

Web-Performance

Walter Ebert

Drupal Meetup
Frankfurt/Rhein-Main
14. März 2013

Backend-Performance

Last-Tests

- Apache Bench
- Siege
- JMeter

Server-Output

Output Cache

- Boost-Modul
- Varnish

Bessere Hardware

- Schnellere CPU / Mehr Kerne
- Schnelleres / mehr RAM
- Schnellere Festplatten, z.B. SSD

Anwendung

PHP-Opcode-Cache

- APC
- XCache
- Zend Optimizer (standard in PHP5.5)

Datenbank

- Query-Cache

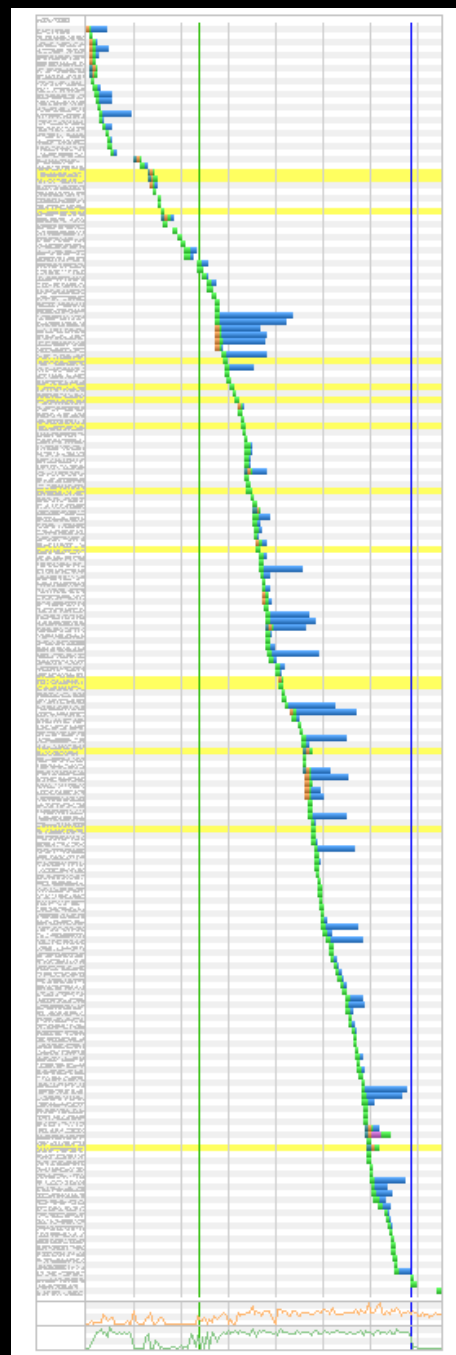
`my.cnf:`

`query_cache_type = 1`

`query_cache_size = 26214400`

PHP-Profiling

- XDebug
- XHProf



Web Page Performance Test for www.faz-institut.de/

From: Frankfurt, DE - IE9 - DSL
Tue 12 Mar 2013 21:54:59 CET

PageSpeed 1.12 Score: 67/100

[Need help improving?](#)

F	A	F	F	F	X
First Byte Time	Keep-alive Enabled	Compress Transfer	Compress Images	Cache static content	Effective use of CDN

[Summary](#) | [Details](#) | [Performance Review](#) | [PageSpeed](#) | [Content Breakdown](#) | [Domains](#) | [Screen Shot](#)
[Re-run the test](#)
[Raw page data](#) - [Raw object data](#)
[Export HTTP Archive \(.har\)](#)

	Load Time	First Byte	Start Render	DOM Elements	Document Complete			Fully Loaded		
					Time	Requests	Bytes In	Time	Requests	Bytes In
First View	6.506s	0.781s	4.841s	446	6.506s	34	964 KB	6.506s	34	964 KB
Repeat View	1.365s	1.098s	1.419s	446	1.365s	5	26 KB	2.118s	5	26 KB

