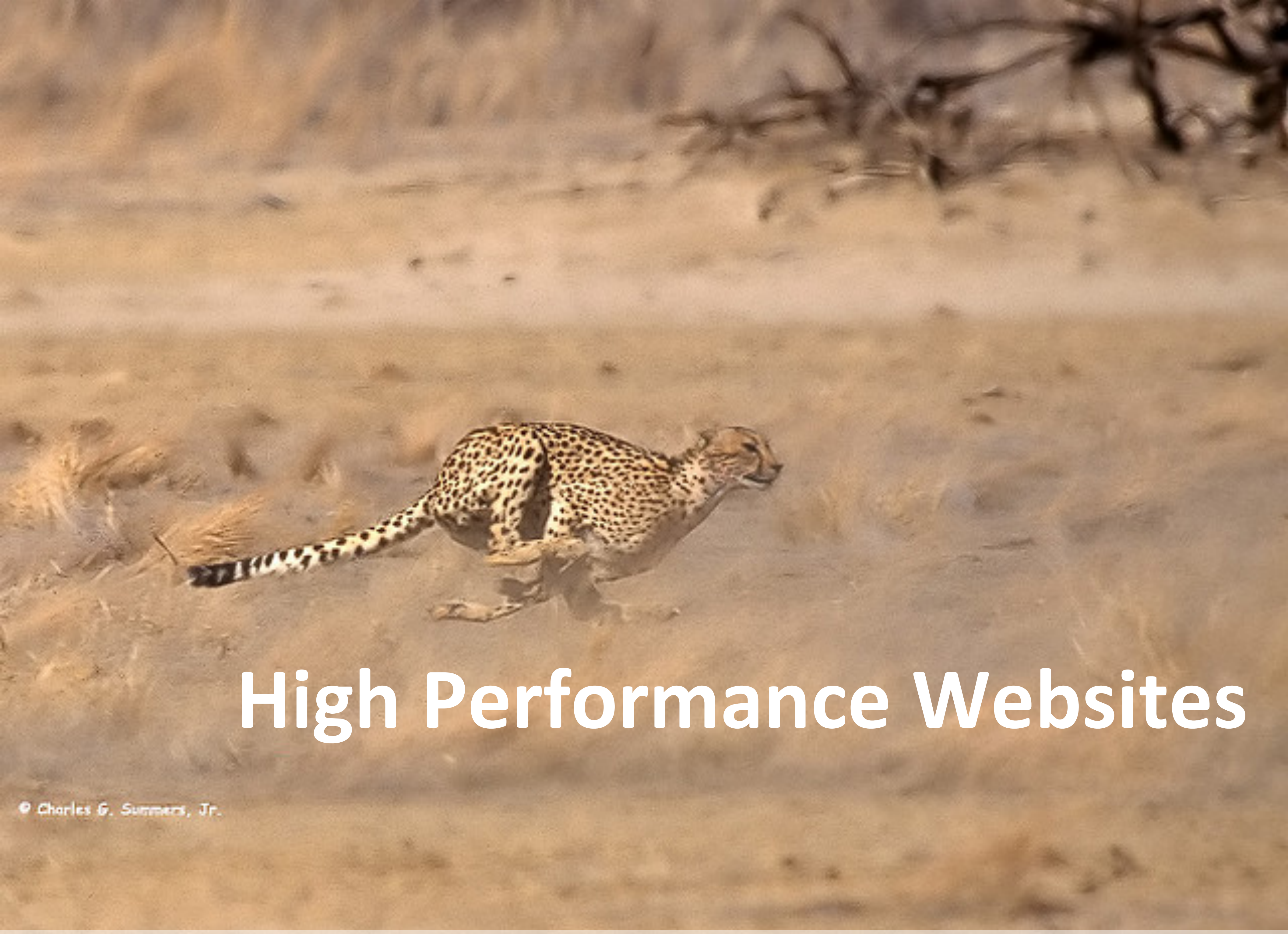


High Performance Websites

http://www.flickr.com/photos/wild_images/1147361271

Please wait.

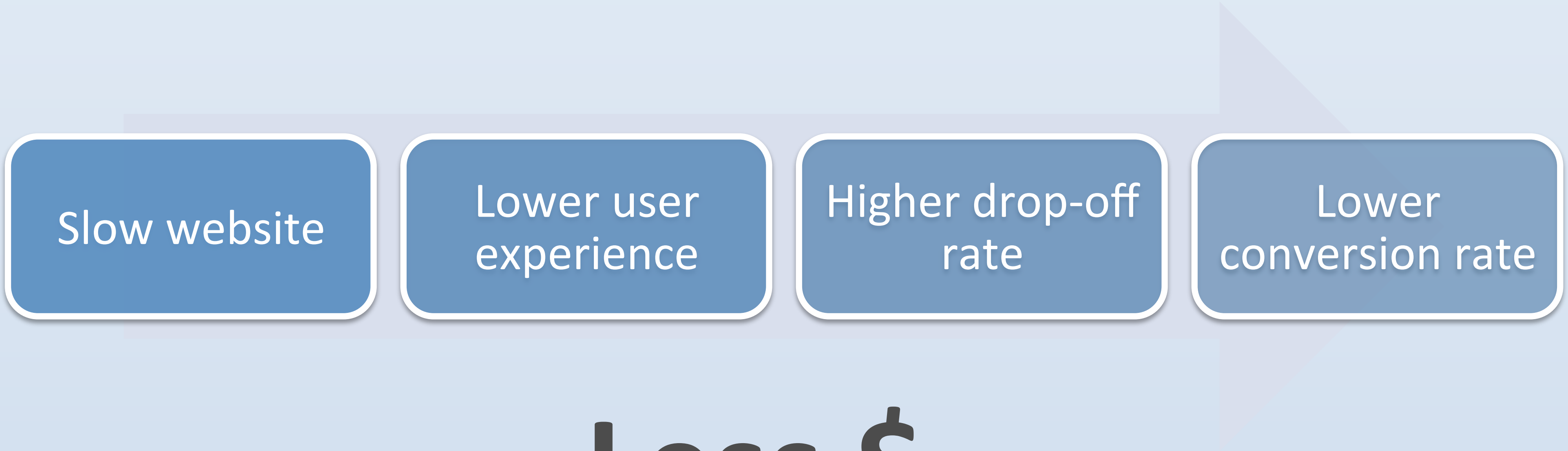
Loading...

80% of the end-user response time is spent on the front-end.

Most of this time is downloading images, stylesheets, scripts, Flash, etc.

Reducing the number and size of HTTP requests is key to faster pages.

Why bother?



Less $

Google +500ms	→	-20% revenue
Amazon +100 ms	→	-1% sales
Yahoo +400 ms	→	-5-9% page traffic

Why slow?

Connection bandwith

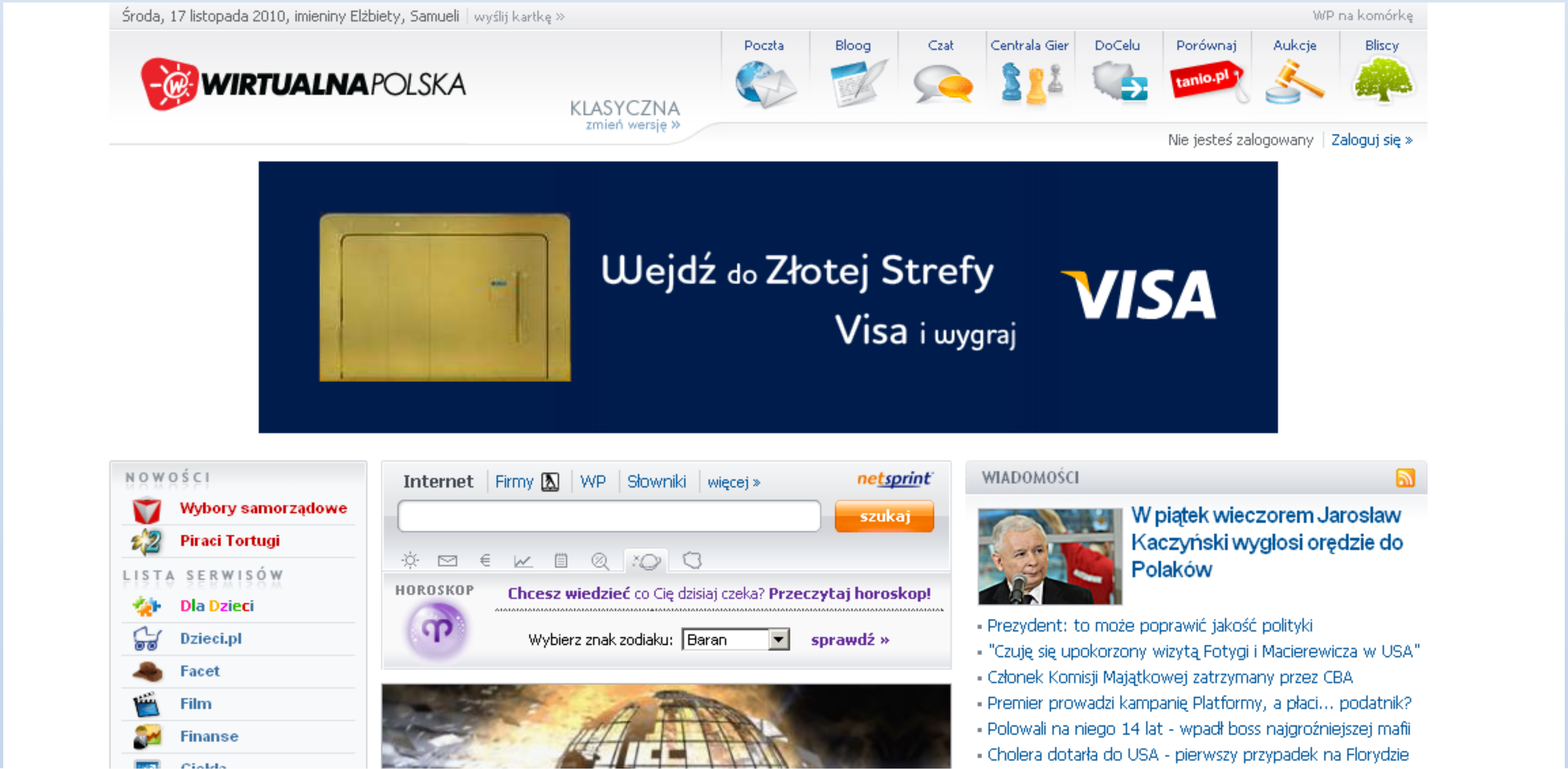
Connection lag

Client side computation

Page size

Number of requests

JS Code/CSS/# of reflows



Środa, 17 listopada 2010, imieniny Elżbiety, Samueli | wyślij kartkę »

WP na komórkę

WIRTUALNAPOLSKA

KLASYCZNA
zmień wersję »

