Free Excerpt: Nicholas Zakas on YUI Connection Manager, from Professional Ajax 2nd Edition

RECENT READERS



tanvika

#4



Cookies

Set Scope Correctly

1



user requests
www.yahoo.com



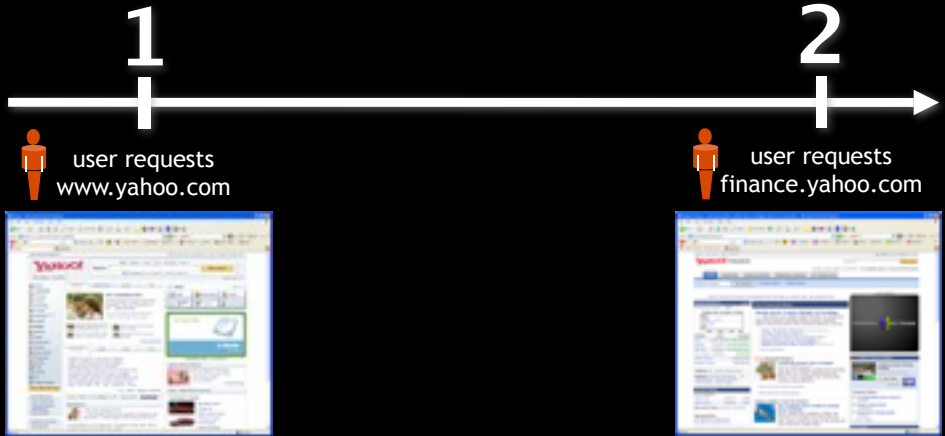
HTTP response header sent by the web server:

HTTP/1.1 200 OK

Content-Type: text/html; charset=utf-8

Set-Cookie: C=abcdefghijklmnpqrstuvwxyz; domain=.yahoo.com

Because broad scope adds up



HTTP request header sent by the browser:

```
GET / HTTP/1.1
Host: finance.yahoo.com
User-Agent: Mozilla/4.0 (compatible; MSIE 6.0; ...
Cookie: C=abcdefghijklmnopqrstuvwxyz;
```

Impact on Response Time

Cookie Size	Time	Delta
0 bytes	78 ms	0 ms
500 bytes	79 ms	+1 ms
1000 bytes	94 ms	+16 ms
1500 bytes	109 ms	+31 ms
2000 bytes	125 ms	+47 ms
2500 bytes	141 ms	+63 ms
3000 bytes	156 ms	+78 ms

Cookie Sizes across the Web

Total Cookie Size	
Amazon	60 bytes
Google	72 bytes
Yahoo	122 bytes
CNN	184 bytes
YouTube	218 bytes
MSN	268 bytes
eBay	331 bytes
MySpace	500 bytes

Experiment Takeaways

- eliminate unnecessary cookies
- keep cookie sizes low
- set cookies at the appropriate domain (or sub-domain) level
- set Expires date appropriately

http://www.yui-blog.com/2007/03/01/performance-research-part-3/

YUI: Yahoo! User Interface Library, JavaScript, Animation, DOM, CSS

http://www.yui-blog.com/2007/03/01/performance-research-part-3/

YUI: Yahoo! User Interface Library, JavaScript, Animation, DOM, CSS

YAHOO! USER INTERFACE BLOG

News and Articles about Designing and Developing with Yahoo! Libraries

Blog

About

search in blog

Performance Research, Part 3: When the Cookie Crumbles

March 1, 2007 at 4:41 pm by Tavis Thayer | In Development |

This article, co-written by Patty Chi, is the third in a series of articles describing experiments conducted to learn more about optimizing web page performance (Part 1, Part 2). You may be wondering why you're reading a performance article on the YUI Blog. It turns out that most of web page performance is affected by front-end engineering – that is, the user interface design and development.

HTTP cookies are used for a variety of reasons such as authentication and personalization. Information about cookies is exchanged in the HTTP headers between web servers and browsers. This article discusses the impact of cookies on the overall user response time.

HTTP Quick Review

Cookies originate from web servers when browsers request a page. Here is a sample HTTP header sent by the web server after a request for `www.yahoo.com`:

```
HTTP/1.1 200 OK
Content-Type: text/html; charset=utf-8
Set-Cookie: C=abcde; path=/; domain=.yahoo.com
```

The header includes information about the response such as the protocol version, status code, and content-type. The Set-Cookie is also included in the response and in this example the name of the cookie is "C" and the value of the cookie is "abcde". Note: The maximum size of a cookie is

SYNDICATE

All Series:



MY YOUTUBE

RSS

All Comments:



MY YOUTUBE

RSS

RECENT POSTS

Welcoming Mike Lee and Dan Glass to the YUI Team

YUI Theater - Doug Geoffrey: "From the Mouth of a Screen-reader"

