

# Speed Matters

Simple Ways to Make Your Web Site Faster



Flickr: Uploaded February 11, 2007 by hawridger

September 17 • DevGroup NW

Jason Grigsby • <http://userfirstweb.com>

# 2003: Business Critical Look at Speed

- Customer complaints
- Limits of our 3 T-1s near
- Months away from new data center



Flickr: Uploaded November 13, 2003 by Wools

September 19 • DevGroup NW

Jason Grigsby • <http://userfirstweb.com>

2

4 years ago we faced a business critical issue. The issue was two-fold.

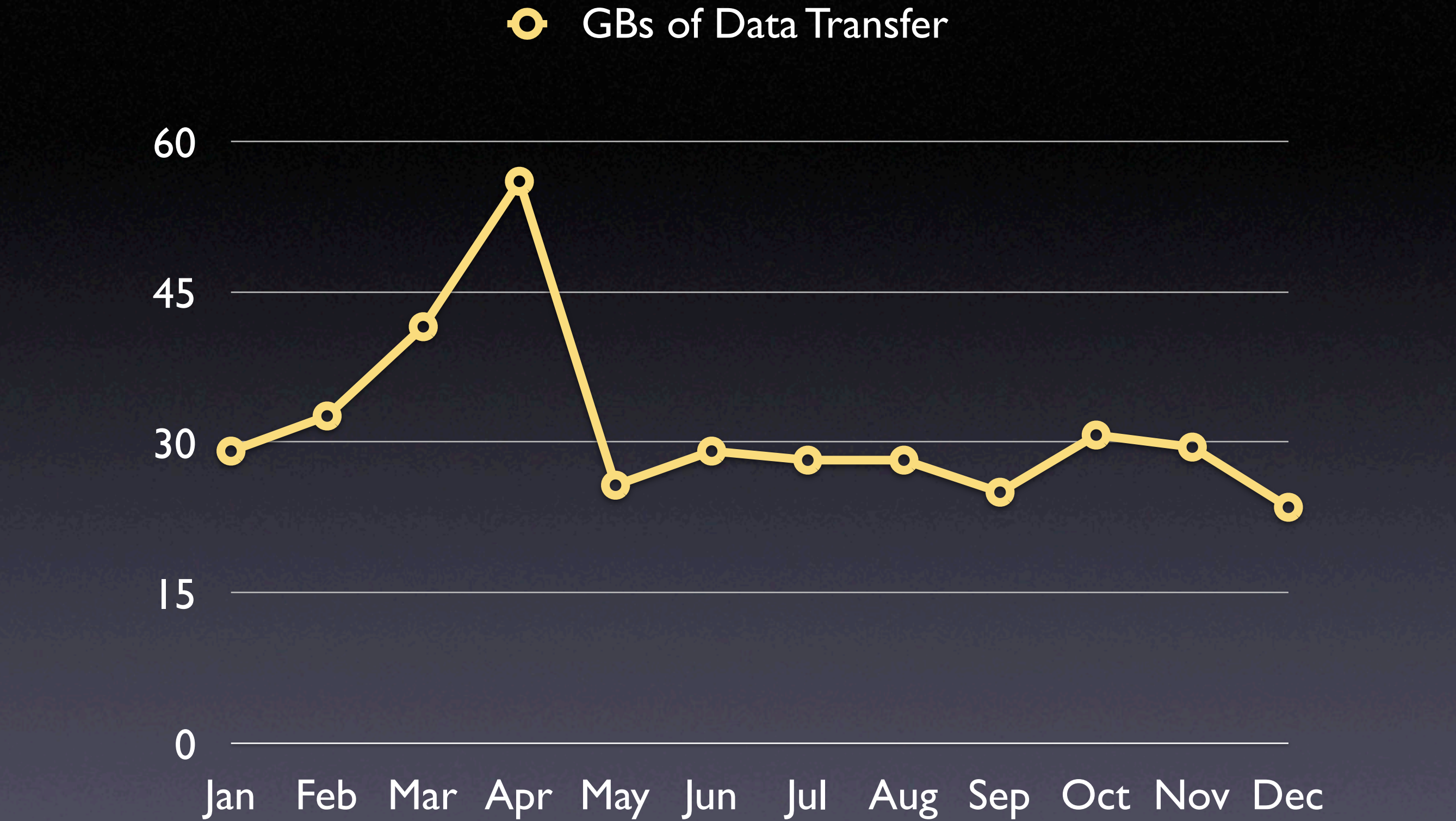
First, customers were complaining about the speed of their web sites. Because we both develop applications and host customer sites, these issues couldn't be ignored.

If the customer issues weren't enough, we were bumping up against what we called a backhoe problem. We had three T1s into our office where we were hosting the sites. This was the maximum we could do without an expensive project tearing up streets to lay new fiber.

But our bandwidth was starting to approach our cap and while we were looking for a new data center, we didn't have anything ready.

It wasn't looking good for the home team.





At this point, our CTO John Keith and I started focusing on efforts to speed up customer sites. We devoured the information in Speed Up Your Site by Andrew King.

We examined everything from the database queries to the way pages were constructed to look for ways we could increase the speed of the sites that we hosted.

You can probably guess from the graph of a single customer's data transfer, which month we implemented our changes. The bandwidth used for each customer site dropped substantially.



75% faster page loads

7 months in data center



Flickr: Uploaded August 2, 2006 by pbo31

September 19 • DevGroup NW

Jason Grigsby • <http://userfirstweb.com>

4

<http://www.flickr.com/photos/pbo31/205375212/>

The end result of our work was pages that loaded somewhere between 75 to 80% faster. Our customers noticed the difference immediately and the number of support calls we got decreased.

Perhaps more importantly, we were able to buy the time to investigate data centers, pick the right one for our business, and move to that data center in an organized fashion.

Finally, we learned that speeding up a site comes from a holistic perspective. First, you've got to believe that it matters and today I'll give you evidence of why speed matters. Second, once you understand its importance, you begin to understand the whole system of user experience. It is this systematic view—a systematics savings—that is the key to the success of your site.