Waterfall



Screen Shot





Latest Performance Report for: <https://www.deutschland.de/de>

[Download PDF](#)

Report generated: Tue, Mar 12, 2013, 1:45 PM -0700

Test Server Region: Vancouver, Canada

Using: Firefox 14.0.1, Page Speed 1.12.9.1, YSlow 3.1.4

Summary

Page Speed Grade:

(91%)↑

A

YSlow Grade:

(85%)↑

B

Page load time: 7.26s

Total page size: 1.87MB

Total number of requests: 70

Options

- ✦ [Re-Test Page](#)
- ✦ [Compare to another URL](#)

Share This Report



I got my scores; what now?

Start optimizing your site! But before you do:

- ✦ **Understand the recommendations**
They are meant to be generic, best practices; not everything will apply to your site.
- ✦ **Rules are sorted in order of impact upon score**
Optimizing rules at the top of

Breakdown

Page Speed

YSlow

Timeline

History

RECOMMENDATION

GRADE

TYPE

PRIORITY

Specify image dimensions

F (0)



Images

High

Defer parsing of JavaScript

F (20)



JS

High

Remove query strings from static resources

F (42)



Content

High

Optimize images

E (56)



Images

High

Remove unused CSS

F (47.1)



CSS

Low

Use efficient CSS selectors

F (0)



CSS

Low

Minify JavaScript

B (87)



JS

High

Overview

Critical Path Explorer

High priority (1)

Enable compression

Medium priority (4)

Leverage browser caching

In-line Small CSS

Minify JavaScript

Avoid bad requests

Low priority (10)

Optimise images

Serve resources from a c...

Minify CSS

Minimise redirects

Optimise the order of styl...

Defer parsing of JavaScript

Specify a cache validator

Minify HTML

Remove query strings fro...

Specify a Vary: Accept-E...

Experimental rules (1)

Already done! (13)

Overview

The page Berliner.de | [Und was ist Dir heute s...](#) got an overall PageSpeed Score of **68** (out of 100). [Learn more](#)



This PageSpeed Report is generated for this page as it appears in desktop browsers. To get suggestions on how to optimize the performance of this page for mobile devices, generate a [mobile report](#).

Suggestion Summary

Click on the rule names to see suggestions for improvement.



- **High priority.** These suggestions represent the largest potential performance wins for the least development effort. You should address this item first:
[Enable compression](#)
- **Medium priority.** These suggestions may represent smaller wins or much more work to implement. You should address these items next:
[Leverage browser caching](#), [In-line Small CSS](#), [Minify JavaScript](#), [Avoid bad requests](#)
- **Low priority.** These suggestions represent the smallest wins. You should only be concerned with these items after you've handled the higher-priority one
[Optimise images](#), [Serve resources from a consistent URL](#), [Minify CSS](#), [Minimise redirects](#), [Optimise the order of styles and scripts](#), [Defer parsing of JavaScript](#), [Specify a cache validator](#), [Minify HTML](#), [Remove query strings from static resources](#), [Specify a Vary: Accept-Encoding header](#)
- **Experimental rules.** These suggestions are experimental, but do not affect the overall PageSpeed Score. Consider this item as a pointer to an area to explore, but your mileage might vary:
[Reduce request serialisation](#)
- **Already done!** There are no suggestions for these rules, since this page already follows these best practices. Good job!

MANNHEIM²SUCHE IN **MANNHEIM**²

Suchen

BÜRGER
INFORMATIONEN
SYSTEM**Grade**

Overall performance score 79 Ruleset applied: YSlow(V2) URL: http://www.mannheim.de/

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

Tweet

Share

F Make fewer HTTP requests**F**[Use a Content Delivery Network \(CDN\)](#)**A** [Avoid empty src or href](#)**F** [Add Expires headers](#)**A** [Compress components with gzip](#)**A** [Put CSS at top](#)**A** [Put JavaScript at bottom](#)**B** [Avoid CSS expressions](#)**n/a** [Make JavaScript and CSS external](#)**A** [Reduce DNS lookups](#)**A** [Minify JavaScript and CSS](#)**Grade F on Make fewer HTTP requests**

This page has 3 external stylesheets. Try combining them into one.
 This page has 30 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](#)

Copyright © 2013 Yahoo! Inc. All rights reserved.

