



Foto: Michael Blum / thinkstockphotos.de

Mario Träger

SCHNELL, SCHNELLER, AUSREICHEND – PageSpeed-Optimierung für Online-Shops

Im April 2010 verkündete Google offiziell, dass die Ladezeit einer Webseite ein weiterer Bestandteil des Google-Rankings sei. Schnell war das Thema PageSpeed in aller Munde und Google stellte Tools für das Messen und Verbessern bereit. Die Ladezeitorientierung ist jedoch kein Selbstläufer. Im Gegenteil, gerade für größere Online-Shops sind die von Google vorgeschlagenen Verbesserungsmaßnahmen oftmals eine echte Herausforderung. Mario Träger zeigt die Bedeutung der Ladezeitorientierung auf und gibt wichtige Tipps für Shopbetreiber. Zusätzlich wurden die Top 100 Online-Shops analysiert, um herauszufinden, auf welche Herausforderungen Online-Shops bei der PageSpeed-Optimierung stoßen.

DER AUTOR



Mario Träger ist Partner der Agentur Webworks (www.webworks-agentur.de). Er unterstützt große und mittelständische Online-Shops im Bereich SEO und PageSpeed. Zuletzt war er bis 2014 Head of SEO der Witt Gruppe (member of the otto group).

Oftmals hat die Ladezeitorientierung bei Online-Shop-Betreibern keinen großen Stellenwert und läuft als unliebsame Zusatzmaßnahme „nebenbei“ mit. Zudem interessiert sich hierfür meist nur das SEO-Team, da es für einen guten SEO selbstverständlich ist, sämtliche Ranking-Faktoren zu beachten und entsprechend zu optimieren. Doch nicht nur aus SEO-Sicht kann sich die Ladezeitorientierung positiv auf den Online-Shop auswirken.

Warum sich PageSpeed-Optimierung für Online-Shops lohnt

Die Frage, warum PageSpeed für Online-Shops wichtig ist, lässt sich relativ deutlich mit drei Argumenten beantworten:

PageSpeed hat Auswirkungen auf das Google-Ranking

Der PageSpeed wurde 2010 von Google offiziell als Ranking-Faktor vorgestellt. Seitdem wurde er mehr oder weniger zu einem festen Bestandteil der **SEO-Optimierung**. Der PageSpeed beeinflusst SEO in zweifacher Hinsicht:

- » direkter Einfluss (als Teil des Ranking-Algorithmus)
- » indirekter Einfluss (durch User-Signale)

Der direkte Einfluss ergibt sich aus der Tatsache, dass die Ladezeit einer Webseite in den Google-Ranking-Algorithmus mit einfließt. Seiten, die nur sehr langsam laden, erleiden Nachteile in der Google-Suche. Matt Cutts teilte in einem Google-Webmaster-Video Folgendes mit:

„If your site is really, really slow, we've said that we do use page speed in our rankings. And so all of the things being equal, yes, a site can rank lower.“ (Quelle: <http://einfach.st/ytsp1>)

Der indirekte Einfluss ergibt sich aus dem Verhalten der Nutzer auf der eigenen Webseite – bei längerer Ladezeit steigt die Bounce-Rate bei gleichzeitig sinkender Verweildauer auf der Webseite. Dieses Nutzerverhalten wird durch Google beobachtet und fließt ebenfalls in den eigenen Ranking-Algorithmus ein. Es ist davon auszugehen, dass das Nutzerverhalten in Zukunft einen noch größeren Stellenwert seitens Google erfahren wird.

PageSpeed hat Auswirkungen auf den Google-AdWords-Qualitätsfaktor

Neben dem „SEO-Effekt“ hat der PageSpeed auch einen Einfluss auf den Google-Qualitätsfaktor und damit auf die Werbeeffizienz bei Google Adwords. Der Google-Adwords-Qualitätsfaktor wird für jedes Keyword, welches man in den bezahlten Suchanzeigen von Google schalten will, berechnet. Google schätzt mit diesem Wert die Qualität der Anzeige und der Zielseite. Ein hoher Qualitätsfaktor bedeutet, dass die Anzeige sowie die Zielseite für den Nutzer relevant sind. Zusammen mit dem maximalen Klickpreis, den man pro Klick auf eine Anzeige zu bezahlen bereit ist, fließt der Qualitätsfaktor in das Ranking der eigenen Anzeige mit ein. Ein hoher Qualitätsfaktor kann entweder ein besseres Ranking in der bezahlten Suche oder aber einen günstigeren Klickpreis bedeuten.