Poczta | Bloog | Czat | Centrala Gier | DoCelu | Porównaj | Aukcje | Bliscy

tanio.pl

Nie jesteś zalogowany | Zaloguj się »

Wejdź do Złotej Strefy
Visa i wygraj
VISA

NOWOŚCI
- Wybory samorządowe
- Piraci Tortugi

LISTA SERWISÓW
- Dla Dzieci
- Dzieci.pl
- Facet
- Film
- Finanse

Internet | Firmy | WP | Słowniki | więcej »
netsprint
szukaj

HOROSKOP
Chcesz wiedzieć co Cię dzisiaj czeka? Przeczytaj horoskop!
Wybierz znak zodiaku: Baran
sprawdź »

WIADOMOŚCI

W piątek wieczorem Jarosław Kaczyński wygłosi orędzie do Polaków

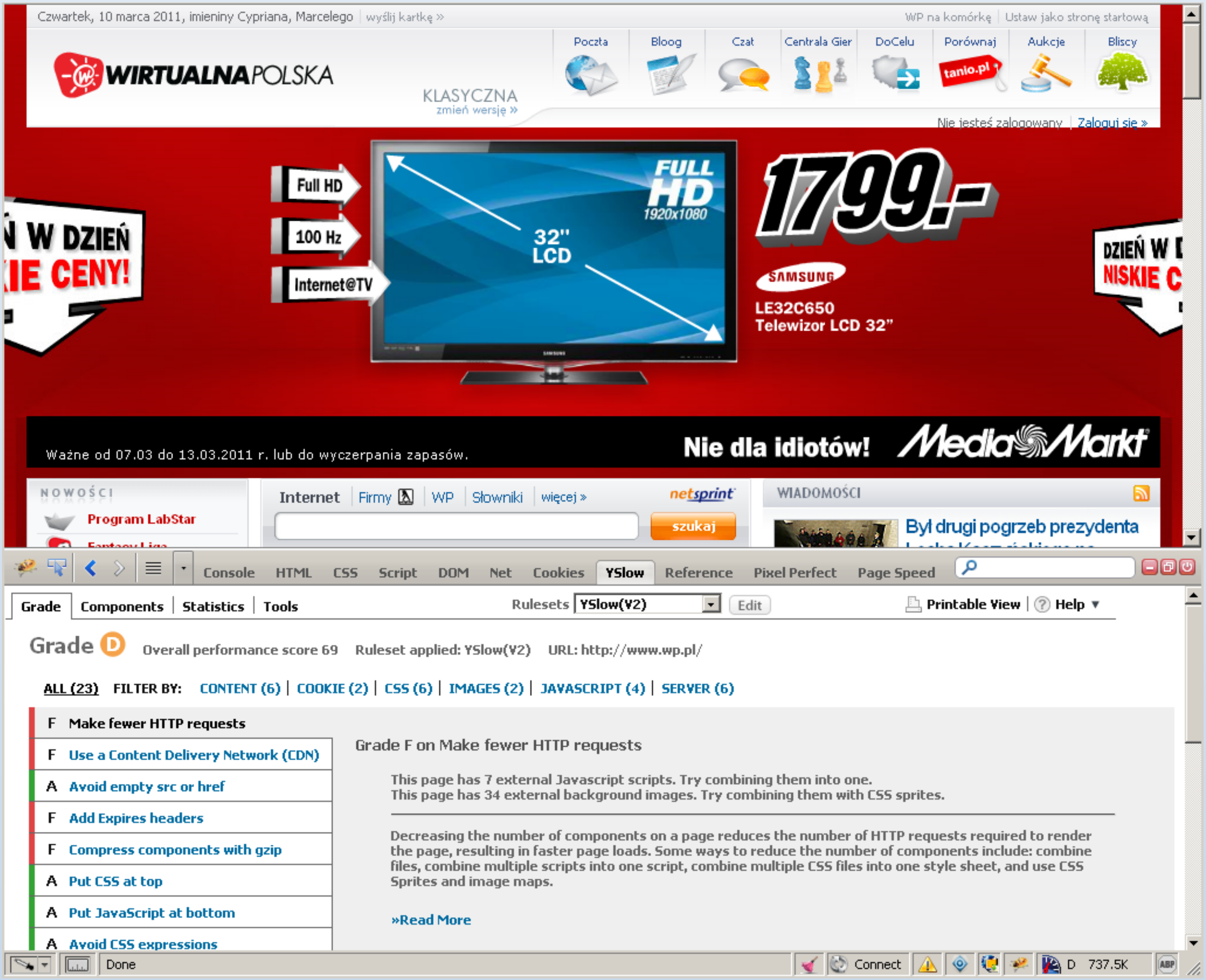
- Prezydent: to może poprawić jakość polityki
- "Czuję się upokorzony wizytą Fotygi i Macierewicza w USA"
- Członek Komisji Majątkowej zatrzymany przez CBA
- Premier prowadzi kampanię Platformy, a płaci... podatnik?
- Polowali na niego 14 lat - wpadł boss najgroźniejszej mafii
- Cholera dotarła do USA - pierwszy przypadek na Florydzie

What can be done?

1. Make fewer HTTP requests
2. Use a CDN
3. Add an Expires header
4. Gzip components
5. Put stylesheets at the top
6. Put scripts at the bottom
7. Avoid CSS expressions
8. Make JS and CSS external
9. Reduce DNS lookups
10. Minify JS
11. Avoid redirects
12. Remove duplicate scripts
13. Configure ETags
14. Make AJAX cacheable
15. Split the initial payload
16. Load scripts without blocking
17. Don’t scatter inline scripts
18. Split dominant content domains
19. Make static content cookie-free
20. Reduce cookie weight
21. Flushing the document early
22. Using iframes sparingly
23. Simplifying CSS Selectors

What can be done. Diagnosis.

Yslow


Czwartek, 10 marca 2011, imieniny Cypriana, Marcelego | wyślij kartkę »
WP na komórkę | Ustaw jako stronę startową
WIRTUALNAPOLSKA
KLASYCZNA
zmień wersję »
Poczta
Bloog
Czat
Centrala Gier
DoCelu
Porównaj
Aukcje
Bliscy
Nie jesteś zalogowany | Zaloguj się »
Full HD
100 Hz
Internet@TV
32" LCD
FULL HD 1920x1080
1799.-
SAMSUNG
LE32C650
Telewizor LCD 32"
W DZIEŃ
IE CENY!
DZIEŃ W
NISKIE C
Ważne od 07.03 do 13.03.2011 r. lub do wyczerpania zapasów.
Nie dla idiotów!
MediaMarkt
NOWOŚCI
Program LabStar
Internet | Firmy | WP | Słowniki | więcej »
netsprint
szukaj
WIADOMOŚCI
Był drugi pogrzeb prezydenta
Console HTML CSS Script DOM Net Cookies YSlow Reference Pixel Perfect Page Speed
Grade | Components | Statistics | Tools
Rulesets YSlow(V2) Edit
Printable View | Help
Done
Connect
D 737.5K

Grade D Overall performance score 69 Ruleset applied: YSlow(V2) URL: http://www.wp.pl/

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

F	Make fewer HTTP requests
F	Use a Content Delivery Network (CDN)
A	Avoid empty src or href
F	Add Expires headers
F	Compress components with gzip
A	Put CSS at top
A	Put JavaScript at bottom
A	Avoid CSS expressions

Grade F on Make fewer HTTP requests

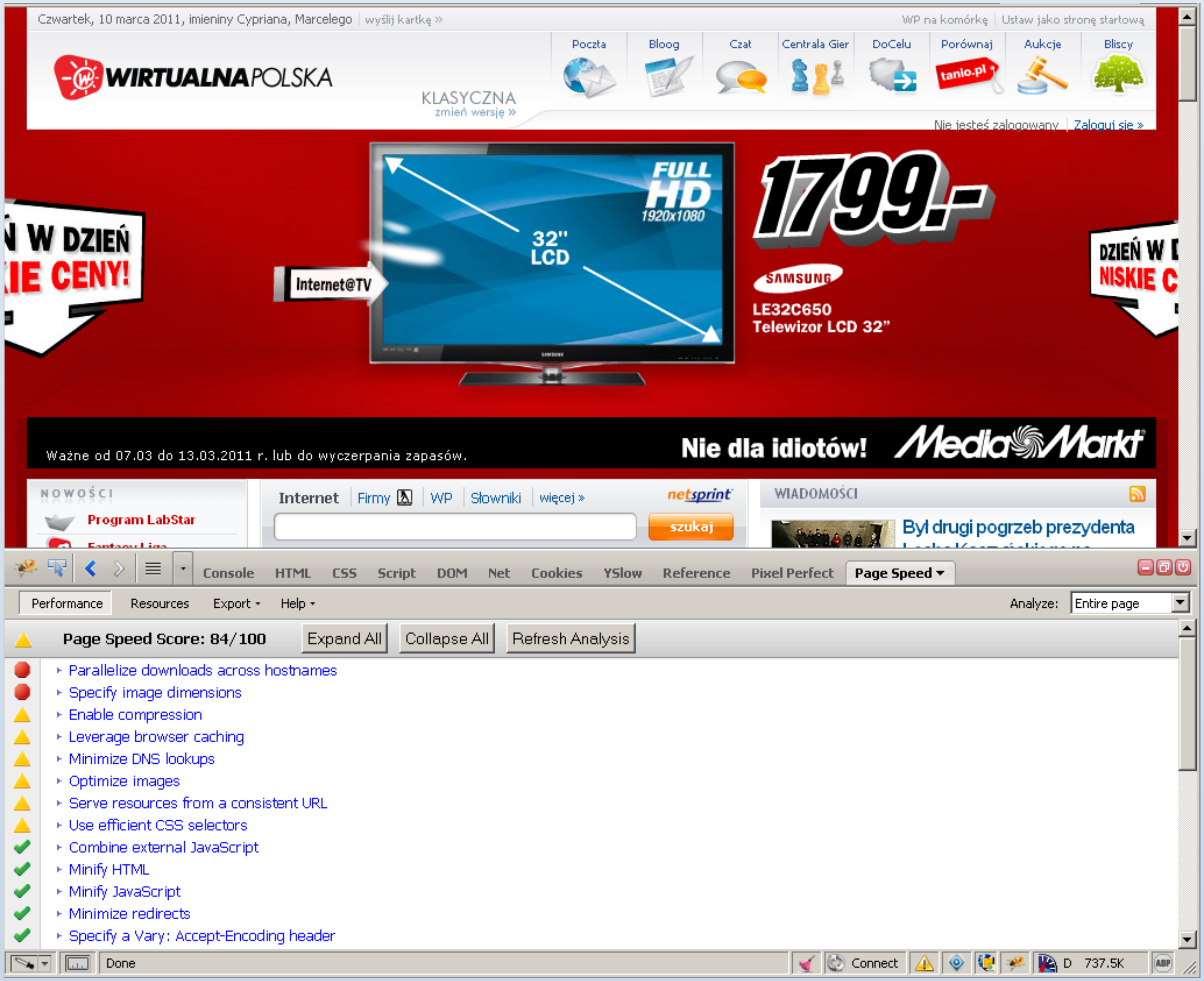
This page has 7 external Javascript scripts. Try combining them into one.
This page has 34 external background images. Try combining them with CSS sprites.