# Speed Matters

- Why Speed Matters
- The Yahoo! 80/20 rule.
- Ten Techniques for Speed
- Case Studies
- Testing Your Work
- Homework Assignments

Flickr: Uploaded January 21, 2007 by themagnificentEM

September 19 • DevGroup NW

Jason Grigsby • <http://userfirstweb.com>

5

[http://www.flickr.com/photo\\_zoom.gne?id=364362984&size=o](http://www.flickr.com/photo_zoom.gne?id=364362984&size=o)

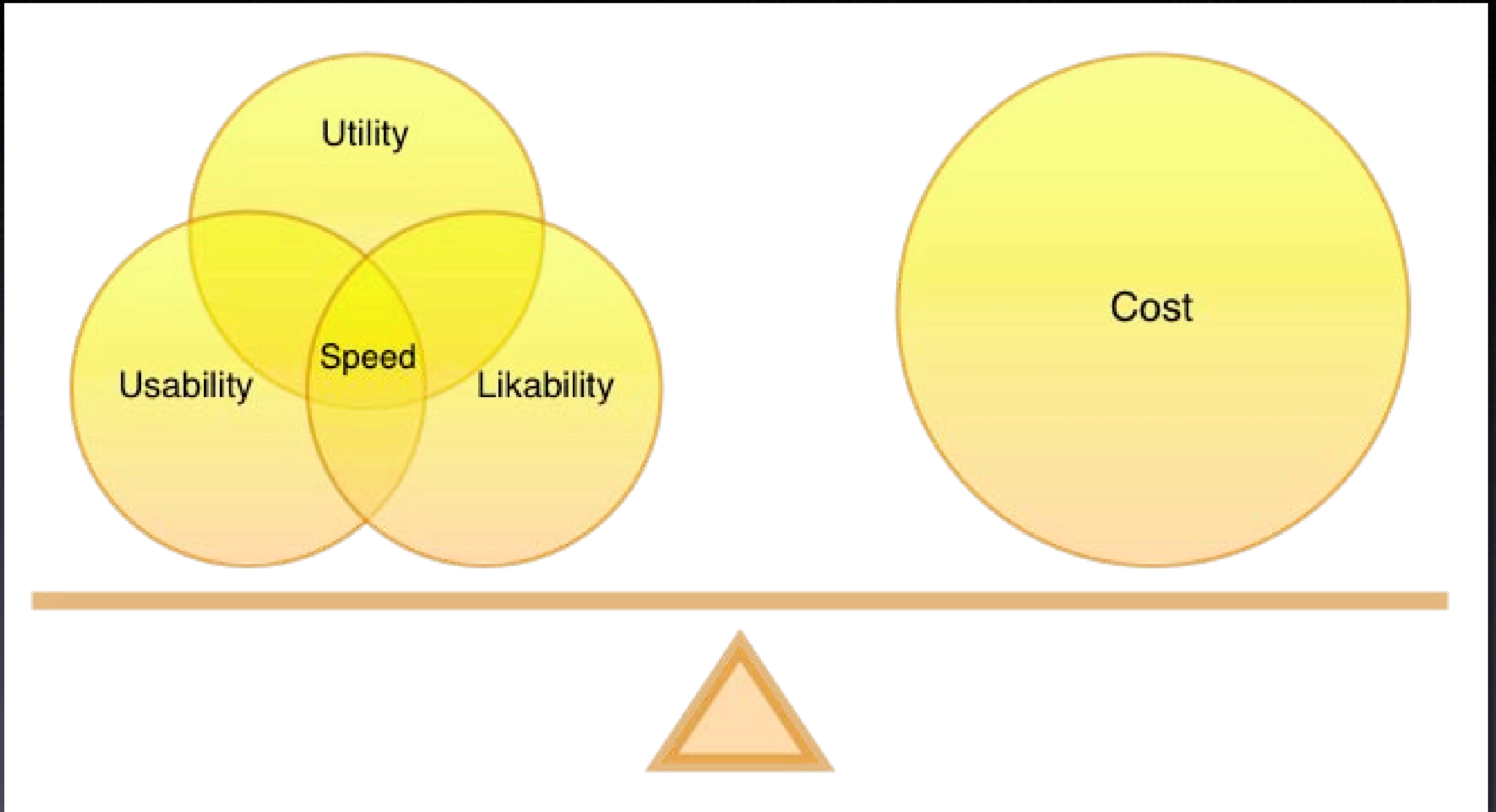
In today's presentation you're going to learn why speed matters, learn where the major improvements to web site performance come from (the so called 80/20 rule). I'll give you ten techniques for speeding up your site. We'll look at a couple of case studies and how you can measure your success.

But you're not getting out of here without homework. I'm going to give you action items to take back to your organization.

Finally, unless you work for the Yahoo Performance Team, I guarantee that you will learn something new today.



# Speed = Success



## Shackel's Acceptability Paradigm

Source: Speed Up Your Site by Andrew King, p. 6.

People are really good at evaluating whether or something will be worth their time. We do it all of the time without even thinking about.

In Human Computing Interface research, a lot of time has been devoted to understand how we evaluate these decisions. Brian Shackel put forward a paradigm that describes this evaluation process of determining system acceptability.

Utility - Perceived usefulness. Does it do what we need it to?

Usability - Perceived ease of use. Can we use it easily?

Likability - Subjective attitude about the system. Do we like the way it feels?

One important note is that all of these items are perception—not objective reality.

And at the center of Usability, Utility and Likability is speed. Study after study shows a high correlation between the speed of a system and someone's perception of the three factors.



# Web Speed = Perception Quality

## Usability

# BRANDED

## Credibility

## Service

Flickr: Uploaded October 15, 2006 by mattlogelin

September 19 • DevGroup NW

Jason Grigsby • <http://userfirstweb.com>

7

[http://www.flickr.com/photo\\_zoom.gne?id=270573110&size=l](http://www.flickr.com/photo_zoom.gne?id=270573110&size=l)

When people started testing Shackel's Paradigm on the web, they found a strong correlation.

The speed of a web site has been shown to impacts:

- Perception of the usability of a site.
- The perceived quality of the experience.
- The perceived quality of the service.
- The perceived credibility of the organization.

The researchers found that slow loading pages cause people to doubt the trustworthiness and expertise of the organization.

We tend to think of a company's brand in terms of logos, colors, and brand identity programs. What this research tells us is that your brand is tied to the speed of your site. If you don't pay attention to it, you can have the greatest brand identity system in the world, but to your users it will look like your brand is the equivalent of spray paint on the side of a dumpster.