YUI Implementation Focus: Dustin Diaz's ODE/Chain

Happy Birthday to the Pattern Library, Todd

YUI Theater - "Browser Wars Episode II: Attack of the DCMy"

Free Excerpt: Michael Dumas on YUI Connection Manager, from Professional Ajax 2nd Edition

RECENT READERS



tavis08

Parallel Downloads

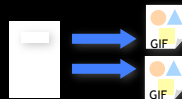
Two components



in parallel



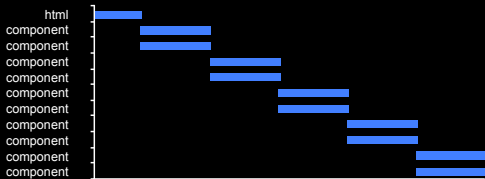
per hostname



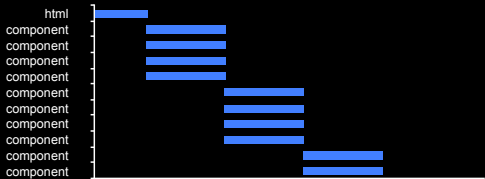
per HTTP/1.1

Parallel Downloads

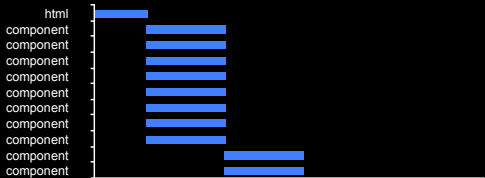
Two in parallel



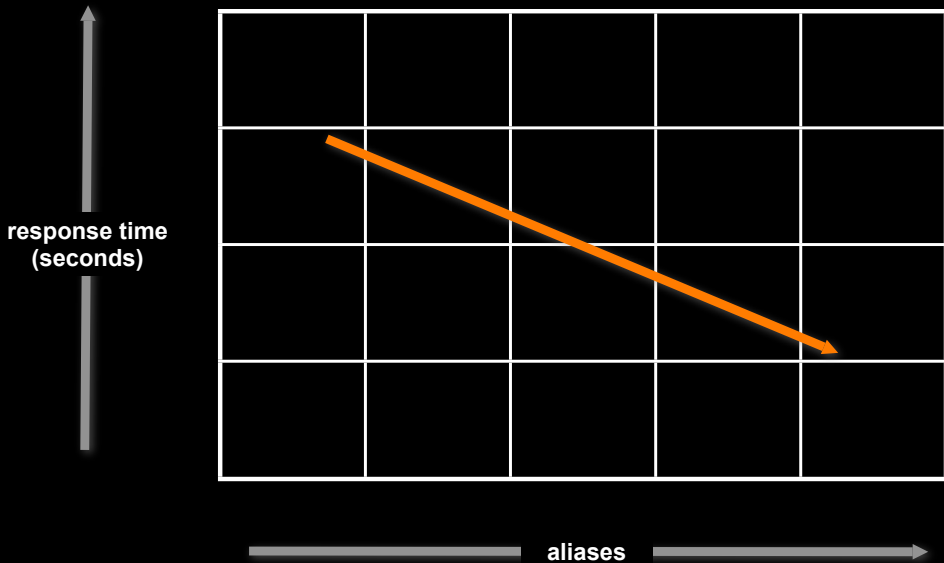
Four in parallel



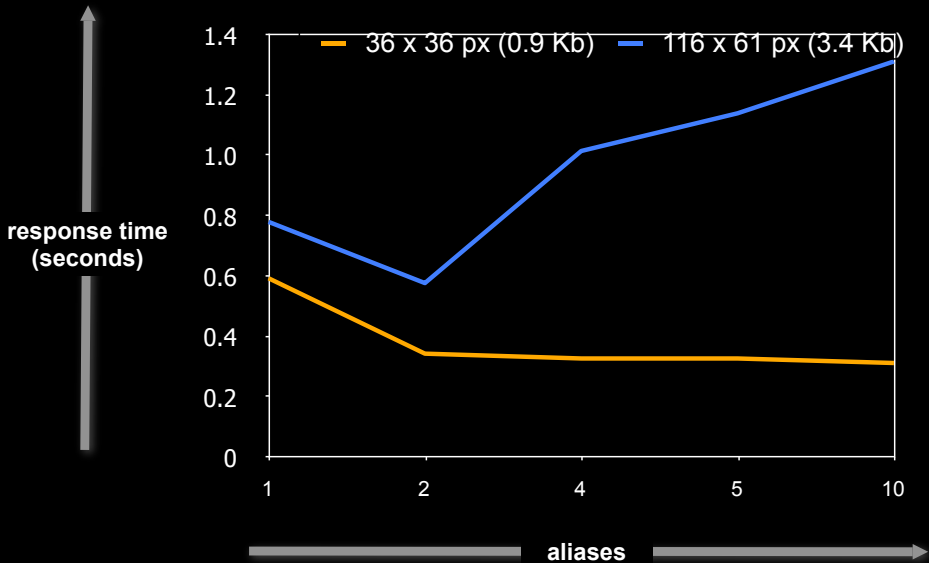
Eight in parallel



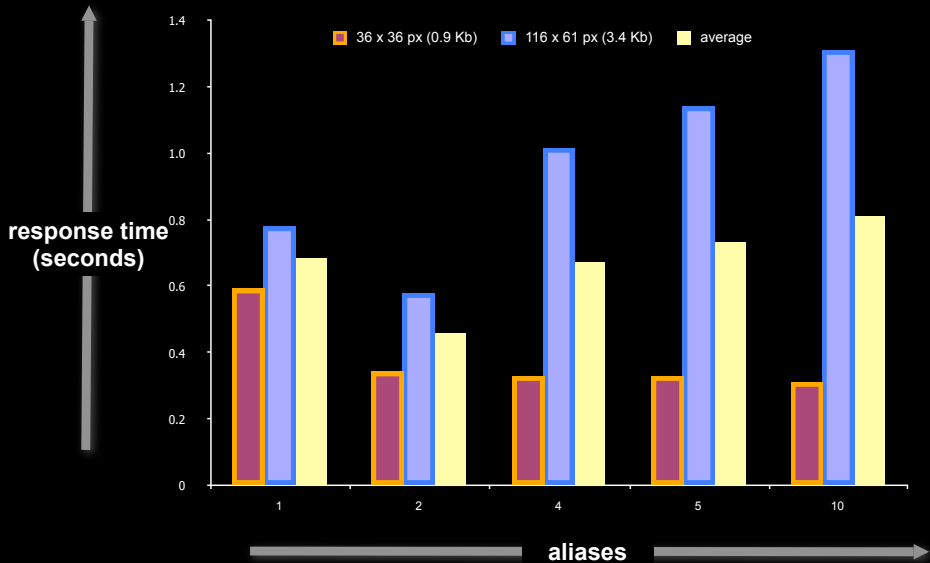
Maximizing Parallel Downloads



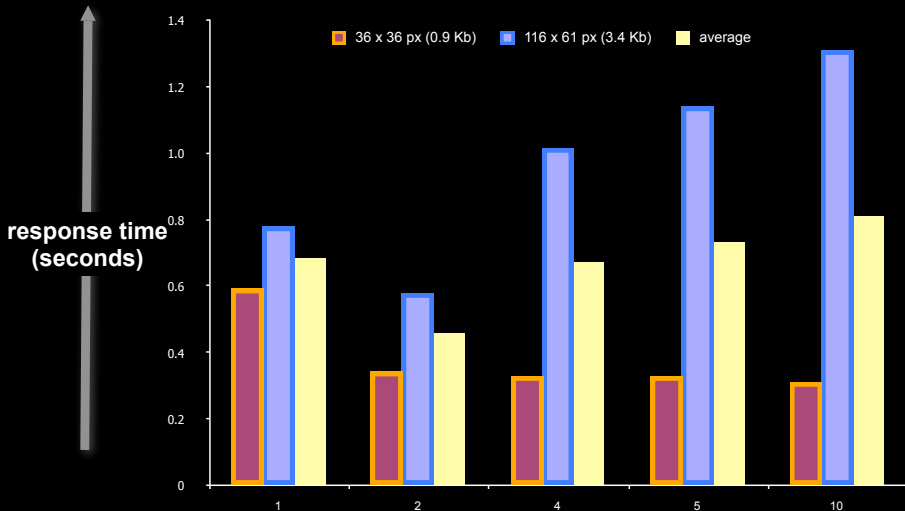
Maximizing Parallel Downloads



Maximizing Parallel Downloads



Maximizing Parallel Downloads



rule of thumb: use at least 2 but no more than 4 aliases

Experiment Takeaways

- beware the effects of CPU thrashing
- DNS lookup times vary across ISPs and geographic locations
- domain names may not be cached

YAHOO! USER INTERFACE BLOG

News and Articles about Designing and Developing with Yahoo! Libraries



Blog

About

search in blog

Performance Research, Part 4: Maximizing Parallel Downloads in the Carpool Lane

April 15, 2007 at 11:47 am by Tanni Theurer (in Development)

This article, co-written by Steve Sauders, is the fourth in a series of articles describing experiments conducted to learn more about optimizing web page performance (Part 1, Part 2, Part 3). You may be wondering why you're reading a performance article on the YUI Blog: it turns out that most of web page performance is affected by front-end engineering, that is, the user interface design and development.