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 reduce the number of components include: combine files, combine multiple scripts into one script, combine multiple CSS files into one style sheet, and use CSS Sprites and image maps.

»Read More

What can be done. Diagnosis.

PageSpeed



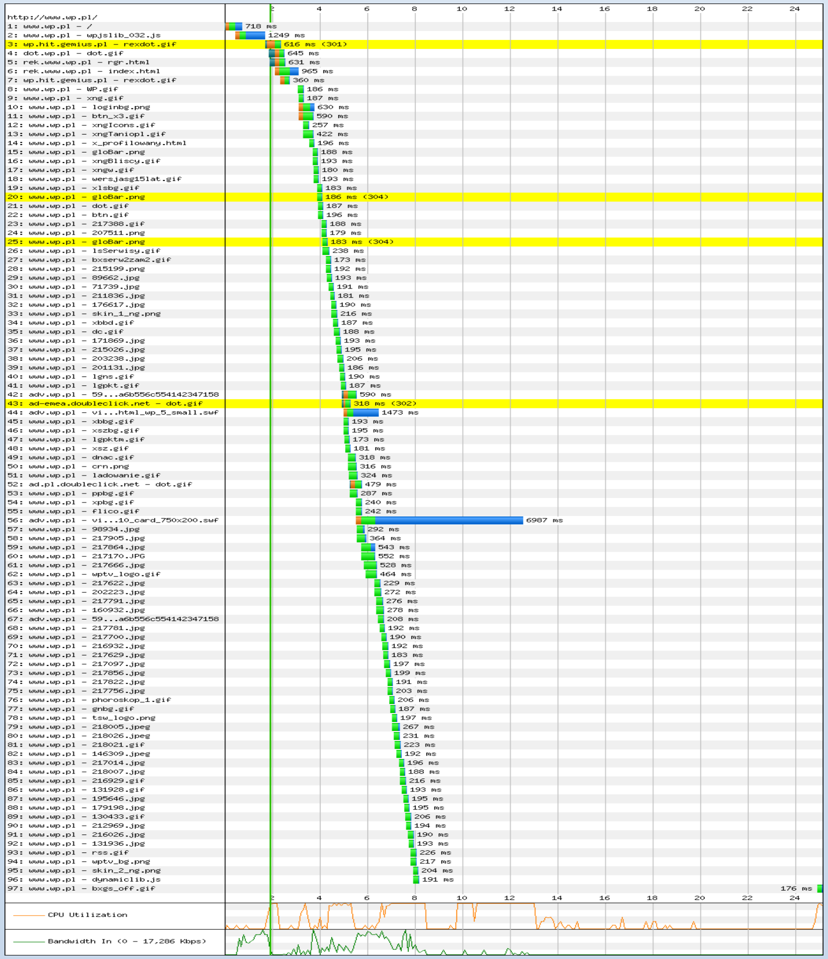
Page Speed Score: 84/100

- 🔴 Parallelize downloads across hostnames
- 🔴 Specify image dimensions
- ⚠ Enable compression
- ⚠ Leverage browser caching
- ⚠ Minimize DNS lookups
- ⚠ Optimize images
- ⚠ Serve resources from a consistent URL
- ⚠ Use efficient CSS selectors
- ✓ Combine external JavaScript
- ✓ Minify HTML
- ✓ Minify JavaScript
- ✓ Minimize redirects
- ✓ Specify a Vary: Accept-Encoding header

Meet waterfall

http://www.wp.pl

First View (Error: Timed Out)

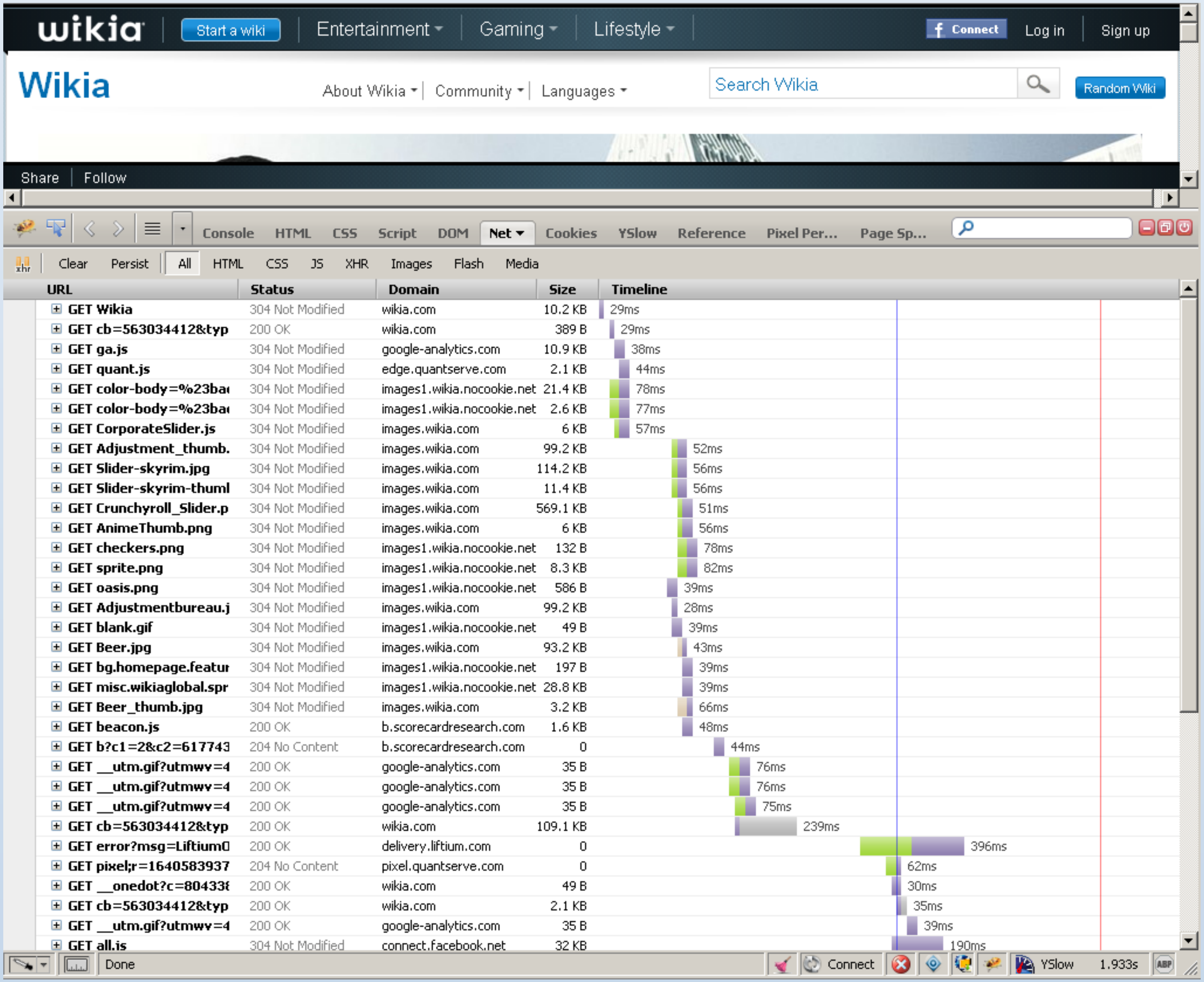


```
http://www.wp.pl/
1: www.wp.pl - /                                  718 ms
2: www.wp.pl - wpjslib_032.js                     1249 ms
3: wp.hit.gemius.pl - rexdot.gif                  616 ms (301)
4: dot.wp.pl - dot.gif                            645 ms
5: rek.www.wp.pl - rgr.html                       631 ms
6: rek.www.wp.pl - index.html                     965 ms
7: wp.hit.gemius.pl - rexdot.gif                  360 ms
8: www.wp.pl - WP.gif                             186 ms
9: www.wp.pl - xng.gif                            187 ms
10: www.wp.pl - loginbg.png                       630 ms
11: www.wp.pl - btn_x3.gif                        590 ms
12: www.wp.pl - xngIcons.gif                      257 ms
13: www.wp.pl - xngTaniopl.gif                    422 ms
14: www.wp.pl - x_profilowany.html                196 ms
15: www.wp.pl - gloBar.png                        188 ms
16: www.wp.pl - xngBliscy.gif                     193 ms
17: www.wp.pl - xngw.gif                          180 ms
18: www.wp.pl - wersjasg15lat.gif                 193 ms
19: www.wp.pl - xlsbg.gif                         183 ms
20: www.wp.pl - gloBar.png                        186 ms (304)
21: www.wp.pl - dot.gif                           187 ms
22: www.wp.pl - btn.gif                           196 ms
23: www.wp.pl - 217388.gif                        188 ms
24: www.wp.pl - 207511.png                        179 ms
25: www.wp.pl - gloBar.png                        183 ms (304)
26: www.wp.pl - lsSerwisy.gif                     238 ms
27: www.wp.pl - bxserw2zam2.gif                   173 ms
28: www.wp.pl - 215199.png                        192 ms
29: www.wp.pl - 89662.jpg                         193 ms
30: www.wp.pl - 71739.jpg                         191 ms
31: www.wp.pl - 211836.jpg                        181 ms
32: www.wp.pl - 176617.jpg                        190 ms
33: www.wp.pl - skin_1_ng.png                     216 ms
34: www.wp.pl - xbbd.gif                          187 ms
35: www.wp.pl - dc.gif                            188 ms
36: www.wp.pl - 171869.jpg                        193 ms
37: www.wp.pl - 215026.jpg                        195 ms
38: www.wp.pl - 203238.jpg                        206 ms
39: www.wp.pl - 201131.jpg                        186 ms
40: www.wp.pl - lgns.gif                          190 ms
41: www.wp.pl - lgpkt.gif                         187 ms
42: adv.wp.pl - 59...a6b556c554142347158          590 ms
43: ad-emea.doubleclick.net - dot.gif             318 ms (302)
44: adv.wp.pl - vi...html_wp_5_small.swf          1473 ms
45: www.wp.pl - xbbg.gif                          193 ms
46: www.wp.pl - xszbg.gif                         195 ms
47: www.wp.pl - lgpktm.gif                        173 ms
48: www.wp.pl - xsz.gif                           181 ms
49: www.wp.pl - dnac.gif                          318 ms
50: www.wp.pl - crn.png                           316 ms
51: www.wp.pl - ladowanie.gif                     324 ms
52: ad.pl.doubleclick.net - dot.gif               479 ms
53: www.wp.pl - ppbg.gif                          287 ms
54: www.wp.pl - xpbg.gif                          240 ms
55: www.wp.pl - flico.gif                         242 ms
56: adv.wp.pl - vi...10_card_750x200.swf          6987 ms
57: www.wp.pl - 98934.jpg                         292 ms
58: www.wp.pl - 217905.jpg                        364 ms
59: www.wp.pl - 217864.jpg                        543 ms
60: www.wp.pl - 217170.JPG                        552 ms
61: www.wp.pl - 217666.jpg                        528 ms
62: www.wp.pl - wptv_logo.gif                     464 ms
63: www.wp.pl - 217622.jpg                        229 ms
64: www.wp.pl - 202223.jpg                        272 ms
65: www.wp.pl - 217791.jpg                        276 ms
66: www.wp.pl - 160932.jpg                        278 ms
67: adv.wp.pl - 59...a6b556c554142347158          208 ms
68: www.wp.pl - 217781.jpg                        192 ms
69: www.wp.pl - 217700.jpg                        190 ms
70: www.wp.pl - 216932.jpg                        192 ms
71: www.wp.pl - 217629.jpg                        183 ms
72: www.wp.pl - 217097.jpg                        197 ms
73: www.wp.pl - 217856.jpg                        199 ms
74: www.wp.pl - 217822.jpg                        191 ms
75: www.wp.pl - 217756.jpg                        203 ms
76: www.wp.pl - phoroskop_1.gif                   206 ms
77: www.wp.pl - gnbg.gif                          187 ms
78: www.wp.pl - tsw_logo.png                      197 ms
79: www.wp.pl - 218005.jpeg                       267 ms
80: www.wp.pl - 218026.jpeg                       231 ms
81: www.wp.pl - 218021.gif                        223 ms
82: www.wp.pl - 146309.jpeg                       192 ms
83: www.wp.pl - 217014.jpg                        196 ms
84: www.wp.pl - 218007.jpg                        188 ms
85: www.wp.pl - 216929.gif                        216 ms
86: www.wp.pl - 131928.gif                        193 ms
87: www.wp.pl - 195646.jpg                        195 ms
88: www.wp.pl - 179198.jpg                        195 ms
89: www.wp.pl - 130433.gif                        206 ms
90: www.wp.pl - 212969.jpg                        194 ms
91: www.wp.pl - 216026.jpg                        190 ms
92: www.wp.pl - 131936.jpg                        193 ms
93: www.wp.pl - rss.gif                           226 ms
94: www.wp.pl - wptv_bg.png                       217 ms
95: www.wp.pl - skin_2_ng.png                     204 ms
96: www.wp.pl - dynamiclib.js                     191 ms
97: www.wp.pl - bxgs_off.gif                      176 ms
```