# Optimizing for Flow



**0.1 seconds —**  
Viewed as  
Instantaneous

**1.0 seconds —**  
Moving Freely

**10 seconds —**  
Retain Focus on Task

Flickr: Uploaded August 7, 2007 by .Hessam

September 19 • DevGroup NW

Jason Grigsby • <http://userfirstweb.com>

8

[http://www.flickr.com/photo\\_zoom.gne?id=1039282425&size=l](http://www.flickr.com/photo_zoom.gne?id=1039282425&size=l)

Call it what you like: Web 2.0. AJAX. Social Media. Whatever.

The underlying facts remain the same. We are building more applications online than ever before.

And when people are using our web sites to get their work done, they are looking to get into a groove. Into a flow.

Every time someone has to switch contexts, they lose time and they forget what they were doing. Their efficiency decreases and the number of mistakes increase.

Therefore we have to make sure our pages are responsive enough to keep people in the flow of their work. We need them to stay in their zone.



- 
- **Ecommerce Bailout Rates**
  - **Bandwidth Charges**

Flickr: Uploaded November 7, 2005 by Tracy O

September 19 • DevGroup NW

Jason Grigsby • <http://userfirstweb.com>

9

[http://www.flickr.com/photo\\_zoom.gne?id=61056391&size=l](http://www.flickr.com/photo_zoom.gne?id=61056391&size=l)

And the final reason that Speed Matters is that Speed equals Money.

Akamai and JupiterResearch recently surveyed a thousand online shoppers and found that four seconds was the maximum length of time an average online shopper will wait before potentially abandoning a retail site. Web sites that weren't responsive diminished goodwill, created a negative brand perception, and a significant loss in overall sales.

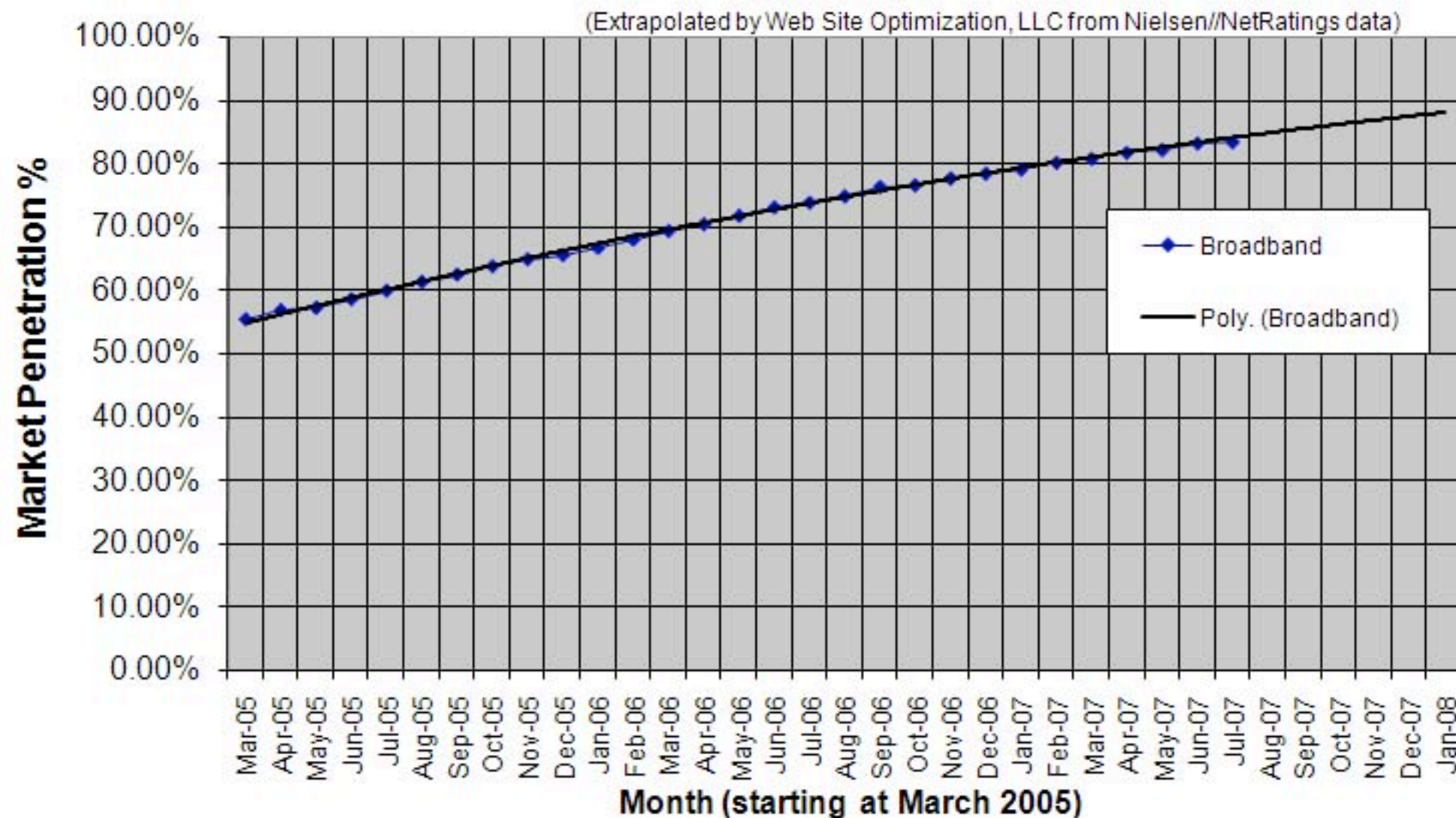
And if you are paying for your bandwidth, which you likely are at some level, then you may be paying more money to deliver that inferior user experience.

So with all of these compelling reasons to focus on the speed of pages, why don't more web developers focus on speed?



# Broadband Solves this Right?

**Broadband Growth Trend - US Home Users**



Source: <http://www.websiteoptimization.com/bw/0708>

For those of us who have been building web sites for awhile, we started out caring a lot about the speed of websites. We learned to optimize images, we tried to make them load as fast as possible. Why was this focus lost?

Basically, we all got faster internet connections so we stopped feeling the pain. We forgot that not everyone has broadband.

And even if they do have broadband, we've increased the size of pages so much with javascript, style sheets, and all of this technology that we didn't consider in the past, that we're often still delivering sites that do not perform as well as people expect.