Parallel Downloads

The biggest impact on end-user response times is the number of components in the page. Each component requires an extra HTTP request, perhaps not when the cache is full, but definitely when the cache is empty. Knowing that the browser performs HTTP requests in parallel, you may ask why the number of HTTP requests affects response time. Can't the browser download them all at once?

The explanation goes back to the HTTP/1.1 spec, which suggests that browsers download two components in parallel per hostname. Many web pages download all their components from a single hostname. Viewing these HTTP requests reveals a stair-step pattern, as shown

SYNDICATE

All Entries:

[RSS](#) [ATOM](#)

All Comments:

[RSS](#) [ATOM](#)

RECENT POSTS

[Performance Research, Part 4: Maximizing Parallel Downloads in the Carpool Lane](#)

[JSON and Browser Security](#)

[YUI Version 2.2.1 Released, and Graded Browser Support Update](#)

[Welcoming Mike Lee and Dev Goss to the YUI Team](#)

[YUI Theater - Doug Geoffrey: "From the Mouth of a Screenreader"](#)

[YUI Implementation Focus: Dustin Diaz's DED iChain](#)

RECENT READERS

Summary

What the 80/20 Rule Tells Us about Reducing HTTP Requests

<http://yuiblog.com/blog/2006/11/28/performance-research-part-1/>

Browser Cache Usage – Exposed!

<http://yuiblog.com/blog/2007/01/04/performance-research-part-2/>

When the Cookie Crumbles

<http://yuiblog.com/blog/2007/03/01/performance-research-part-3/>

Maximizing Parallel Downloads in the Carpool Lane

<http://yuiblog.com/blog/2007/04/11/performance-research-part-4/>

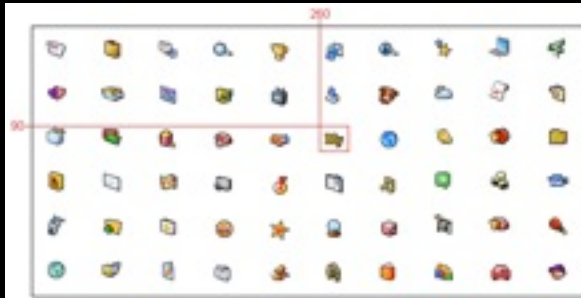


Performance Fruit (some low, some high)

Rule: Make fewer HTTP requests.

- CSS sprites
- Combined / concatenated JS and CSS files
- image maps
- inline (data) images

CSS Sprites



```
<span style="
background-image: url('sprites.gif');
background-position: -260px -90px;">
</span>
```

size of combined image is less

Combined Scripts, Combined Stylesheets

	Scripts	Stylesheets
amazon.com	3	1
aol.com	18	1
cnn.com	11	2
ebay.com	7	2
froogle.google.com	1	1
msn.com	9	1
myspace.com	2	2
wikipedia.org	3	1
yahoo.com	4	1
youtube.com	7	3
Average	6.5	1.5

(This could well be)
The End.

Rule: Use a CDN

amazon.com

Akamai

aol.com

Akamai

cnn.com

ebay.com

Akamai, Mirror Image

google.com

msn.com

SAVVIS

myspace.com

Akamai, Limelight

wikipedia.org

yahoo.com

Akamai

youtube.com

distribute your static content before
distributing your dynamic content

Rule: Add an Expires header

(not just for images)

	Images	Stylesheets	Scripts	%	Median Age
amazon.com	0/62	0/1	0/3	0%	114 days
aol.com	23/43	1/1	6/18	48%	217 days
cnn.com	0/138	0/2	2/11	1%	227 days
ebay.com	16/20	0/2	0/7	55%	140 days
froogle.google.com	1/23	0/1	0/1	4%	454 days
msn.com	32/35	1/1	3/9	80%	34 days
myspace.com	0/18	0/2	0/2	0%	1 day
wikipedia.org	6/8	1/1	2/3	75%	1 day
yahoo.com	23/23	1/1	4/4	100%	n/a
youtube.com	0/32	0/3	0/7	0%	26 days

Rule: Gzip components

- you can affect users' download times
- 90%+ of browsers support compression

Gzip: not just for HTML

	HTML	Scripts	Stylesheets
amazon.com	x		
aol.com	x	some	some
cnn.com			
ebay.com	x		
froogle.google.com	x	x	x
msn.com	x	deflate	deflate
myspace.com	x	x	x
wikipedia.org	x	x	x
yahoo.com	x	x	x
youtube.com	x	some	some

gzip scripts, stylesheets, XML, JSON
(not images, PDF)

YAHOO! USER INTERFACE BLOG

News and Articles about Designing and Developing with Yahoo! Libraries

Blog

About

search in blog

Free Hosting of YUI Files from Yahoo!