5 pages analyzed for <http://www.drupal.de>

Test performed: Wed Mar 13 00:25:45 CET 2013 with rules sitespeed.io-1.7 with user agent: Mozilla/6.0 viewport: 1280x800

Score

86.4 (86)

Critical Rendering Path Score

42.0 (40)

Time to first byte

0.315 s (0.264)

Number of JS synchronously in head

3.6 (4)

Number of JS

3.6 (4)

Number of CSS

4.4 (4)

Number of CSS images

3.8 (3)

Number of images

12.2 (14)

Number of requests

26.0 (30)

Requests without expires

0.0 (0)

Total page weight

892.0 kb (757.0)

Document page weight

4.6 kb (4.4)

Amount of text content

26.6 % (25.4)

Pages with SPOF

0

Number of SPOF per page

0.0 (0)

Number of DOM elements

210.2 (261)

Time spent in backend:

15.2 %

Time spent in frontend:

84.8 %

Cache time:

13 days (14 days)

Time since last modification:

292 days (200 days)

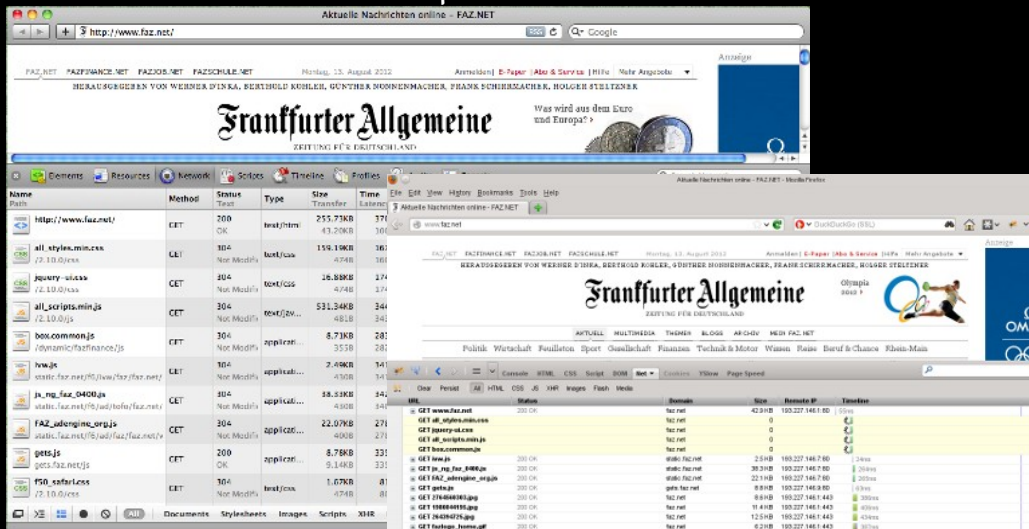
* The value inside of the parentheses are the median value (median)