Earlier this week, I searched YouTube looking for humorous videos that demonstrated how slow the web can be. I found over 900 videos about Internet speed. Most of these videos were angry rants about the speed of their connection.

I learned two things from these videos:

1. I learned is that a lot of people are so mad about their speed online to take the time to create a video.
2. I learned that in no way is it a laughing matter for these people.

Later, I'm going to show you why even if you have a broadband connection, that you are unlikely to ever download a web page at even a fraction of the speed of that connection. For the moment, it is just important to know that we've still got 20% of U.S. users and even more internationally that still do not have high speed connections. Do you want to write off those potential customers?



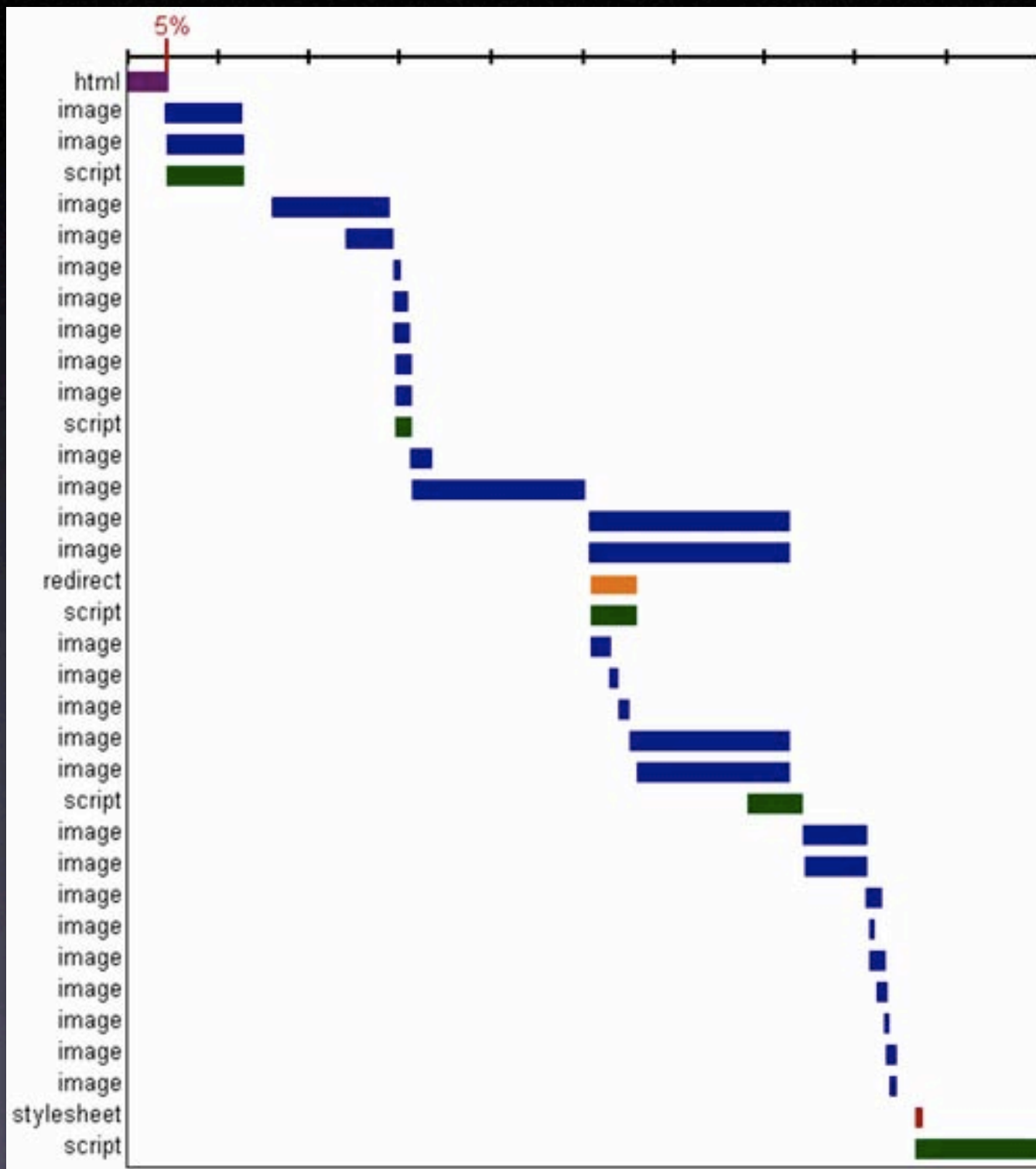


And then there are the game changers. The 2.7 billion mobile devices in the world. More than the number of computers and landing lines combined. More than the number of televisions.

Ask any iPhone owner how fast their connection is when they are using AT&T's Edge Network. They'll be happy to tell you how much they enjoy sites that load fast like Google and avoid sites that take too long to load.



# Yahoo's 80/20 Rule



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

September 19 • DevGroup NW

Jason Grigsby • <http://userfirstweb.com>

12

Now that you're convinced of the importance of speed, where do we focus our energies? In order to deliver a web page, you have databases, web servers, programming languages, disk drives, routers, firewalls, load balancing, internet connections, the size of the page, the browser rendering and even the cpu speed of the computer viewing the page. That's a lot of stuff to worry about.

Fortunately, Yahoo! created a performance team a few years ago and tasked this team with finding out how to speed the performance of Yahoo's many web sites. This performance team has published their findings. One of the most important findings is what they call the 80/20 rule.

Yahoo found that 80% of the time that users spent waiting for a page to load was spent after the html document was downloaded.

Why is this important? If the html document has been downloaded, then this usually means that all of the server side work has been done. What is left is the browser parsing the html and requesting the images, javascript, etc.



# Time Spent Loading Popular Sites

	Time Retrieving HTML	Time Elsewhere
Yahoo!	10%	90%
Google	25%	75%
MySpace	9%	91%
MSN	5%	95%
ebay	5%	95%
Amazon	38%	62%
YouTube	9%	91%
CNN	15%	85%

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

The team at Yahoo was stunned by this result. The team was full of engineers that had spent their careers optimizing databases and server systems to eek the best performance out of systems.

When they were assigned the task to find out how to speed up Yahoo’s sites, they expected to find the same results.