February 22, 2007 at 9:18 pm by Nate Koechley (in Development)

Coinciding with this week's release of YUI version 2.2.0, the one year anniversary of the YUI open-source release, and as announced at the YUI Party just moments ago, we're opening up free YUI hosting from the Yahoo! network to all YUI implementers. If you're using YUI for your own project, we'll serve the files for you — gzipped, with good cache-control, using our state-of-the-art network, for free. You can count on these files being continuously available because they're the same files, served by the same source, that we use for most YUI implementations at Yahoo!.

Files served from Yahoo!'s network include version numbers in filepaths, allowing you to reference a specific version in your code. Previous versions are retained even as new versions are released. While we are providing no explicit SLA with respect to the availability of legacy code, our current policy is to support permanent availability of legacy YUI files.

Why Provide YUI Hosting on Yahoo!'s Network?

We're opening up the service of YUI from Yahoo! servers for the same reasons we open-sourced YUI in February: Yahoo! is quintessentially a web company. The progress being made by developers in richness and usability today is healthy for the web and, by extension, good for Yahoo! We want to do everything we can do to enhance that evolution — whether it's opening up YUI, hosting YUI files, or creating best-of-breed APIs like the recently-announced Browser-Based Authentication system.

SYNDICATE

All Posts

[BY YAHOO!](#) RSS

All Comments

[BY YAHOO!](#) RSS

All Development Posts

[BY YAHOO!](#) RSS

All Design Posts

[BY YAHOO!](#) RSS

YUI Theater Posts

[BY YAHOO!](#) RSS

"In the Wild" Posts

[BY YAHOO!](#) RSS

Performance Posts

[BY YAHOO!](#) RSS

RECENT POSTS

An Interview with Ted Husted,
Creator of YUI Community Site
"Rune Yazaar"

Douglas Crockford To Speak at
the Yahoo! Widgets Conference
on "JavaScript: The Good Parts"

YUI Theater — Grady Booch:
"The Promise, the Limits, the
Beauty of Software"

YUI's Free Hosting:

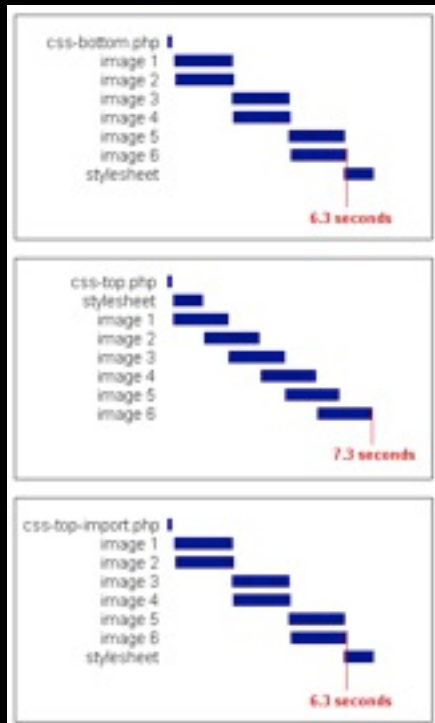
- Aggregated files
- With Expires headers
- On a CDN
- Gzipped

Rule: Put CSS at the top

- stylesheets block rendering in IE
- solution: put stylesheets in HEAD (per spec)
- avoids Flash of Unstyled Content

use `<link>` (not `@import`)

Slowest is
actually the
Fastest



Rule: Choose `<link>` over `@import`

- When you use `@import` you override the browsers' native performance assistance.

Rule: Move scripts to the bottom

- scripts block rendering of everything below them in the page
- scripts block parallel downloads across all hostnames



What about `defer`?

`script defer` attribute is not a solution

- blocks rendering and downloads in FF
- slight blocking in IE

Rule: Avoid CSS expressions

Can be used to set CSS properties dynamically in IE

```
width: expression(  
    document.body.clientWidth < 600 ?  
    "600px" : "auto" );
```

But problematic because expressions execute many times

- mouse move, key press, resize, scroll, etc.

Rule: Make JS & CSS external

Inline: bigger HTML but no HTTP request

External: cachable but extra HTTP

Variables:

- page views per user (per session)
- empty vs. full cache stats
- component re-use

External is typically better

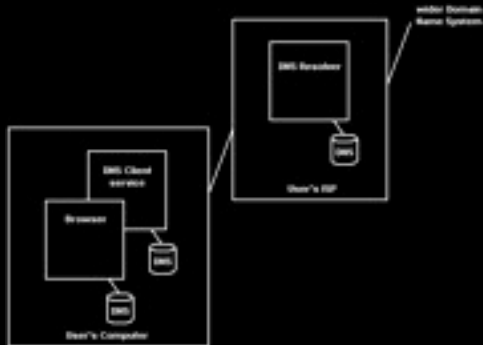
- home pages may be an exception due to cache behavior of browser's startpage.