Die Wirkung des PageSpeed auf den CPC bzw. die Position der eigenen Werbeanzeige wird durch Google offiziell bestätigt:

„Wenn es nach dem Klick auf Ihre Anzeige zu lange dauert, bis die

Website geladen wird, geben Nutzer häufig auf und verlassen Ihre Website. Dieses unerwünschte Verhalten ist für Google ein Anzeichen für eine schlechte Zielseiten-erfahrung und kann sich negativ auf Ihren Anzeigenrang auswirken. Daher sollten Sie die Ladezeit Ihrer Zielseite möglichst optimieren.“ (Quelle: <http://einfach.st/adw21>)

PageSpeed hat Auswirkungen auf die Conversion-Rate

Das wohl wichtigste Argument im Zusammenhang mit der PageSpeed-Optimierung ist die Verbesserung der User Experience sowie die dadurch steigende Conversion-Rate. Je schneller eine Webseite lädt, umso „wohler“ fühlt sich auch der Kunde. Im Folgenden eine kleine Auswahl verschiedener Case Studies:

» Walmart erhöhte die Conversion-Rate bei jeder Sekunde Ladezeit – Verbesserung um 2 %.

Walmart.com konnte laut einer internen Studie für jede Sekunde Ladezeitverkürzung eine Erhöhung von 2 % Conversion-Rate erzielen. Zusätzlich steigerte jede Verbesserung der Ladezeit um 100 ms den Umsatz um 1 %. Darüber hinaus konnten Verbesserungen im SEO-Ranking sowie eine Verringerung der Bounce-Rate festgestellt werden. (Quelle: www.slideshare.net/devonauerswald/walmart-pagespeeds-lide)

» Amazon fand heraus, dass jede 100 ms an zusätzlicher Ladezeit 1 % an Conversion kostet.

Amazon fand 2006 mithilfe eines A/B-Tests heraus, dass jede 100 ms Verzögerung der Ladezeit 1 % Conversion-Rate kostet. (Quelle: <https://sites.google.com/site/glinden/Home/StanfordDataMining.2006-11-28.ppt?attredirects=0>)

» Shopzilla erreichte eine Verbesserung der Conversion-Rate um 7 bis 12 % durch die Reduzierung der Ladezeit um fünf Sekunden.

Shopzilla konnte 2008 bei der Reduzierung der Ladezeit um fünf Sekunden eine Verbesserung der Conversion-Rate um 7 bis 12 % sowie eine Erhöhung der Page Views um 25 % beobachten. (Quelle: <http://einfach.st/timsl>)

Den eigenen PageSpeed zu optimieren, bringt also nicht nur SEO-technische Vorteile mit sich. Insbesondere eine webseitenübergreifende Verbesserung der Conversion-Rate dürfte zweifelsohne das stärkste Argument sein, warum der PageSpeed ein entscheidender Hebel im Online-Marketing ist.

Die Top 100 Online-Shops in Deutschland und die PageSpeed-Optimierung

Fest steht nun, PageSpeed ist wichtig und kann den Umsatz eines Online-Shops deutlich steigern. Wie gehen aber die Top 100 Online-Shops in Deutschland mit dem Thema PageSpeed um? Um das herauszufinden, wurden die Top 100 Online-Shops (Quelle: statista 2014) mithilfe des Google PageSpeed Insights Tools (Link: <http://einfach.st/gdev8>) analysiert und die gewonnenen Daten aufbereitet.

IST Analyse der Top 100 Online-Shops

Das Google PageSpeed Insights Tool bietet einen hervorragenden Einstieg in das Thema Ladezeitorientierung. Mithilfe des Tools kann für jede Webseite ein PageSpeed-Wert ermittelt werden. Dieser bewegt sich auf einer Skala von 0 bis 100, wobei 100 für das Optimum steht. Neben dem PageSpeed-Wert zeigt Google getrennt nach Mobile und Desktop verschiedenste Optimierungsmaßnahmen zur Verbesserung des Page-

Speeds auf. Mithilfe dieser Informationen kann man den eigenen Online-Shop optimieren, den PageSpeed dadurch verbessern und eine bessere Seitenladezeit erreichen.