When presented with information so contrary to their expectations, they decide to look to see if Yahoo was unique. Do other popular sites show the same characteristics?

Yes, as you can see, the majority of the time to view a page is spent after the html is downloaded. This means that if you want to speed up a web site, you have to focus on the front-end. You have to focus on the design, html and other things that fall outside of the server.

Which is why I’m here talking to you about this. This isn’t a task for the system administrators. This is a task for web developers.



# 10 Techniques for Speed

1. Web Standards
2. HTML Optimization
3. Speed Up Tables
4. CSS Optimization
5. Javascript Optimization
6. Increase Caching
7. Reduce DNS Lookups
8. Avoid Redirects
9. Fewer HTTP Requests
10. GZIP

Flickr: Uploaded July 24, 2006 by julipan

September 19 • DevGroup NW

Jason Grigsby • <http://userfirstweb.com>

14

[http://www.flickr.com/photo\\_zoom.gne?id=197149338&size=l](http://www.flickr.com/photo_zoom.gne?id=197149338&size=l)

Fortunately, creating faster web pages is simple. Today I'm going to show you 10 techniques for speeding up your site.

I've saved the best for last. Efforts to create fewer HTTP requests and using GZIP can often speed up sites 70 to 80% alone and often require nothing more than changing a web server configuration.



# I. Web Standards

- CSS-based vs. table layouts
- Separate content, style, behavior
- Less markup overall
- Fewer http requests (no spacers)
- ESPN proved that this can be significant alone.

Flickr: Uploaded December 29, 2006 by bertheymans

September 19 • DevGroup NW

Jason Grigsby • <http://userfirstweb.com>

15

[http://www.flickr.com/photo\\_zoom.gne?id=337851288&size=l](http://www.flickr.com/photo_zoom.gne?id=337851288&size=l)

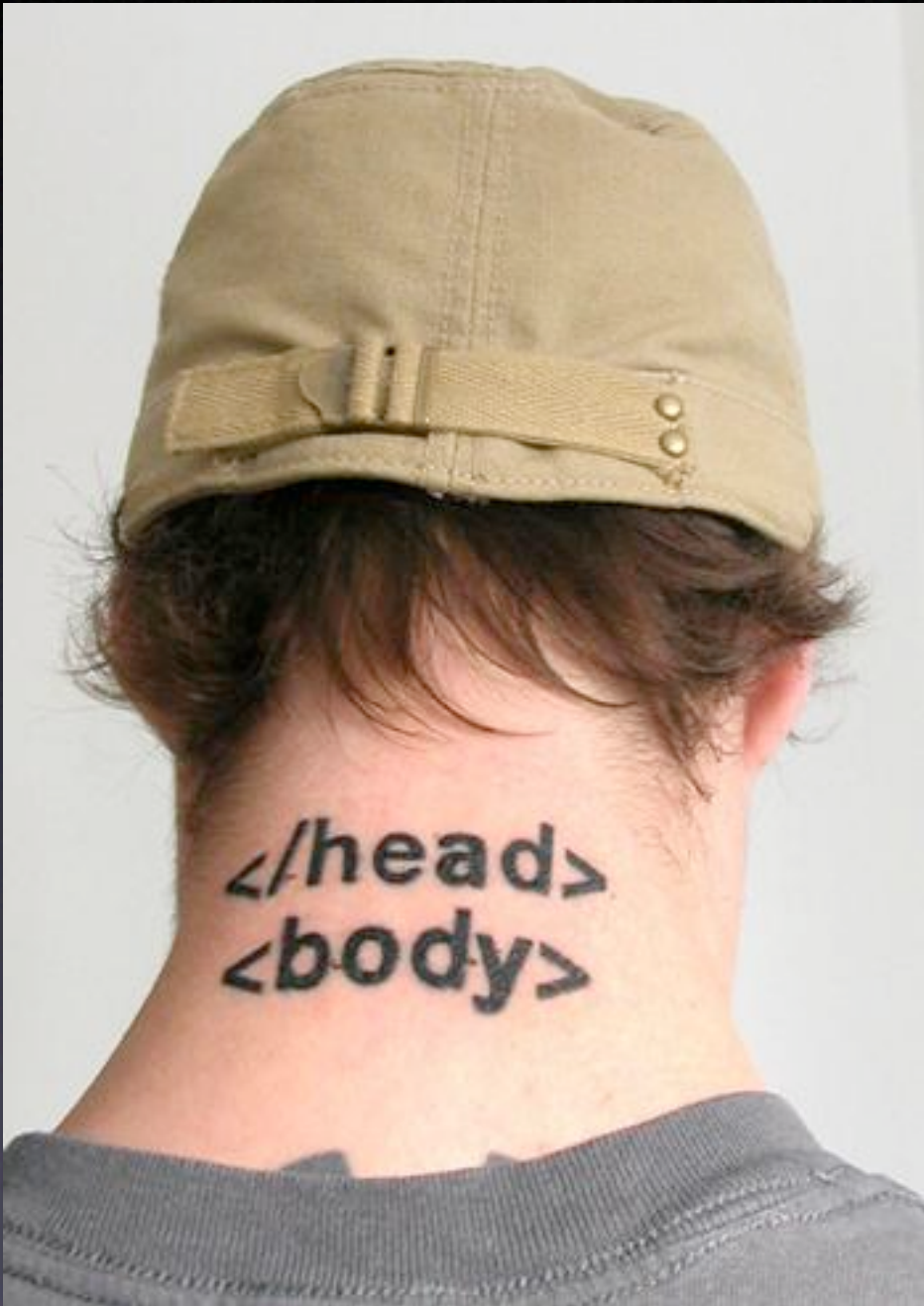
In 2003, ESPN redesigned to use web standards. A css-based layout.

They shaved 50k off their site. At the time, they receive 40 Million page views a day. They projected saving 2 terabytes of bandwidth a day, 61 terabytes a month, and 730 terabytes of data over a year.

Site speed isn't one of the main reasons usually given for designing with web standards, but it has a large impact.



# 2. HTML Optimization



- Remove Unnecessary White Space
- Replace HTML Comments with Server-side Comments
- Use Short URLs
- Minimize the size of the `<head>` tag.
- Reduce tables. Speed up those that remain.