Rule: Reduce DNS lookups

typically 20-120 ms

block parallel downloads

OS and browser both have DNS caches



Best practice:

Max 2-4 hosts

Use keep-alive

Rule: Minify JavaScript

	Minify External?	Minify Inline?
www.amazon.com	no	no
www.aol.com	no	no
www.cnn.com	no	no
www.ebay.com	yes	no
froogle.google.com	yes	yes
www.msn.com	yes	yes
www.myspace.com	no	no
www.wikipedia.org	no	no
www.yahoo.com	yes	yes
www.youtube.com	no	no

don't forget to minify inline scripts, too

Which Compressor?

- YUI Compressor is highly regarded
- web search “CompressorRater”

Rule: Avoid redirects

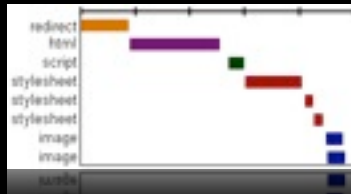
3xx status codes – mostly 301 and 302

HTTP/1.1 301 Moved Permanently

Location: <http://yahoo.com/newuri>

add Expires headers so redirect headers are cached

Redirects are worst form of blocking



<http://www.w3.org/Protocols/rfc2616/rfc2616-sec10.html>

Rule: Make AJAX cacheable and small

- XHR, JSON, iframe, dynamic scripts can still be cached, minified, and gzipped
- a personalized response should still be cacheable by that person

Rule: Use GET for AJAX Requests

- POST is a two-step process.
- GET can send in one packet. Use if possible.
- Max payload is 2K in IE.

Rule: Post-load Components

- Delay all non-crucial elements.
 - YUI Image Loader offers precise control.
 - YUI Get makes on-the-fly JS & CSS easy.
- Used on www.yahoo.com homepage.

Rule: Preload Components

- Unconditional preload - all following onload
 - Ex: google.com's sprite
- Conditional preload - based on user action
 - Ex: search.yahoo.com's Search Assist
- Anticipated preload - give redesigns a boost

Rule: Minimize iFrames

- Costly even if blank
- Blocks page onload
- Questionable semantic value

Rule: Smart Event Handles

- Event Delegation to reduce total number
- onAvailable (etc) for earlier access
- YUI Event has onAvailable, onContentLoaded, & onDomReady

Rule: Optimize CSS Sprites

- Horizontal alignment is usually smaller than vertical.
- Combine similar colors in a sprite to minimize color count.
 - Only 256 colors can fit in a PNG8.
- Be mobile-friendly. Avoid big gaps between images. Minimal impact on k-size, but takes more memory to decompress.

Rule: favicon.ico

- Always provide a favicon.ico so the request doesn't 404
- And the request carries cookies
- Mitigate by setting an expires header,
 - (but NOT forever since you can't rename the file.)
- Keep under 1K

Rule: iPhone-Friendly Tip

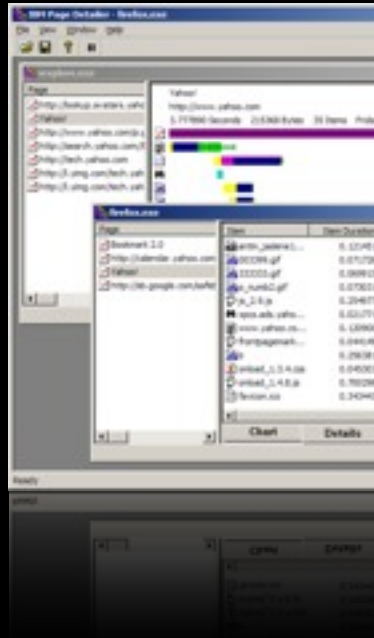
- iPhone will not cache assets larger than 25K
 - That's 25K uncompressed.
- <http://yuiblog.com/blog/2008/02/06/iphone-cacheability/>

Tools for your own
analysis

IBM Page Detailer

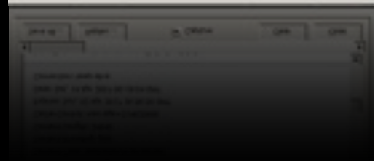
Packet sniffer. Windows only (IE, FF, any .exe). Free trial, \$300 license

<http://alphaworks.ibm.com/tech/pagedetailer>



LiveHTTPHeaders

View HTTP headers. Firefox extension. Free.