Rules definition

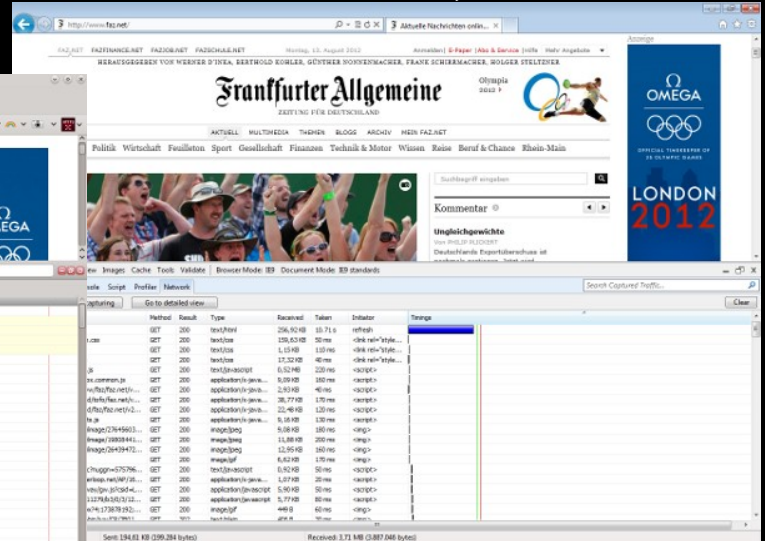
The rules are simple: **Green** color means you don't need to do anything (of course there could still be things you can do to get better performance). **Yellow** tells you that you should look into this, and guess what. **Red** means that something is really bad. The **blue** color are things for your information, no rules but things that will be good to know.

- **Sitespeed score are based on the rules (modified and enhanced Yslow rules):** Over 90 is good, between 80-90 is not so good, and below 80 is bad, really bad.
- **The number of requests:** Below 25 is good, between 25-88 is something you should look into (88 is the average amount of requests a site has, collected from <http://httparchive.org>) and above 88 is really bad.

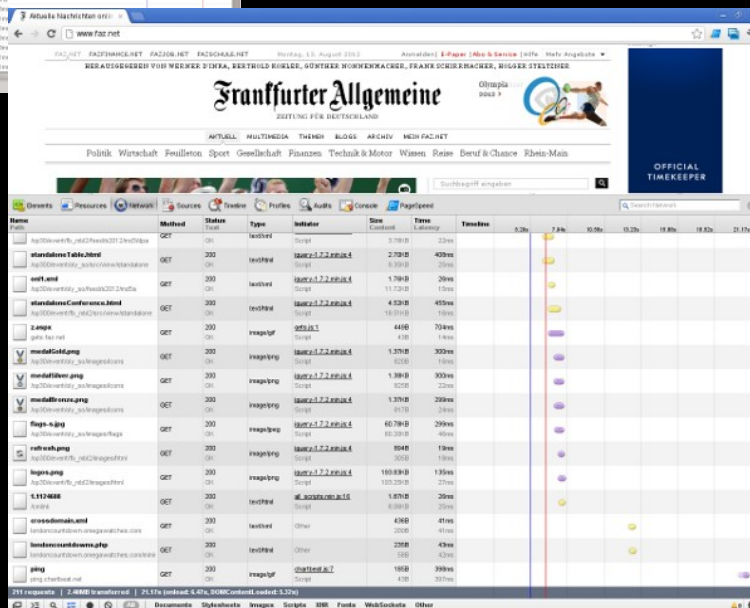
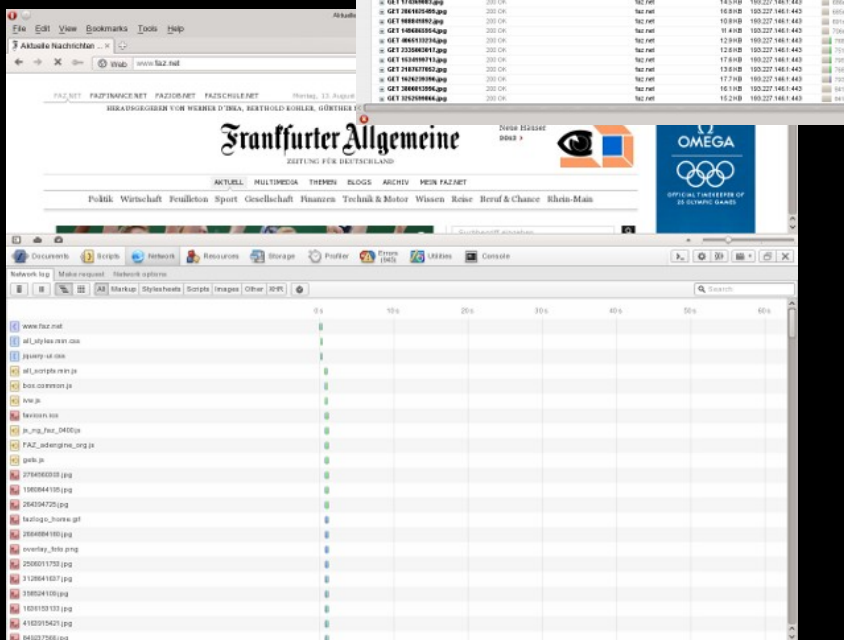
Safari Web Inspector