Flickr: Uploaded March 15, 2007 by .dan

[http://www.flickr.com/photo\\_zoom.gne?id=422213306&size=o](http://www.flickr.com/photo_zoom.gne?id=422213306&size=o)

If you are following web standards best practices, you're already doing much of the HTML optimization. Now you can save further space by removing unnecessary white space, replacing html comments with server-side comments. etc.

Look for ways to automate the reduction of white space so your development environment has the code in a readable fashion and then compress it before shipping it.

Later we'll talk about GZIP which if you use it, the benefits of removing the white space is diminished and depending on the amount of traffic you get, may be unnecessary.

It is important to minimize the amount of information in the head tag because the head tag has to be completed before the rest of the page will render.

We'll spend a lot of time looking at ways to speed up rendering. This is because seeing the page loading is like a progress bar for users.



# 3. Speed Up Tables

- Enable Incremental Display
  - If you want the browser to format a table in one pass, you need to tell the browser the number of columns in the table and their widths.
- `<colgroup><col></col><col></col></colgroup>`
  - These Tags Declare Table Structure Allowing the Browser to Display the page in one pass
- Fixed Table Layout — `{ table-layout: fixed;}`
  - Microsoft Claims 100-Fold Improvement By Using this Code
  - Test before deploying

Flickr: Uploaded August 15, 2007 by bdesham

Under the "fixed" layout method, the entire table can be rendered once the first table row has been downloaded and analyzed. This can drastically speed up rendering time over the "automatic" layout method, but subsequent cell content may not fit in the column widths provided (the 'clip' and 'overflow' properties control the cell appearance in such a case.)



# 4. CSS Optimization

- Put stylesheets in the <head> tag only.
- Replace CSS Comments with Comments in Your Programming Language of Choice.
- Minify CSS (YUI Compressor)
- Let Stylesheets Cascade — Inherit Styles
- Use CSS Shorthand Properties (margin: 10px 5px 10px 5px; instead of margin-left, etc.)
- Optimize CSS Colors (#ffc instead of #ffffcc)

Flickr: Uploaded March 28, 2007 by mkw87



# 5. Javascript Optimization

- Put scripts at the bottom of the page.
- Minify Javascript (JSTMin or YUI Compressor)
- External files for caching purposes.
- Remove duplicate scripts
- Write efficient javascript

Flickr: Uploaded March 18, 2007 by Amodiovalerio Verde

[http://www.flickr.com/photo\\_zoom.gne?id=425333516&size=o](http://www.flickr.com/photo_zoom.gne?id=425333516&size=o)

Like stylesheets, the placement of scripts impacts the speed of the page. But the solution is the opposite. Put the scripts at the bottom of the page if possible.

Progressive page rendering is blocked until the javascript is processed. In addition, when javascript is downloading, some browsers won't start any other downloads until the javascript is downloaded.

Minifying the javascript reduces the white space and often makes scripts run faster.

The yahoo folks talk about removing duplicate scripts. Apparently, they found a lot of examples of larger teams having this issue.

Finally, write efficient javascript. With all of the emphasis on AJAX now, there is



# 6. Increase Caching

- All content has a last-modified date
- Add Expires header to static content
- Develop plan for versions of static files so you can break the cache when needed.
- Configure or remove ETags from files. Removing them may be preferable.
- Watch your logs to make sure things are getting cached as you expect.

Flickr: Uploaded July 9, 2007 by John Wardell (Netinho)



# 7. Reduce DNS Lookups

- Each DNS lookup costs performance and you have no control over the DNS speed.
- Nothing can be downloaded from the domain until the lookup completes.
- Optimum is between 2 and no more than 4 domains per web page because of concurrent download benefits.

Flickr: Uploaded March 3, 2007 by André-Batista



# 8. Avoid Redirects

- Redirects cost the user time so avoid the easy ones.
- Permanent 301 redirects should be modified to be cacheable (expires header)
- Put the trailing slash on your links and shave time off of Apache doing the work.

Flickr: Uploaded September 17, 2006 by (nz) dave



# 9. Fewer HTTP Requests

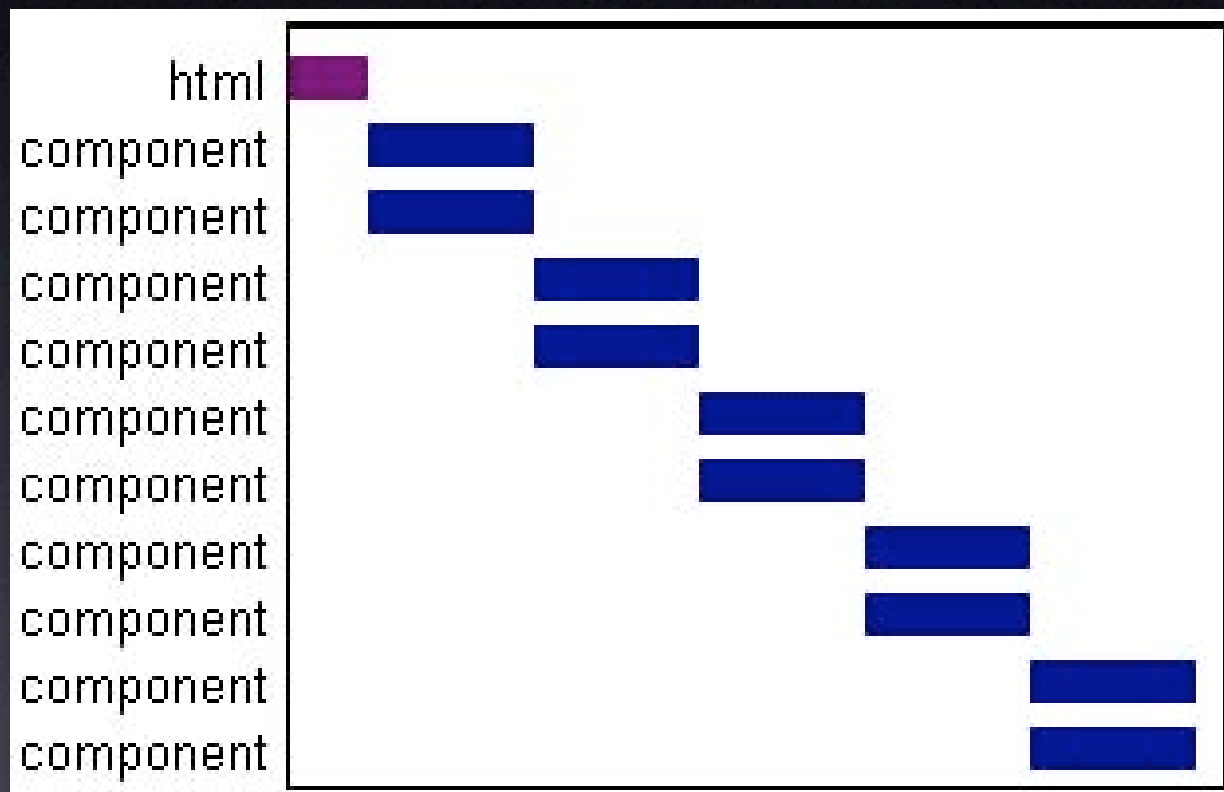
A large, bold, yellow-painted word "SLOW" on a dark, textured asphalt surface. The paint is slightly worn and the letters are thick.