Firebug

- web development evolved
- inspect and edit HTML, tweak and visualize CSS, debug and profile JavaScript
- monitor network activity (caveat: caching)
- Free Firefox extension

Yahoo! - Mozilla Firefox

File Edit View History Bookmarks Tools Help

http://www.yahoo.com/

Make Y! your home page

Yahoo! Voice: Make international calls for as low as 1¢/min.

at&t YAHOO! HIGH SPEED INTERNET

My Yahoo! My Mail

Web Images Video Local Shopping more ▾

Search: Web Search

11 Answers Ask a question Answer questions

Page Options ▾



Autos Calendar Finance Games GeoCities Groups HotJobs Maps Movies Music

Featured Entertainment Sports Life

Apr 14, 2007

 **Disturbia thoughts**

Find out what Yahoo! users think of "Disturbia" – a modern twist on the classic "Rear Window." – Reviews

- Showtimes  Trailer  Photos
- LaBeouf to star in "Indiana Jones 4"

 Users review the new thriller "Disturbia"

 Scientists: Scrap the Internet and start over

Hi, Steven Sign Out

Mail Messenger Radio

Weather 85°F Local Horoscopes

Get the BIGGEST REFUND and do your taxes for FREE. **TurboTax** Free Edition

GET IT DONE NOW!

Inspect Edit **swf2** { <div#ed.ad <div.colpadding <div#right <div#rightbox <div#colcx <div#page <body.youde <html

Console HTML CSS Script DOM Net YSlow Code

Options ▾

```

<div class="colpadding">
  <div id="pa" class="ad">
    <script type="text/javascript">
      <div id="ad" class="ad">
        <script language="javascript">
          <script src="http://us.js2.yimg.com/us.yimg.com/a/1-/java/pw">
          <script language="javascript">
            <div id="inbdiv" style="margin-top: 2px;">
              <script language="javascript">
                <script>img width=1 height=1 alt=""
                  src="http://us.br.yahoo.com/b/P/TPnA9j8esGfHrrgRi8oEDw27EsD/
            </div>
          <script src="text/javascript">

```

Style Layout DOM Options ▾

body { www.yahoo.com (line 134)

line-height: 1.22em;

Inherited from **div#ed.ad**

.ad { www.yahoo.com (line 2535)

text-align: center;

body { www.yahoo.com (line 134)

line-height: 1.22em;

Inherited from **div.colpadding**

body { www.yahoo.com (line 134)

5.820s YSlow

YSlow

- Extension for Firebug extension
- Performance report card
- HTTP/HTML summary
- List of components in the page
- Tools including JSLint



HIGH SPEED INTERNET

My Yahoo!

My Mail

Web Images Video Local Shopping more

Search:

Web Search

Y! Answers: Ask a question | Answer questions

Page Options

- Autos
- Calendar
- Finance
- Games
- GeoCities
- Groups
- HotJobs
- Maps
- Movies
- Music

Featured

Entertainment

Sports

Life

Apr 14, 2007



Disturbing thoughts

Find out what Yahoo! users think of "Disturbia" - a modern twist on the classic "Rear Window" - [Reviews](#)

- Shorttimes
- Trailer
- Photos
- LeBeau to star in "Indiana Jones 4"

Users review the new thriller "Disturbia"

Scientists: Scrape the Internet and start over

Hi, Steven

Sign Out

Mail

Messenger

Radio

Weather 81°F

Local

Horoscopes

Get the BIGGEST REFUND and do your taxes for FREE.

TurboTax
Free Edition

GET IT DONE NOW!

Inspect Performance Stats Components Tools
 Console HTML CSS Script DOM Net YSlow CodeView

Performance Grade: A (94)

Expand All Collapse All

1. Minimize HTTP requests
 This page has 4 external JavaScript files.
2. Use edge computing
3. Add an Expires header
4. Gzip components
 These components are not gzipped:
 (1.9K) http://1.yimg.com/es.js yimg.com/lib/ba/ba_2.0.3.js
5. Move CSS to the top

Conclusion

Takeaways

- focus on the front-end
- harvest the low-hanging fruit
 - **reduce HTTP requests**
 - enable caching
- you *do* control user response times
- LOFNO – be an advocate for your users

For more information:

High Performance Web Sites

Essential Knowledge for
Front-End Engineers

by Steve Souders,
with research from Yahoo!'s
Performance Engineering team.



Exceptional Performance on YDN

<http://developer.yahoo.com/performance>

YAHOO! DEVELOPER NETWORK

Exceptional Performance

Performance | B

Best Practices

Yahoo!'s Exceptional Performance team evangelizes best practices for improved research, build tools, write practices and blogs, and speak at conferences. Their high-performance web sites.

Best Practices

The Exceptional Performance team has identified a number of best practices. 34 best practices divided into 7 categories.