http://www.webpagetest.org/result/101117_CTH4/1/details/

http://www.flickr.com/photos/thesussman/4030969285

Meet waterfall



URL	Status	Domain	Size	Timeline
GET Wikia	304 Not Modified	wikia.com	10.2 KB	29ms
GET cb=563034412&typ	200 OK	wikia.com	389 B	29ms
GET ga.js	304 Not Modified	google-analytics.com	10.9 KB	38ms
GET quant.js	304 Not Modified	edge.quantserve.com	2.1 KB	44ms
GET color-body=%23ba	304 Not Modified	images1.wikia.nocookie.net	21.4 KB	78ms
GET color-body=%23ba	304 Not Modified	images1.wikia.nocookie.net	2.6 KB	77ms
GET CorporateSlider.js	304 Not Modified	images.wikia.com	6 KB	57ms
GET Adjustment_thumb.	304 Not Modified	images.wikia.com	99.2 KB	52ms
GET Slider-skyrim.jpg	304 Not Modified	images.wikia.com	114.2 KB	56ms
GET Slider-skyrim-thuml	304 Not Modified	images.wikia.com	11.4 KB	56ms
GET Crunchyroll_Slider.p	304 Not Modified	images.wikia.com	569.1 KB	51ms
GET AnimeThumb.png	304 Not Modified	images.wikia.com	6 KB	56ms
GET checkers.png	304 Not Modified	images1.wikia.nocookie.net	132 B	78ms
GET sprite.png	304 Not Modified	images1.wikia.nocookie.net	8.3 KB	82ms
GET oasis.png	304 Not Modified	images1.wikia.nocookie.net	586 B	39ms
GET Adjustmentbureau.j	304 Not Modified	images.wikia.com	99.2 KB	28ms
GET blank.gif	304 Not Modified	images1.wikia.nocookie.net	49 B	39ms
GET Beer.jpg	304 Not Modified	images.wikia.com	93.2 KB	43ms
GET bg.homepage.featu	304 Not Modified	images1.wikia.nocookie.net	197 B	39ms
GET misc.wikiaglobal.spr	304 Not Modified	images1.wikia.nocookie.net	28.8 KB	39ms
GET Beer_thumb.jpg	304 Not Modified	images.wikia.com	3.2 KB	66ms
GET beacon.js	200 OK	b.scorecardresearch.com	1.6 KB	48ms
GET b?c1=2&c2=617743	204 No Content	b.scorecardresearch.com	0	44ms
GET __utm.gif?utmwv=4	200 OK	google-analytics.com	35 B	76ms
GET __utm.gif?utmwv=4	200 OK	google-analytics.com	35 B	76ms
GET __utm.gif?utmwv=4	200 OK	google-analytics.com	35 B	75ms
GET cb=563034412&typ	200 OK	wikia.com	109.1 KB	239ms
GET error?msg=Liftium0	200 OK	delivery.liftium.com	0	396ms
GET pixel;r=1640583937	204 No Content	pixel.quantserve.com	0	62ms
GET __onedot?c=80433	200 OK	wikia.com	49 B	30ms
GET cb=563034412&typ	200 OK	wikia.com	2.1 KB	35ms
GET __utm.gif?utmwv=4	200 OK	google-analytics.com	35 B	39ms
GET all.js	304 Not Modified	connect.facebook.net	32 KB	190ms

Browser connection limits

Simultaneous number of connections per domain varies with browser:

FF2, IE6, IE7:

2 connections

IE8, FF3, Safari,Chrome:

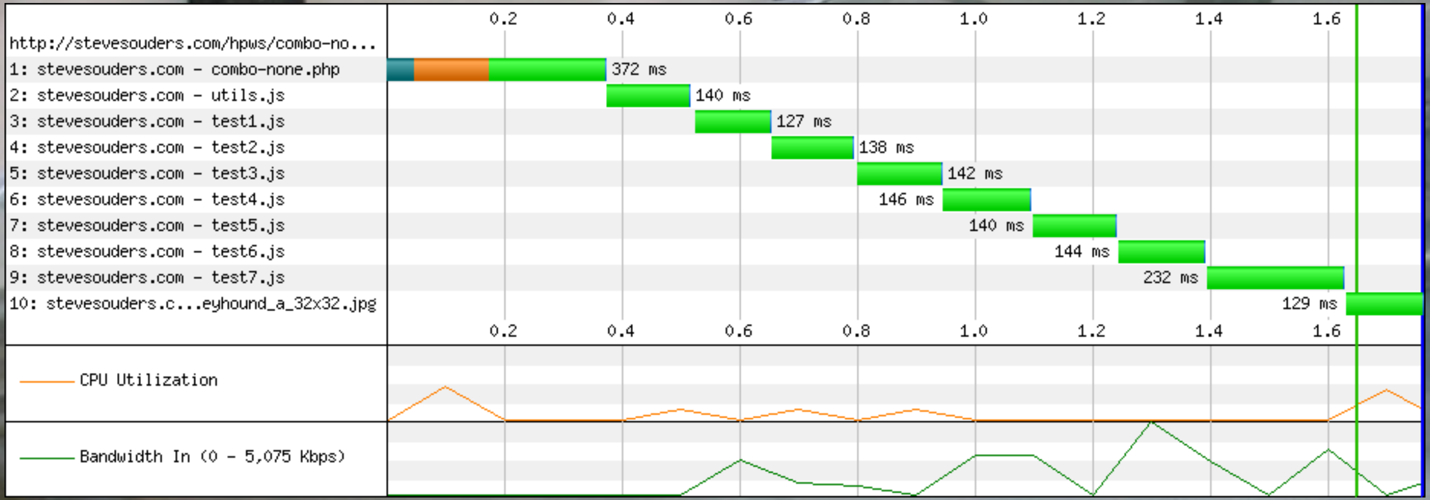
6 connections

But not all of them are used if JS is downloaded

http://stevesouders.com/hpws/js-blocking.php

http://www.flickr.com/photos/thesussman/4030969285

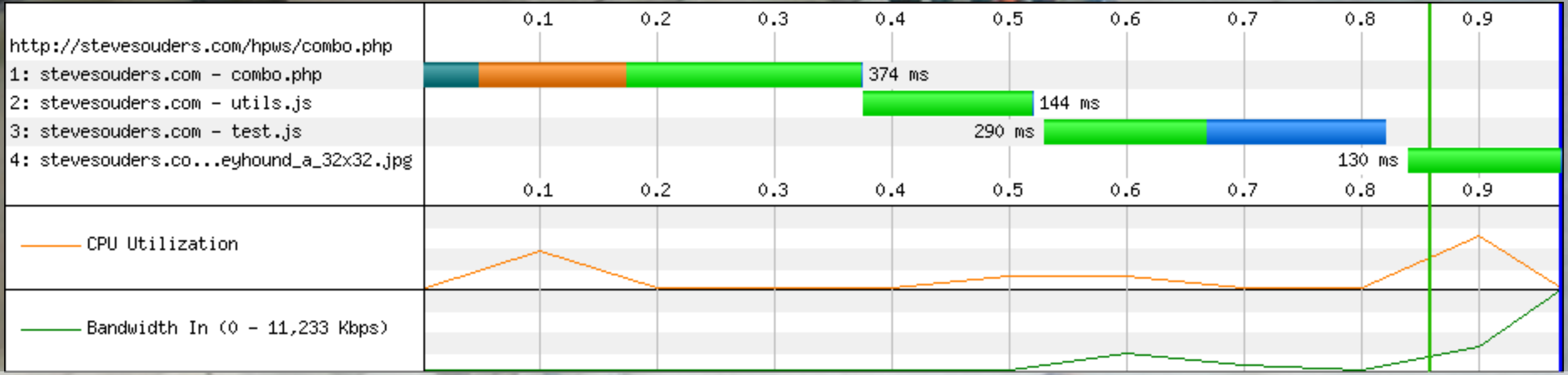
Reduce number of requests



Combine JS and CSS files

Use sprites (http://www.google.com/images/srpr/nav_logo37.png)

Avoid redirects.



http://www.flickr.com/photos/andrewpaulcarr/288909779/

Size matters.