Internet Explorer Developer Tools



Firefox Firebug



Chrome Developer Tools

Opera Dragonfly

Komprimiere Output

```
<IfModule mod_deflate.c>
  <IfModule mod_setenvif.c>
    <IfModule mod_headers.c>
      SetEnvIfNoCase ^(Accept-EncodXng|X-cept-Encoding|X{15}|~{15}|-{15})$
      ^((gzip|deflate)\s*,?\s*)+|[X~-]{4,13}$ HAVE_Accept-Encoding
      RequestHeader append Accept-Encoding "gzip,deflate" env=HAVE_Accept-
      Encoding
    </IfModule>
  </IfModule>
<IfModule mod_filter.c>
  AddOutputFilterByType DEFLATE application/atom+xml \
                          application/javascript \
                          ...
                          text/css \
                          text/html \
                          text/plain \
                          text/x-component \
                          text/xml

</IfModule>
</IfModule>
```

<https://github.com/h5bp/server-configs/blob/master/apache/.htaccess>

Browser Cache

```
<IfModule mod_expires.c>
  ExpiresActive on
  ExpiresDefault "access plus 1 month"
  ExpiresByType text/html "access plus 0 seconds"
  ExpiresByType text/xml "access plus 0 seconds"
  . . .
  ExpiresByType application/rss+xml "access plus 1 hour"
  ExpiresByType image/gif "access plus 1 month"
  . . .
  ExpiresByType text/css "access plus 1 week"
  ExpiresByType application/javascript "access plus 1 week"
</IfModule>
```

```
<IfModule mod_headers.c>
  Header unset ETag
</IfModule>
```

FileETag None

<https://github.com/h5bp/server-configs/blob/master/apache/.htaccess>

PUBLIC



h5bp / server-configs

★ Star

< 890

🔗 Fork

< 145

Code

Network

Pull Requests 2

Issues 21

Wiki

Graphs

Webserver configuration templates for various platforms. Goes well with HTML5 Boilerplate. — [Read more](#)

ZIP

HTTP

SSH

Git Read-Only

<https://github.com/h5bp/server-configs.git>

Read-Only access



branch: master

Files

Commits

Branches 4

Tags

server-configs / +

🔒 423 commits

Add `CONTRIBUTING.md` file ...



alrra authored 12 days ago

latest commit efe3910aef

📁 [apache](#)

16 days ago

.htaccess: AddCharset & AddType require mod_mime [[alrra](#)]📁 [gae](#)

2 months ago

Change WOFF MIME type to `application/font-woff` [[alrra](#)]📁 [iis](#)

2 months ago

Change WOFF MIME type to `application/font-woff` [[alrra](#)]📁 [lighttpd](#)

3 months ago

Port wiki docs to README's [[necolas](#)]📁 [nginx](#)

2 months ago

Change WOFF MIME type to `application/font-woff` [[alrra](#)]📁 [test](#)