Filter rules by category: **General** **Tools** **Build** **Test** **Performance**

1. Reduce HTTP Requests
2. Reduce DNS Lookups
3. Avoid Redirects
4. Minify Assets
5. Use Async Components
6. Throttle Components
7. Reduce the Number of DOM Elements
8. Split Components Across Domains
9. Minimize the Number of Frames
10. No K&M

Follow for Finding

Follow analysis web pages and tell you why they're slow based on the best practices. Follow is a Firefox add-on integrated with the popular Firefox web-developer performance on over 100 Yahoo! properties by 20-30%. We're pleased to be so community as that others can pick up it accelerating the user experience of the

Articles on YUIBlog

[http://yuiblog.com/blog/
category/performance](http://yuiblog.com/blog/category/performance)



Let's keep talking...

nate@koechley.com
<http://nate.koechley.com/blog>
"natekoechley"

Thanks again to Yahoo!'s Exceptional Performance Engineering Team

Links

book: <http://www.oreilly.com/catalog/9780596514211/>

examples: <http://stevesouders.com/examples/>

image maps: <http://www.w3.org/TR/html401/struct/objects.html#h-13.6>

CSS sprites: <http://alistapart.com/articles/sprites>

inline images: <http://tools.ietf.org/html/rfc2397>

jsmin: <http://crockford.com/javascript/jsmin>

dojo compressor: <http://dojotoolkit.org/docs/shrinksafe>

HTTP status codes: <http://www.w3.org/Protocols/rfc2616/rfc2616-sec10.html>

IBM Page Detailer: <http://alphaworks.ibm.com/tech/pagedetailer>

Fasterfox: <http://fasterfox.mozdev.org/>

LiveHTTPHeaders: <http://livehttpheaders.mozdev.org/>

Firebug: <http://getfirebug.com/>

YUIBlog: <http://yuiblog.com/blog/2006/11/28/performance-research-part-1/>

<http://yuiblog.com/blog/2007/01/04/performance-research-part-2/>

<http://yuiblog.com/blog/2007/03/01/performance-research-part-3/>

<http://yuiblog.com/blog/2007/04/11/performance-research-part-4/>

YDN: http://developer.yahoo.net/blog/archives/2007/03/high_performanc.html

http://developer.yahoo.net/blog/archives/2007/04/rule_1_make_few.html

CC Images Used

“Zipper Pocket” by [jogales](http://www.flickr.com/photos/jogales/11519576/): <http://www.flickr.com/photos/jogales/11519576/>

“Need for Speed” by [Amnemona](http://www.flickr.com/photos/marinacvinhal/379111290/): <http://www.flickr.com/photos/marinacvinhal/379111290/>

“I wonder what flavour it is?” by [blather](http://www.flickr.com/photos/deadlyphoto/411770353/): <http://www.flickr.com/photos/deadlyphoto/411770353/>

“takeout boxes from Grand Shanghai” by [massdistraction](http://www.flickr.com/photos/sharynmorrow/11263821/): <http://www.flickr.com/photos/sharynmorrow/11263821/>

“takeout” by [dotpolka](http://www.flickr.com/photos/dotpolka/249129144/): <http://www.flickr.com/photos/dotpolka/249129144/>

“ice cream cone melting/rome” by [Megandavid](http://www.flickr.com/photos/megandavid/189332042/): <http://www.flickr.com/photos/megandavid/189332042/>

“nikon em bokeh” by [dsevilla](http://www.flickr.com/photos/dsevilla/249202834/): <http://www.flickr.com/photos/dsevilla/249202834/>

“maybe” by [Tal Bright](http://www.flickr.com/photos/bright/118197469/): <http://www.flickr.com/photos/bright/118197469/>

“how do they do that” by [Fort Photo](http://www.flickr.com/photos/fortphoto/388825145/): <http://www.flickr.com/photos/fortphoto/388825145/>

“Gorgeous iceberg 7 [Le Toit du Monde]” by [Adventure Addict](http://www.flickr.com/photos/adventureaddict/35290307/): <http://www.flickr.com/photos/adventureaddict/35290307/>

“molasses-spice cookies” [ilmungo](http://www.flickr.com/photos/ilmungo/65345233/): <http://www.flickr.com/photos/ilmungo/65345233/>

“Driving is fun” by [Ben McLeod](http://www.flickr.com/photos/benmcleod/59948935/): <http://www.flickr.com/photos/benmcleod/59948935/>

“Dozen eggs” by [aeA](http://www.flickr.com/photos/raeallen/96238870/): <http://www.flickr.com/photos/raeallen/96238870/>

“Max speed 15kmh” by [xxxtoff](http://www.flickr.com/photos/xxxtoff/219781763/): <http://www.flickr.com/photos/xxxtoff/219781763/>

“Stairway to heaven” [ognita](http://www.flickr.com/photos/ognita/503915547/): <http://www.flickr.com/photos/ognita/503915547/>

“flaps up” by <http://www.flickr.com/photos/jurvetson/74274113/>

“Fast Cat” by <http://www.flickr.com/photos/raylopez/708023176/>

nate@koechley.com

<http://nate.koechley.com/blog>

natekoechley = twitter,
delicious, flickr, tripit, last.fm, linkedin,
fireeagle, friendfeed, ... everywhere.

Thanks again to Yahoo!'s Exceptional Performance Engineering Team