Minify JS/CSS

Minify HTML

Gzip JS/CSS/HTML

http://www.flickr.com/photos/mia3mom/2704693800

Size matters.

Minify JS

- Google Closure Compiler
  http://code.google.com/closure/compiler/
- YUI Compressor
  http://developer.yahoo.com/yui/compressor/

Optimize CSS

- YUI Compressor
- Dust Me Selectors FF plugin
  http://www.sitepoint.com/dustmeselectors/
- CSSCompressor
  http://www.csscompressor.com/

mod_pagespeed

- http://code.google.com/speed/page-speed/docs/module.html

Size matters.

Minify HTML

HTMLCompressor for Java, mod_pagespeed

view-source:http://www.google.com/

Gzip JS/CSS/HTML

```
GET / HTTP/1.1
Accept-Encoding: gzip,deflate
```

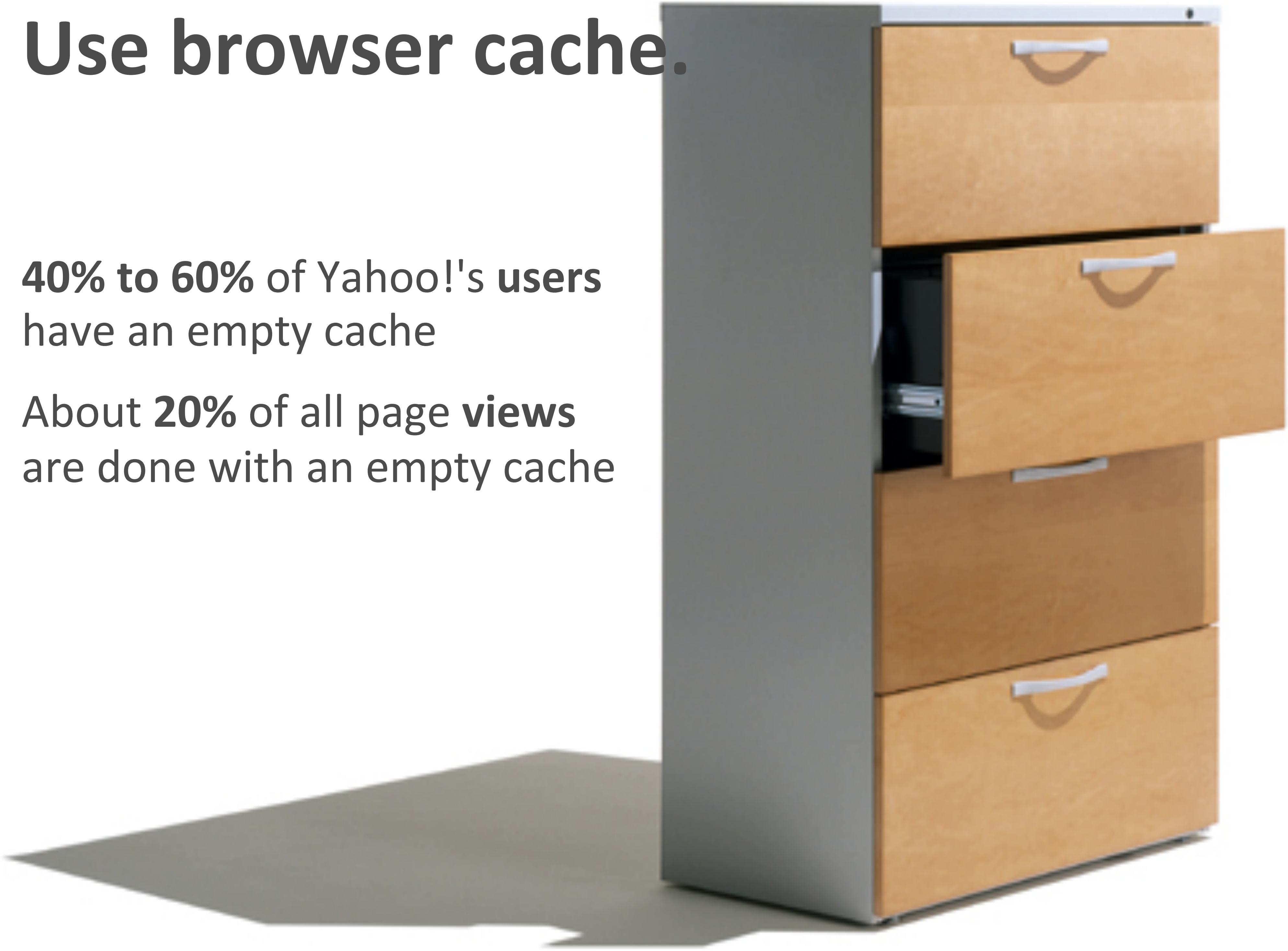
```
HTTP/1.1 200 OK
Date: Thu, 04 Dec 2003 16:15:12 GMT
Server: Apache/2.0
Vary: Accept-Encoding
Content-Encoding: gzip
Cache-Control: max-age=300
Expires: Thu, 04 Dec 2003 16:20:12 GMT
Content-Length: 1533
Content-Type: text/html; charset=ISO-8859-1
```

Home page	HTML (plain)	HTML (compressed)	savings
Google.com	3,873	1,412	63.5%
Google search	26,321	5,505	79.1%
Orbitz.com	44,183	9,046	79.5%

Typical size savings range from 60-85%

http://www.websiteoptimization.com/speed/tweak/compress/

Use browser cache.

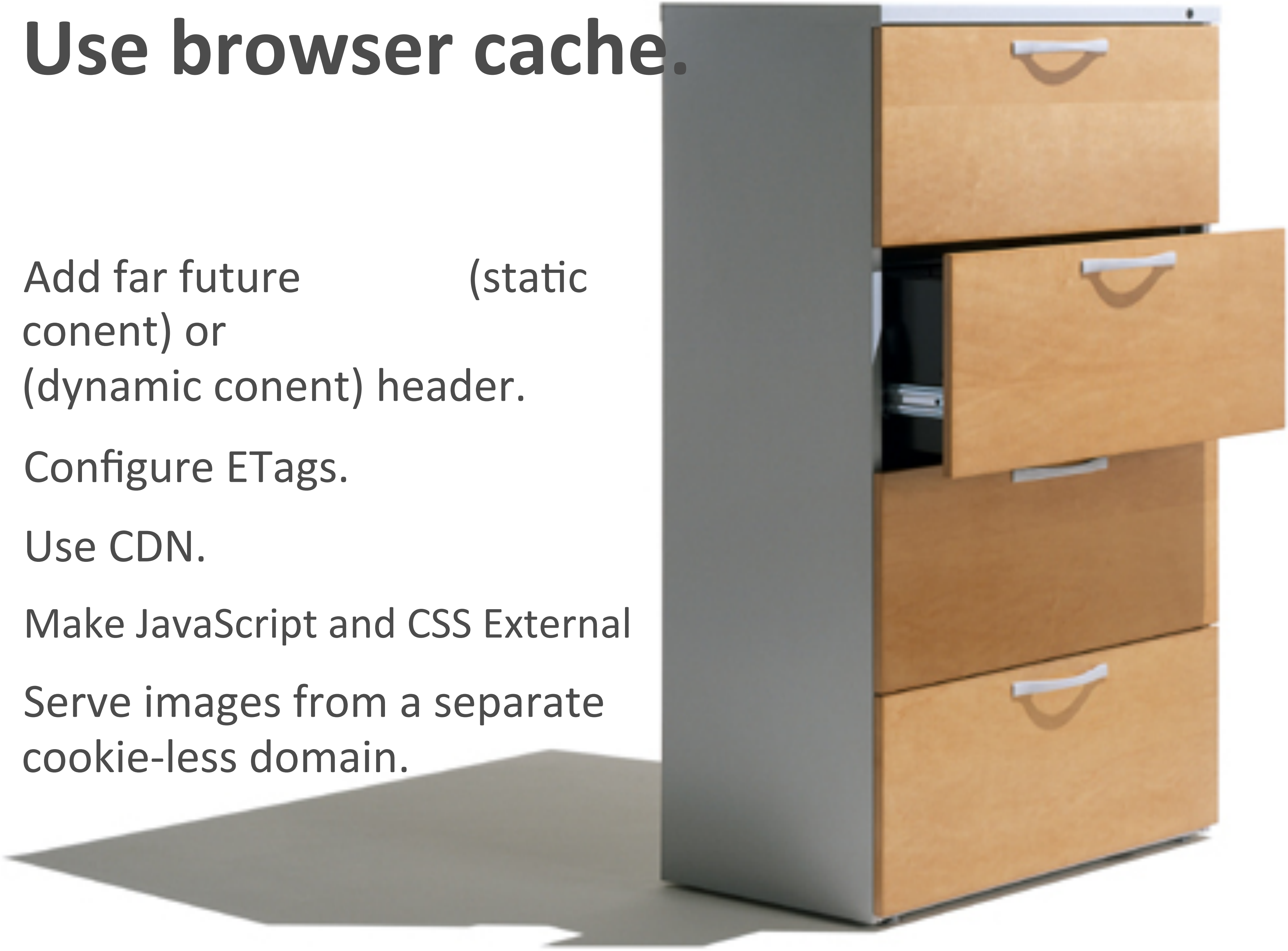


40% to 60% of Yahoo!'s **users** have an empty cache

About **20%** of all page **views** are done with an empty cache

http://www.flickr.com/photos/bfionline/2380398365/

Use browser cache.



Add far future (static conent) or (dynamic conent) header.

Configure ETags.

Use CDN.