**Most browsers only open 2 http connections to one domain at a time.**

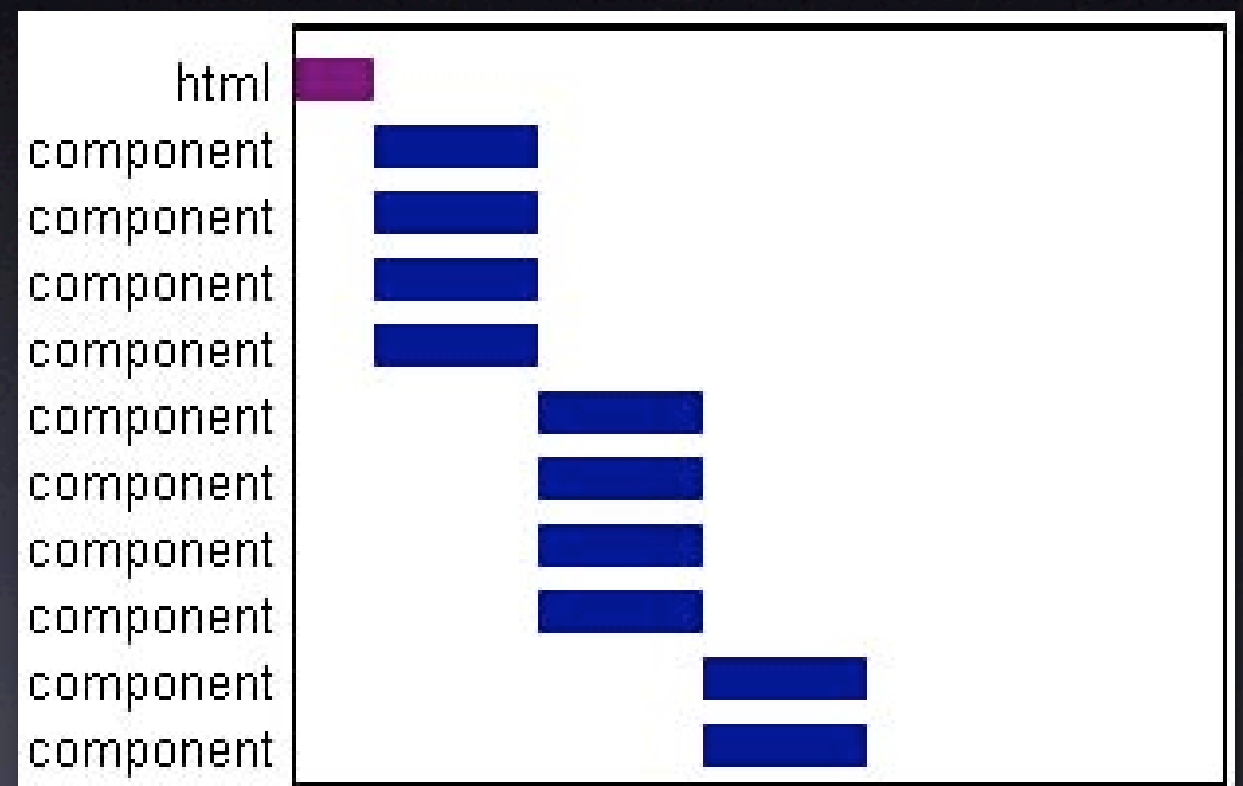
Flickr: Uploaded April 28, 2007 by ((IANB))