2 months ago

Change WOFF MIME type to `application/font-woff` [[alrra](#)]📄 [.gitattributes](#)

a year ago

Add gitignore and gitattributes files [[necolas](#)]📄 [.gitignore](#)

a year ago

Added package.json and removed it from .gitignore [[niftylettuce](#)]📄 [.gitmodules](#)

2 months ago

add node repository as a submodule [[AD7six](#)]<https://github.com/h5bp/server-configs>

App Cache

offline.appcache

CACHE MANIFEST

v1

assets/6/script/mainmin.js

assets/6/style/mainmin.css

assets/6/style/fonts/pro.ttf

assets/6/style/imgs/sprites1.png

HTML

<html manifest="offline.appcache">

Kombiniere CSS / Javascript

```
<link rel="stylesheet" type="text/css" href="style.css" />  
<link rel="stylesheet" type="text/css" href="forms.css" />  
<link rel="stylesheet" type="text/css" href="layer.css" />  
<link rel="stylesheet" type="text/css" href="jcarousel.css" />  
<link rel="stylesheet" type="text/css" href="lightbox.css" />
```

Wird

```
<link rel="stylesheet" type="text/css" href="combined.css?v=20120818" />
```

Oder

```
<link rel="stylesheet" type="text/css" href="combined-20120818.css" />
```

Minifiziere CSS / Javascript

Tools

- Yahoo! YUI Compressor (Java)
- Google Closure Compiler (Java)
- Uglify.js (Node)
- Microsoft Ajax Minifier (Windows)
- Assetic (PHP + externe Bibliotheken)

Optimiere Bilder

Tools

- ImageOptim (Mac)
- Trimage (Linux)
- RIOT (Windows)
- Kommandozeile:
 - jpegtran, jpegoptim, optipng, pngcrush, advpng, pngquant, gifsicle
- kraken.io / smush.it (Web)
- Google Pagespeed (Chrome / Firefox)

The h5bp ant build script

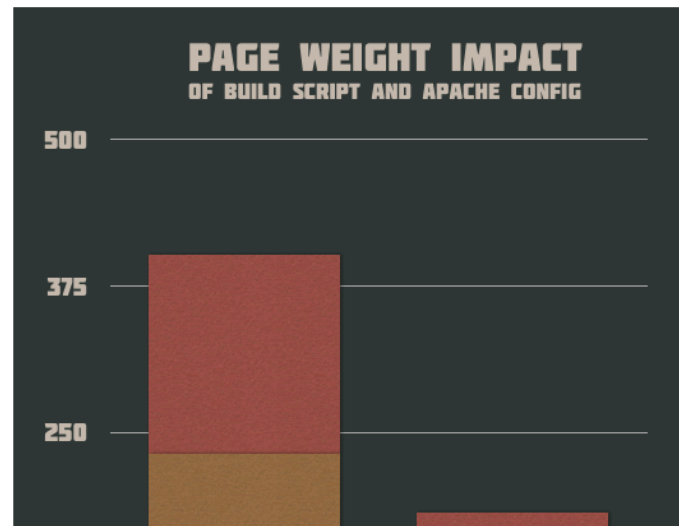
The build script is a tool that optimizes your code for production use on the web.

Why use it?

Faster page load times and happy end users :)

What it does

- Combines and minifies javascript (via yui compressor)
- Inlines stylesheets specified using `@import` in your CSS
- Combines and minifies CSS
- Optimizes JPGs and PNGs (with jpegtran & optipng)
- Removes development only code (any remaining console.log files, profiling, test suite)
- Basic to aggressive html minification (via htmlcompressor)
- Autogenerates a cache manifest file (and links from the `html` tag) when you enable a property in the project config file.
- Revises the file names of your assets so that you can use heavy caching (1 year expires).
- Upgrades the .htaccess to use heavier caching
- Updates your HTML to reference these new hyper-optimized CSS + JS files
- Updates your HTML to use the minified jQuery instead of the development version
- Remove unneeded references from HTML (like a root folder favicon)
- Runs your JavaScript through a code quality tool (optional)
- Optionally precompile LESS formatted CSS
- Optionally output JSDOC3 documentation