Make JavaScript and CSS External

Serve images from a separate cookie-less domain.

http://www.flickr.com/photos/bfionline/2380398365/

Etags (Entity Tags)

```
GET /i/yahoo.gif HTTP/1.1


HTTP/1.1 200 OK
Last-Modified: Tue, 12 Dec 2006 03:03:59 GMT
ETag: "10c24bc-4ab-457e1c1f"
Content-Length: 12195
```

```
GET /i/yahoo.gif HTTP/1.1
If-Modified-Since: Tue, 12 Dec 2006 03:03:59 GMT
If-None-Match: "10c24bc-4ab-457e1c1f"


HTTP/1.1 304 Not Modified
```

Entity tags can reduce performance if configured incorrectly

Browser connection limits

Simultaneous number of connections **per domain** varies by browser:

FF2, IE6, IE7:

2 connections

FF3, Safari,Chrome:

6 connections

http://www.flickr.com/photos/thesussman/4030969285

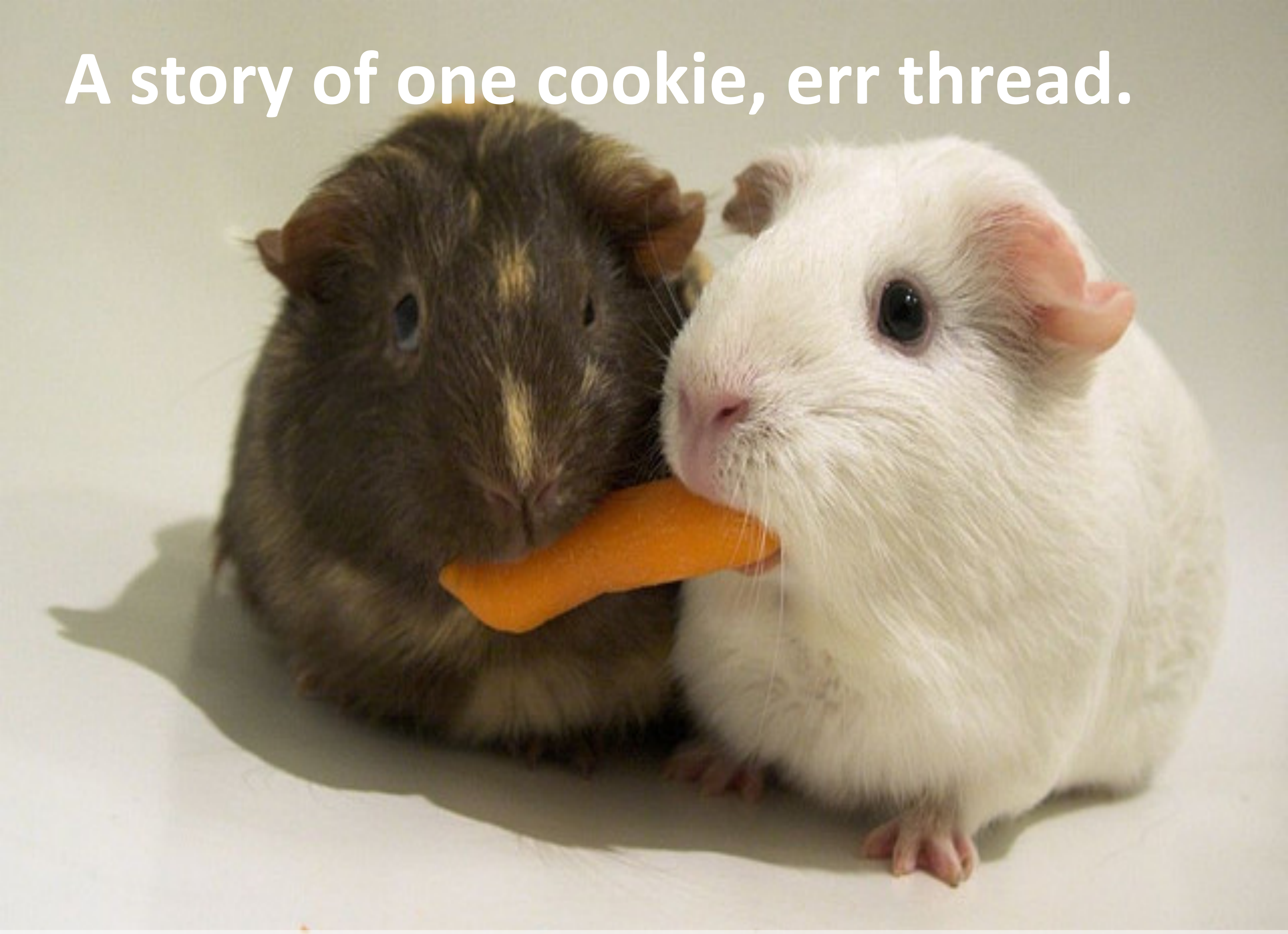
Split resources across domains

- Split resources across domains
  - 2 domains is enough to improve parallelization
  - Images (i.domain.com) – no cokies, safer (domainimg.com)
  - js + css (www.domain.com)

- Downgrade connections to HTTP 1.0

	HTTP 1.1	HTTP 1.0
FF2, IE6, IE7	2	4
IE8,FF3	6	6
Chrome	6	6

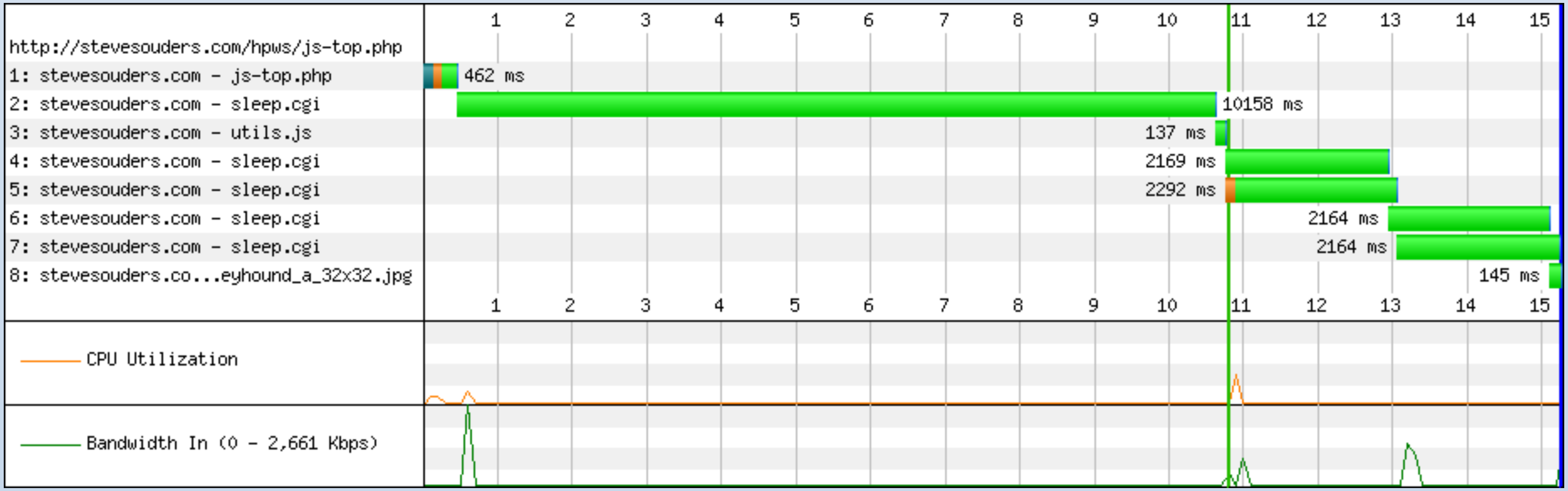
A story of one cookie, err thread.



http://www.flickr.com/photos/ryanr/142455033

JS blocks rendering

```
<html>
    <head>
        <script src="/bin/sleep.cgi?&sleep=10"></script>
        <title>Scripts at the Top</title>
```



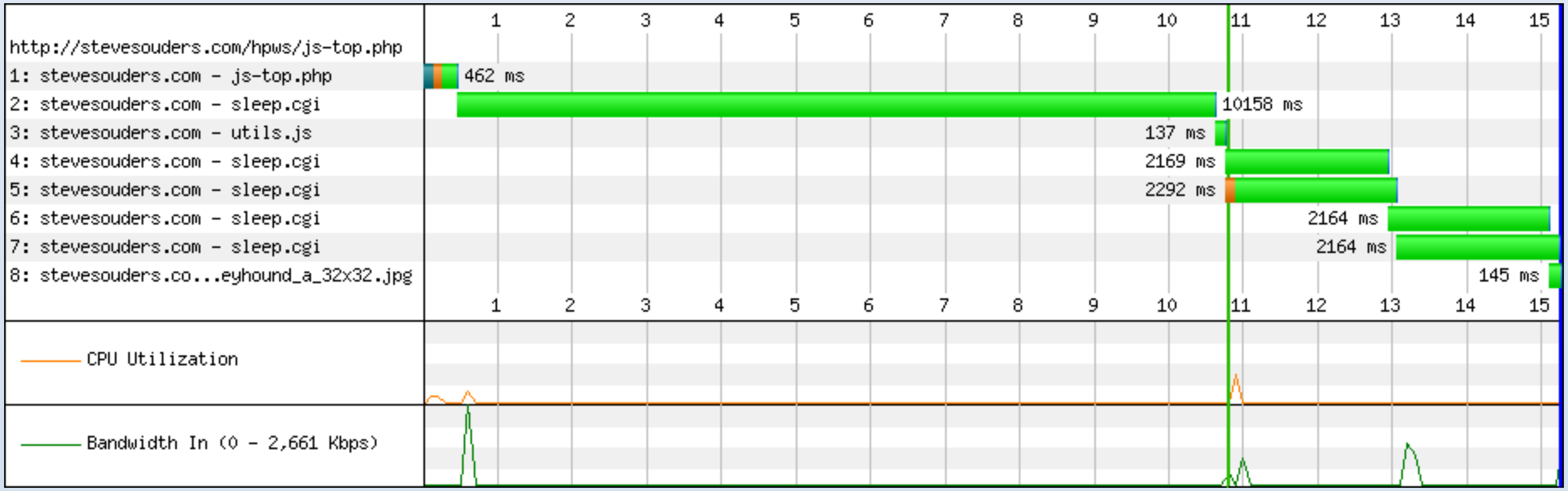
http://stevesouders.com/hpws/js-top.php?t1290026279466

Render first. JS second.

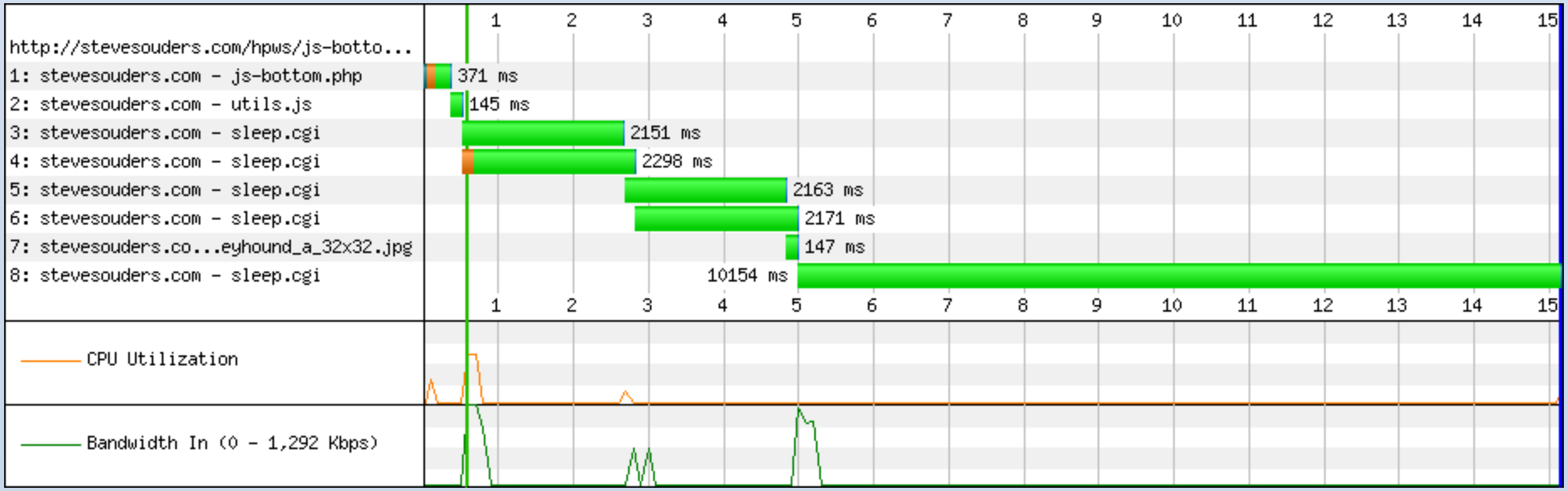
```
<html>
    <head>
        <title>Scripts at the bottom</title>
        <link rel="stylesheet" href="/bin/sleep.cgi">
    </head>
     <body>
     ...
        <script src="/sleep.cgi?&sleep=10"></script>
    </body>
</html>
```

Render first. JS second.