# Parallel HTTP Requests



One Domain



Two Domains

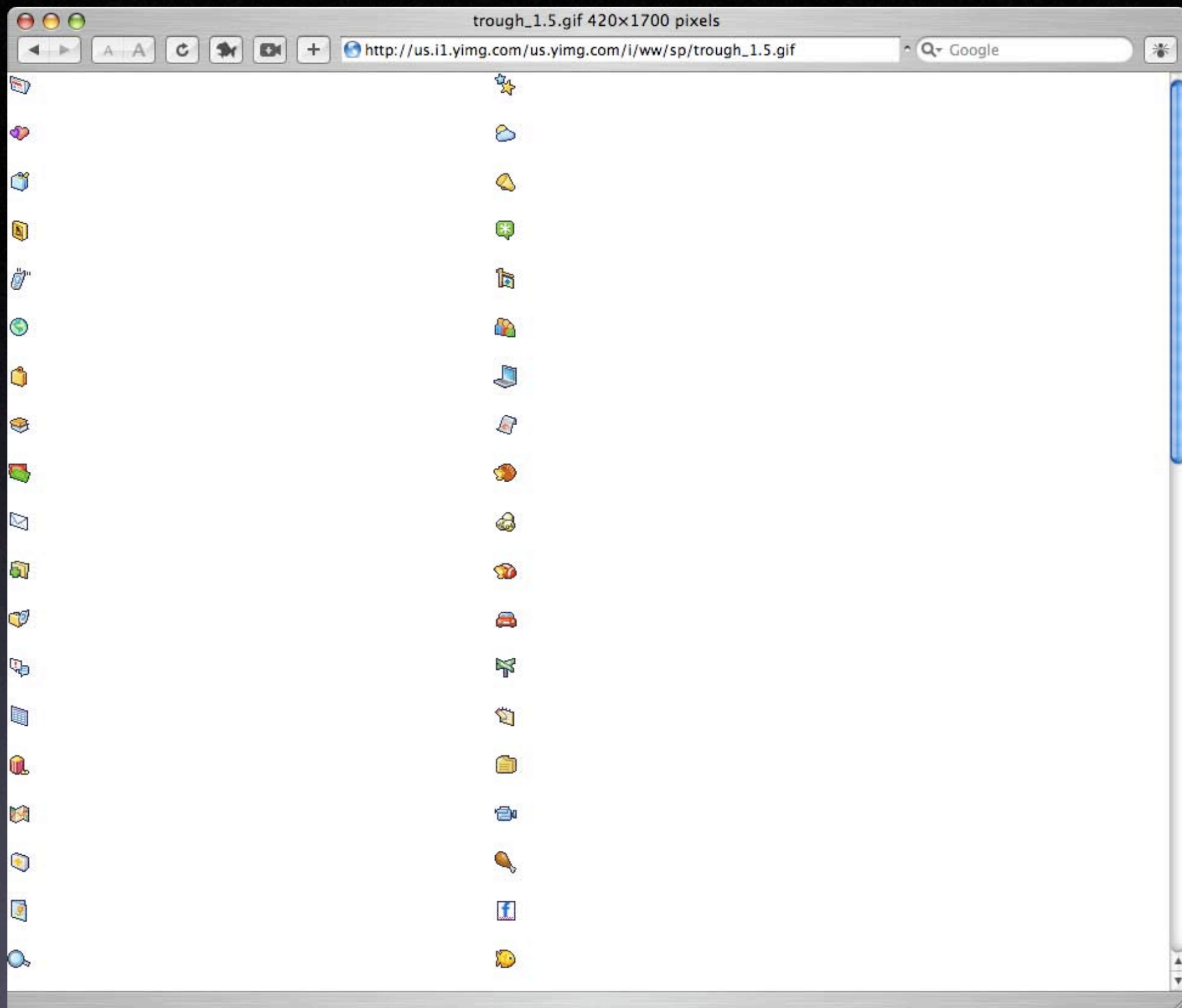
Source: <http://yuiblog.com/blog/2007/04/11/performance-research-part-4>



# Reducing HTTP Requests

- CSS Sprite techniques for images
- Combine files into one stylesheet file, one javascript file.
- Ensure that only essential items get downloaded and once they are downloaded that they are cached.





# 74 icons in single file.

[http://us.i1.yimg.com/us.yimg.com/i/ww/sp/trough\\_1.5.gif](http://us.i1.yimg.com/us.yimg.com/i/ww/sp/trough_1.5.gif)

What we now call CSS Sprites was originally a technique developed by a web developer called Petr Stanicek (aka Pixy) from the Czech Republic. Later, Douglas Bowman popularized the technique in A List Apart article and called it the Sliding Door technique.

Both Pizy and Bowman were trying to address issues where IE wouldn't preload an image in a css-based navigation system. Because it didn't preload the images, when you first hovered over the navigation it would flicker.

This technique for solving IE's flicker issues can be used to speed up sites. Instead of separate image files for things like icons, create a single image. The single image will be smaller in size than the size of each image in a separate optimized file.

When you take into consideration the price paid for HTTP Requests, you can see why Yahoo chose to put 74 icons in a single image that they reference throughout their design.



# 10. GZIP Compression

- If you do nothing else, DO THIS!
- GZIP can reduce files and download time up to 75%.
- Works on HTML, CSS, Javascript, XML and JSON files
- Most modern browsers handle GZIP correctly and the libraries like mod\_gzip handle exceptions correctly.

Flickr: Uploaded February 11, 2007 by hawridger



# Case Study

Version	HTML	CSS	JS	Total
Original	40,837 bytes	17,764 bytes	1,443 bytes	60,044 bytes
Optimized	24,907 bytes	5,320 bytes	1,443 bytes	31,670 bytes
GZIP	5,722 bytes	5,320 bytes	1,443 bytes*	12,485 bytes

**Nearly 80% reduction in file size.**



# Testing Your Work

- Web Page Speed Report

<http://www.websiteoptimization.com/services/analyze/>

Integrated with Web Developer Toolkit

- YSlow Firefox Plugin (requires Firebug)

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

- Live HTTP Headers Plugin

<http://livehttpheaders.mozdev.org>

Flickr: Uploaded February 11, 2007 by hawridger



# Your Homework

1. Find out if your company is using GZIP. If not, make it a priority.
2. Look for ways to consolidate your css and javascript files into a smaller number of files.
3. Work with your coworkers to establish standards for your organization.
4. GZIP! GZIP!



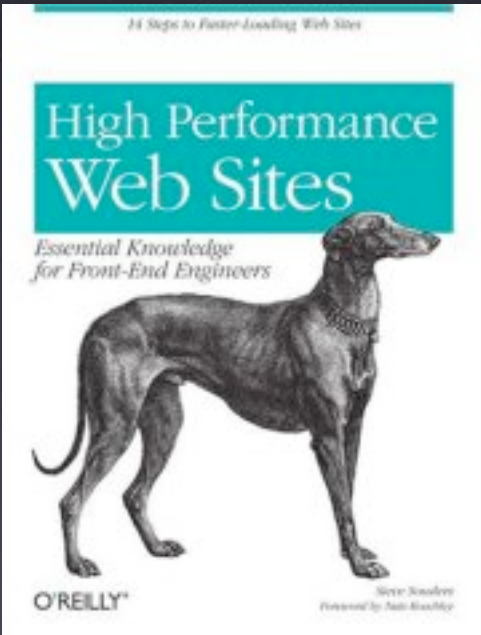
# Speed: Easy, Fun & Addictive

- Speed truly does matter & broadband isn't going to save you
- There isn't a silver bullet (but GZIP comes close)
- # of HTTP request plays a huge role
- There are simple things you can do today.
- Race car mechanic shaving bits to save seconds

Flickr: Uploaded May 26, 2007 by pbo31



# More Resources



- Speed Up Your Site by Andrew King  
<http://www.speedupyoursite.com>
- High Performance Web Site by Steve Souders
- Yahoo! Exceptional Performance Team  
<http://developer.yahoo.com/performance/>
- YSlow Plugin  
<http://developer.yahoo.com/yslow>
- YUI Compressor  
<http://developer.yahoo.com/yui/compressor/>
- Slides and additional materials available at:  
<http://userfirstweb.com/>