<https://github.com/h5bp/ant-build-script>

Make the Web Faster +1 3.6k

Overview

▼ PageSpeed

► Analysis

▼ Optimization

► Service

▼ mod_pagespeed

► Get Started

► Filter

Documentation

► Optimization SDK

► Public DNS

► Hosted Libraries

► Protocols

► Standards

► Performance Best Practices

► Articles

Tools

Community

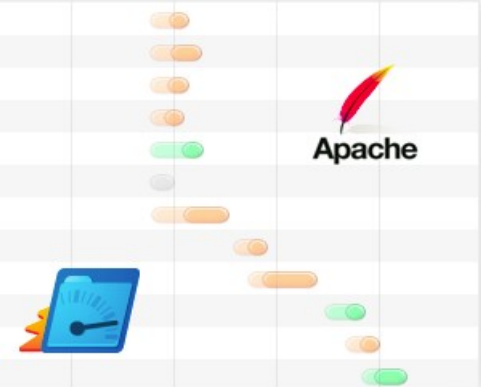
mod_pagespeed

mod_pagespeed speeds up your site and reduces page load time. This open-source Apache HTTP server module automatically applies [web performance best practices](#) to pages, and associated assets (CSS, JavaScript, Images) without requiring that you modify your existing content or workflow.

Features

- Automatic website and asset optimization
- Latest web optimization techniques
- 40+ configurable optimization filters
- Free, open-source, and frequently updated
- Deployed by individual sites, hosting providers, CDN's

Enabled	Status	Type	Initiator	Size	Time
☒	200	text/javascript	isapi:21	0B	61ms
☒	200	application/x-javascript	isapi:21	0B	213ms
☒	200	text/css	format+en.def...	0B	90ms
☒	200	text/javascript	isapi:21	0B	59ms
☒	200	text/css	isapi:21	0B	112ms
☒	200	application/x-javascript	isapi:21	0B	164ms
☒	200	text/css	/speed/.49	0B	25ms
☒	200	text/css	/speed/.63	0B	244ms
☒	200	text/css	/speed/.319	0B	199ms
☒	200	text/html	in.js:18	0B	131ms
☒	200	text/html	jquery.min.js:18	0B	331ms



Get Started

- [Download and install](#)
- [Review documentation](#)
- [Frequently asked questions](#)
- [Discuss mod_pagespeed](#)
- [Check mod_pagespeed announcements](#)
- [Learn about performance best practices](#)

Learn about mod_pagespeed



Users & Partners

- 120,000+ optimized, individual web sites
- Hosting providers and CDN's, including:



How does mod_pagespeed speed up web-sites?

<https://developers.google.com/speed/pagespeed/mod>

CSS Sprites



```
.flag {
  display: inline-block;
  width: 16px;
  height: 11px;
  overflow: hidden;
  background: transparent url(flags.png) scroll -1px -1px no-repeat;
}

.europeanunion
  {background-position: -1px -1px;}

.AUD
  {background-position: -1px -287px;}

.BRL
  {background-position: -1px -40px;}

.BGN
  {background-position: -1px -27px;}

.CAD
  {background-position: -1px -52px;}
```

DATA URIs

Kommando

```
$ base64 -w 0 fb.png > fb.png.txt  
$ php -r "echo base64_encode(file_get_contents('fb.png'));" > fb.png.txt
```

CSS

```
a.facebook {background-image: url('data:image/png;base64,iVBORw...g==');}
```

< Internet Explorer 8

HTML

```
<link rel="stylesheet" type="text/css" href="style.css" />  
<!--[if lt IE 8]>  
<link rel="stylesheet" type="text/css" href="lt-ie-8.css" />  
<![endif]-->
```

CSS

```
a.facebook {background-image: url('facebook.png');}
```


DATA URIs

HTML

```
<img src='data:image/png;base64,iVBORw...g==' alt='' />
```

Adaptive Images

Deliver small images to small devices

Adaptive Images detects your visitor's screen size and automatically creates, caches, and delivers device appropriate re-scaled versions of your web page's embeded HTML images. No mark-up changes needed. It is intended for use with [Responsive Designs](#) and to be combined with [Fluid Image](#) techniques.