JS at the Top



JS at the Bottom



http://stevesouders.com/hpws/move-scripts.php

Load scripts asynchronously

XHR eval

XHR injection

Script in iframe

Script DOM element

document.write

defer attribute

Split initial content (load what necessary, rest later)

XHR Eval

```
var xhrObj = getXHRObject();
xhrObj.onreadystatechange =
  function() {
    if ( xhrObj.readyState != 4 ) return;
    eval(xhrObj.responseText);
  };
xhrObj.open('GET', 'A.js', true);
xhrObj.send('');
```

Same domain constraint

Douglas Crockford does no *eval*

http://stevesouders.com/efws/links.php?ex#Chapter4

XHR Injection

```
var xhrObj = getXHRObject();
xhrObj.onreadystatechange =
  function() {
    if ( xhrObj.readyState != 4 ) return;
    var se=document.createElement('script');
    document.getElementsByTagName('head')
        [0].appendChild(se);
    se.text = xhrObj.responseText;
  };
xhrObj.open('GET', 'A.js', true);
xhrObj.send('');
```

Same domain constraint
Faster than eval

Script in Iframe

```
<iframe src='A.html' width=0 height=0 frameborder=0 id=frame1></iframe>
```

iframe must have same domain as main page

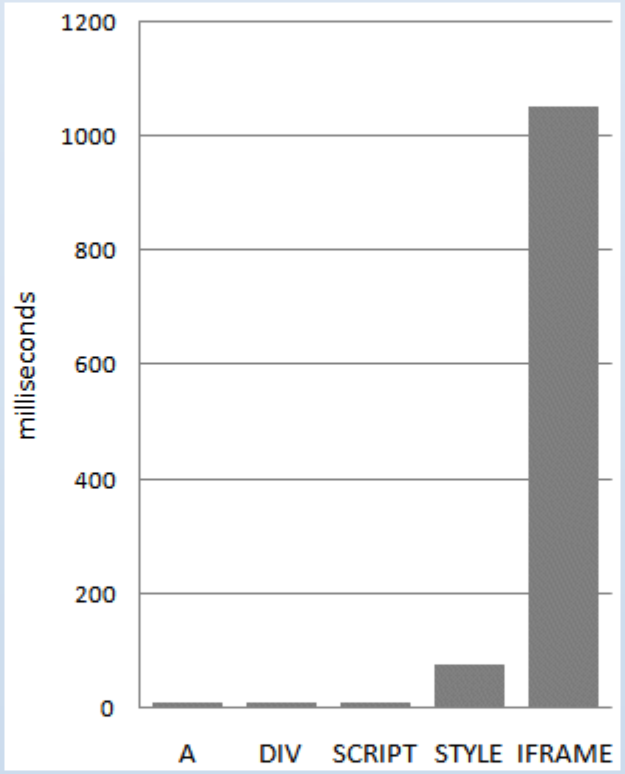
must refactor script to access script in iframe and access parent document from script

```
// access iframe from main page
window.frames[0].someMethodInScript();


// access main page from iframe
parent.document.createElement('div');
```

Avoid iframes

Iframes are most expensive DOM elements



load 100 *empty* elements of each type

Iframes block window.load

Iframes share connections with parent document

DOM Script element

```
var script = document.createElement('script');

script.src = 'http://anydomain.com/A.js';

document.getElementsByTagName('head’)[0].appendChild(se);
```

Different domains allowed

No refactoring needed

Sounds like a good choice? Yeah, but IE does not ensure order

document.write Script Tag

```
document.write("<scri" + "ipt type='text/javascript' src='A.js'>"
+ "</scri" + "ipt>");
```

Parallel downloads only for scripts and only in IE

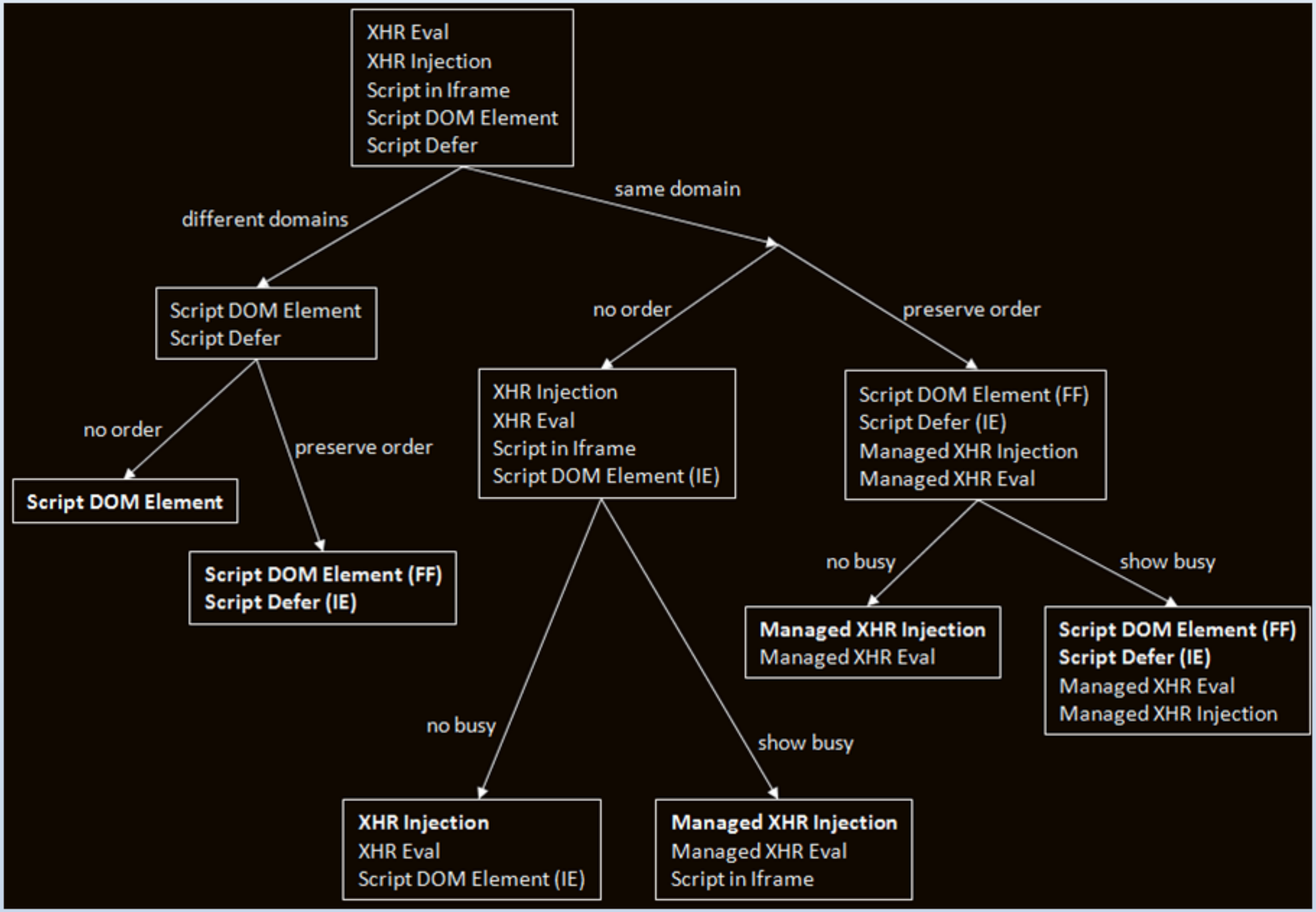
Script *defer*

```
<script defer src='A.js'></script>
```

Only IE

Cannot be used for scripts with *document.write* (ads...)

Which technique



Load scripts asynchronously

Various techniques:

	|| downloads	ensures order	domains can differ	existing scripts	browser busy	block render	block onload
normal Script Src	no	IE,FF	yes	yes	IE,FF	IE,FF	IE,FF
XHR Eval	IE,FF	no	no	no	no	no	no
XHR Injection	IE,FF	no	no	yes	no	no	no
Script in Iframe	IE,FF	no	no	no	IE,FF	no	IE,FF
Script DOM Element	IE,FF	FF	yes	yes	FF	no	FF
Script Defer	IE	IE	yes	yes	IE,FF	FF	IE,FF
document.write Script Tag	IE*	IE	yes	yes	IE,FF	IE,FF	IE,FF
	*Only other document.write scripts are downloaded in parallel (in the same script block).						