Why? Because your site is being increasingly viewed on smaller, slower, low bandwidth devices. On those devices your desktop-centric images load slowly, cause UI lag, and cost you and your visitors un-necessary bandwidth and money. Adaptive Images fixes that.



Features

- Works on your existing site
- Requires no mark-up changes
- Device agnostic
- Mobile-first philosophy
- Easy & powerful customisations
- Up and running within minutes

[See more features](#)

Set-up

- Add `.htaccess` and `adaptive-images.php` to your document-root folder.
- Add one line of JavaScript into the `<head>` of your site.
- Add your CSS Media Query values into `$resolutions` in the PHP file.

You're done. Seriously.

Requirements

- Apache 2
- PHP 5.x
- GD lib *

* Normally a default with PHP

Javascript am Ende

Also:

```
</div>  
<script src=scripts.js></script>  
</body>  
</html>
```

Anstatt:

```
<head>  
<script src=scripts.js></script>  
</head>  
<body>
```

Alternativ:

```
<script src=scripts.js defer></script>
```

Drittanbieter

>

Single Point of Failure (SPOF)

Parallele bzw. nicht-blockierende Downloads

Aus:

```
<script src="//connect.facebook.net/de_DE/all.js"></script>
```

Wird:

```
<script>
(function(d, s, id) {
  var js, fjs = d.getElementsByTagName(s)[0];
  if (d.getElementById(id)) return;
  js = d.createElement(s); js.id = id;
  js.src = "//connect.facebook.net/de_DE/all.js";
  fjs.parentNode.insertBefore(js, fjs);
}(document, "script", "facebook-jssdk"));
</script>
```

Oder:

```
<script src="//connect.facebook.net/de_DE/all.js" async></script>
```

Webfonts

Aus:

```
<link rel='stylesheet' id='fontdeck-css'  
      href='http://f.fontdeck.com/s/css/qjAxanDb3GzL8rviVV7PWUySExw/*.smashingmagazine.c  
om/12777.css?ver=1363158490' type='text/css' media='screen, print' />
```

Wird:

```
<script>  
WebFontConfig = {  
  fontdeck: {  
    id: '12777'  
  }  
};  
(function() {  
  var wf = document.createElement('script');  
  wf.src = ('https:' == document.location.protocol ? 'https' : 'http') +  
    '://ajax.googleapis.com/ajax/libs/webfont/1/webfont.js';  
  wf.type = 'text/javascript';  
  wf.async = 'true';  
  var s = document.getElementsByTagName('script')[0];  
  s.parentNode.insertBefore(wf, s);  
})();  
</script>
```

https://developers.google.com/webfonts/docs/webfont_loader

Vergleich

Original



6.2

SPOF



22.8

Abmessungen festlegen

```

```


MP4 moov atom

```
qt-faststart input.mp4 output.mp4
```

Oder:

```
ffmpeg -i input.mp4 -c:a copy -c:v copy  
-movflags faststart output.mp4
```

Mehr Infos

<http://stevesouders.com/hpws/rules.php>

<http://browserdiet.com/>

<http://www.bookofspeed.com/>

<http://calendar.perfplanet.com/>

Fazit

- Caching auf mehreren Ebenen
- Weniger Requests
- Kleinere Datenmengen
- Parallele Downloads
- Optische Täuschung

Walter Ebert

@wltrd

walterebert.de

slideshare.net/walterebert

drupal.org/user/699574