No single technique warranties order execution in all browsers

Use script **loaders**

Script loaders

Many loaders (LabJS, RequireJS, ControlJS, head.js, yepnope, YUI)

https://spreadsheets.google.com/lv?key=0Aqln2akPWiMIdERkY3J2OXdOUVJDTkNSQ2ZsV3hoWVE

Typical code

```
<script src="framework.js"></script>
<script src="plugin.framework.js"></script>
<script src="myplugin.framework.js"></script>
<script>
    myplugin.init();
    framework.init();
    framework.doSomething();
</script>
```

LABjs example

```
<script>
    $LAB.script("framework.js").wait()
        .script("plugin.framework.js")
        .script("myplugin.framework.js")
        .wait(function(){
            myplugin.init();
            framework.init();
            framework.doSomething();
        });
</script>
```

Take aways

Javascript blocks downloads

Reduce number of HTTP requests

Load JS asynchronously

TO BE CONTINUED

Javascript performance

Reduce time spent in JS

- One thread JS and UI
- Avoid actions longer than **100ms**

Split/delegate work:

- Web workers (FF3.5, Safari 4, Chrome 7 , Opera 10.6. No IE)
- Yelding (setTimeout)

Worker example

Main program code

```
var worker = new Worker("worker.js");
// Watch for messages from the worker
worker.onmessage = function(e){
  // The message from the client:
  e.data
};
worker.postMessage("start");
```

worker.js

```
onmessage = function(e){
  if ( e.data === "start" ) {
    // Do some computation
    done()
  }
};
function done(){
  // Send back the results to the parent page
  postMessage("done");
}
```

Scroll event handlers

requestAnimationFrame

CSS transitions

Yelding example

Split long running task into chunks.

```
function chunk(array, process, context) {
    setTimeout(function() {
        var item = array.shift();
        process.call(context, item);

        if (array.length > 0) {
            setTimeout(arguments.callee, 100);
        }
    }, 100);
}
```

Example use

```
function printUppercase(item) {
 console.log(item.toUppercase());
}

var items[] = „wpadla”, „gruszka”, „do”, fartuszka”;

chunk(items, printUppercase, window);
```

Javascript performance

Minimize operations on DOM

Use local variables scope

Avoid with/catch

If else > switch 1, 2

If else if else if < switch 1, 2, 3

Array[no] > switch > if

Optimize loops Var len = arr.len;

Reversed loop for(i--)

String concatenation < array.join

Don’t touch DOM... Too often

reflow – time to apply CSS, re-layout elements, and repaint

```
elem.className = "newclass";
elem.style.cssText = "color: red";
elem.style.padding = "8px";
elem.style.display = "";
```

reflow can happen multiple times for long-lasting Web apps

Don’t touch DOM… too often

```
var $ul = $("<ul/
>").appendTo("body");
for (var i = 0; i < 100; i++) {
    $("<li/>").appendTo($ul);
}
```

29ms (FF3.6) – plain page

159ms (FF3.6) – jquery.com

268ms (IE8) – jquery.com

```
var $ul = $("<ul/>")
for (var i = 0; i < 100; i++) {
    $("<li/>").appendTo($ul);
}
$ul.appendTo("body");
```

9ms (FF3.6) – plain page

88ms (FF3.6) – jquery.com

262ms (IE8) – jquery.com

Don’t touch DOM... too often

iframes are the most expensive DOM element to create
http://stevesouders.com/efws/costofelements.php

watch out for event handlers

```
$sitemap.find("a")
    .click(expandChildren);
```

```
$sitemap
    .delegate("a", "click",
    expandChildren);
```

Image optimization

Compression
- GIF and PNG favour horizontal patterns (eg. repeating colors)
- JPEG can look good in 70%
- Progressive JPEG (baseline < 10kB, progressive > 10kb)
  - Yahoo test on 10 000 images

Image format choice
- Graphics (logos, graphs, cartoons, icons) - GIF, PNG8
- Photos (JPEG – lossy , PNG24, PNG32 – non lossy)
- JPEG usually compresses better for true color photos but introduces artifacts around sharp color transitions

Pallette reduction
- Full pallete vs indexed color

Image transparency done wroong

AlphaImageLoader

```
.shadow {
    background-image: url(shadow.png);
    _background-image: none;
    _filter:progid:DXImageTransform.Microsoft.AlphaImageLoader(
        src='corner.png',
        sizingMethod='scale'
    );
}
```

freezes browser (IE waits for images to load before rendering page)

increased memory usage

invoked for every instance of the image on page (in UI thread)

Yahoo: 8ms per call * number of processed images

Image transparency done right

Transparency
- GIF < PNG8
- PNG8 -> alpha -> full transparency IE6

Photoshop can’t save proper alpha PNG8 for IE6
- *pngquant*

  ```
  pngquant 256 image.png
  ```

  http://www.libpng.org/pub/png/apps/pngquant.html
- *pngng (claims better results)*

  ```
  pngnq -n 256 image.png
  ```

  http://pngnq.sourceforge.net/
- Adobe Fireworks

Image optimization

Crushing PNGs

removing unnecessary chunks

```
pngcrush -rem alla -brute -reduce src.png dest.png
```

http://pmt.sourceforge.net/pngcrush/

remove all but alpha, try different optimizations, try to reduce pallette

(other: *pngout, OptiPNG, PngOptimizer)*

Stripping JPEG metadata

comments

application info (PS puts some garbage in there)

EXIF (geo, camera, exposure,date, **copyright**, etc.)

```
jpegtran -copy -none -optimize src.jpg > dest.jpg
```

http://jpegclub.org/

ExifTool – meta information management

http://www.sno.phy.queensu.ca/~phil/exiftool/

Image optimization

Smush.it

- Optimizes images using aformentioned tools
- Available from Yslow results panel
- http://www.smushit.com/ysmush.it/

Do not resize images in HTML

```
<img width="100" height="100" src="500x500.jpg" alt="my badly
      resized image" />
```

Provide favicons

- Avoid 404 pages
- cache *favicon.ico* or control location with *link rel=„shortcut icon” and use CDN*
- Limit ico to one size 16x16 (less than 1KB)

Provide Apple touch

- 57x57 *apple-touch –icon.png* in root

```
<link rel="apple-touch-icon" href="/customIcon.png"/>
```

Optimize CSS selectors

"The style system matches a rule by starting with the rightmost selector and moving to the left through the rule's selectors. As long as your little subtree continues to check out, the style system will continue moving to the left until it either matches the rule or bails out because of a mismatch.„

https://developer.mozilla.org/en/Writing_Efficient_CSS

```
#toc > LI { font-weight: bold; }
```

find every LI whose parent is id="toc"

```
#toc A { color: #444; }
```

find every A and climb its ancestors until id="toc" or DOM root (!) is found

```
P.class0007 SPAN { border: red }
```

find every span and climb its ancestors until finds matching P

Optimize CSS selectors

1. avoid universal selectors
   bad: `A.class0007 * { ... }`

2. don't qualify ID selectors
   bad: `DIV #navbar {}`
   good: `#navbar {}`

3. don't qualify class selectors
   bad: `LI .tight {}`
   good: `.li-tight {}`

4. make rules as specific as possible
   bad: `#navbar A {}`
   good: `.a-navbar {}`

5. avoid descendant selectors
   bad: `UL LI A {}`
   better: `UL > LI > A {}`

Optimize CSS selectors

6. avoid tag-child selectors
   bad: `UL > LI > A {}`
   best: `.li-anchor {}`

7. be wary of child selectors
   `DIV:first-child {}`

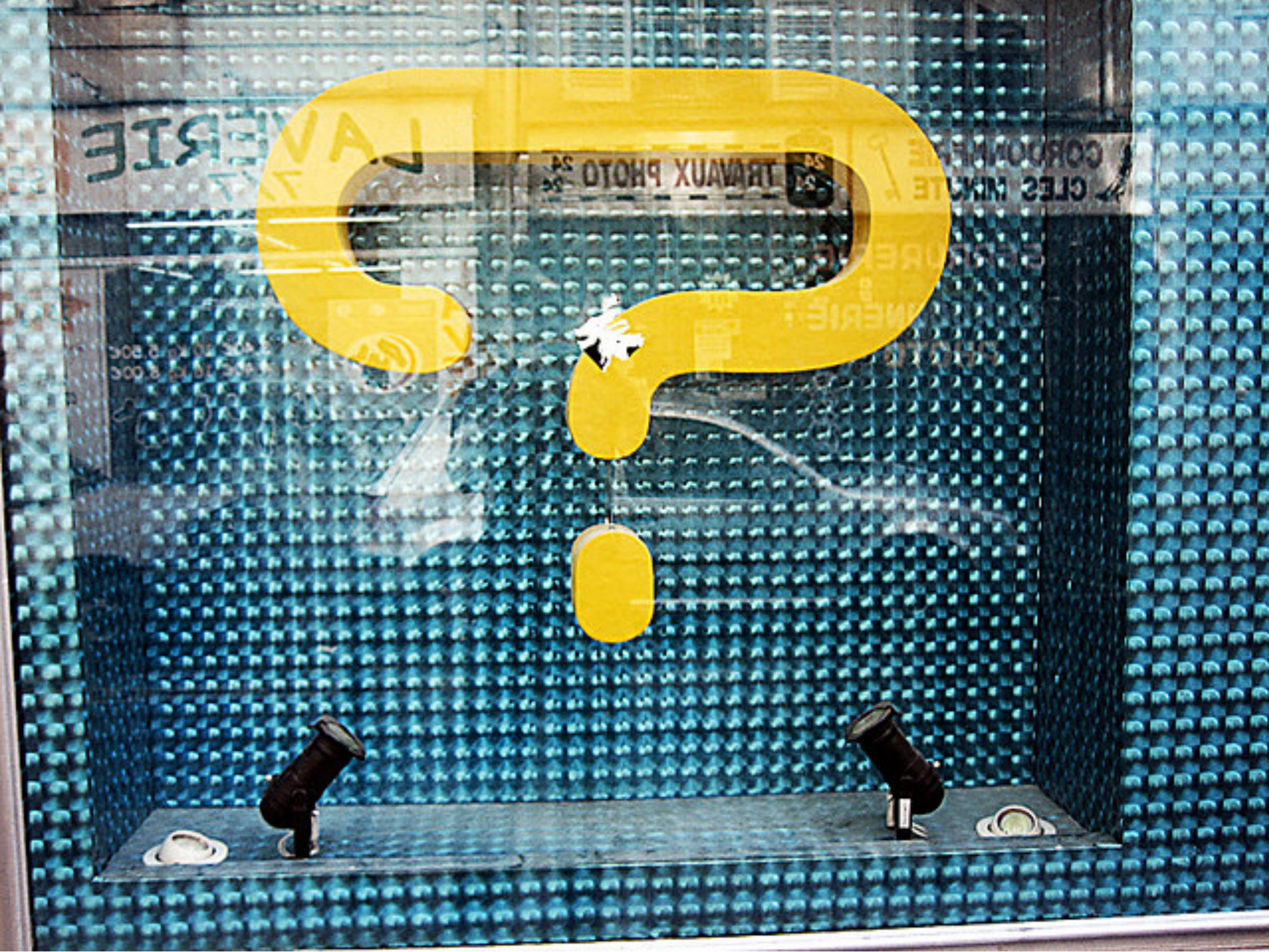
8. Avoid attribute selectors
   bad: `.class0007 [href]`

9. CSS3 selectors are really slow compared to .class/id selectors

10. rely on inheritance
    http://www.w3.org/TR/CSS21/propidx.html

Tools.

- Firebug Net Panel
- http://www.webpagetest.org
- Yslow (http://developer.yahoo.com/yslow/)
- Google Page Speed (http://code.google.com/speed/page-speed/)
- Fiddler
- mod_pagespeed
- DynaTrace Ajax edition



More reading

http://developer.yahoo.com/performance/rules.html

http://stevesouders.com/hpws/

http://stevesouders.com/efws/

http://code.google.com/speed/

http://www.websiteoptimization.com/speed/

Copyrights and attribution

Based on books and presentations of Steve Souders, a web preformance guru.