Mithilfe eines Ampelsystems teilt Google die Werte in „gut“, „neutral“ und „schlecht“ ein. Den PageSpeed von 0 bis 64 kennzeichnet Google als schlecht, den Wert von 65 bis 84 als neutral und ab dem Wert 85 vergibt Google eine positive Bewertung.

Um eine durchschnittliche Bewertung der Shops gewährleisten zu können, wurde ein Mittelwert aus Startseite, Kategorienseiten und Produktseiten erhoben. Dabei wurde nur der PageSpeed für die Desktop-Variante berücksichtigt. In Abbildung 1 ist der aktuelle Stand deutscher Top 100 Online-Shops im Bereich der PageSpeed-Bewertung zu sehen.

Auf den ersten Blick kann man erkennen, dass über 55 % der Top 100 Online-Shops eine neutrale Bewertung erhalten. Nur 6 % der Online-Shops werden von Google als positiv bewertet. Erschreckend ist die Tatsache, dass fast 40 % der Shops von Google als „schlecht“ in Bezug auf die Ladezeit eingestuft werden.

Der durchschnittliche PageSpeed-Wert aller Top 100 Online-Shops liegt momentan bei 68 von 100 und befindet sich damit noch knapp im neutralen Bereich.

In Sachen PageSpeed-Bewertung sieht es für die Top 100 Online-Shops somit nicht besonders rosig aus. Um die Ursachen hinter den Ergebnissen verstehen zu können, ist es notwendig, sich die größten von Google vorgeschlagenen Optimierungshebel anzusehen (siehe Abbildung 2).

Man kann sehr gut erkennen, dass das Thema „JavaScript- und CSS-Rendering“ 74 % der Online-Shops betrifft und somit das Hauptproblem der Google-PageSpeed-Optimierung darstellt, dicht gefolgt von den Optimierungs-



Abb. 1: Auswertung der Top 100 Online-Shops in Deutschland nach PageSpeed-Bewertung

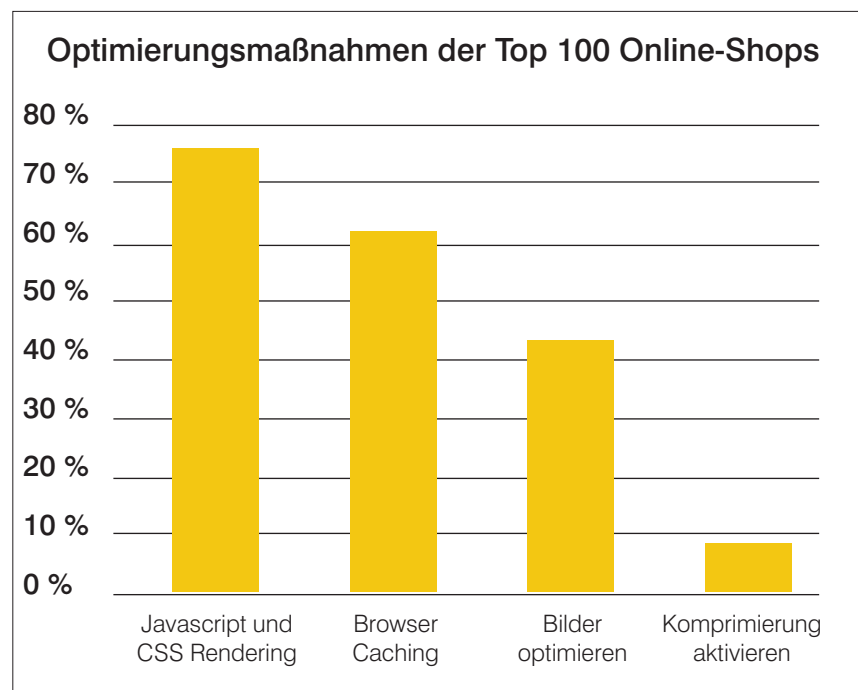


Abb. 2: Auswertung der Top 100 Online-Shops in Deutschland nach Optimierungsmaßnahmen

maßnahmen „Browser-Caching“ mit 62 % und „Bilder optimieren“ mit 44 %. Die Optimierungsmaßnahme „Komprimierung aktivieren“ betrifft immerhin noch 9 % aller Top 100 Online-Shops. Weitere Maßnahmen betreffen einzelne Shops oder fallen nur leicht ins Gewicht.

Im Nachfolgenden werden die vier Optimierungsmaßnahmen kurz erklärt und es wird auf die Probleme in Verbindung mit Online-Shops näher eingegangen.

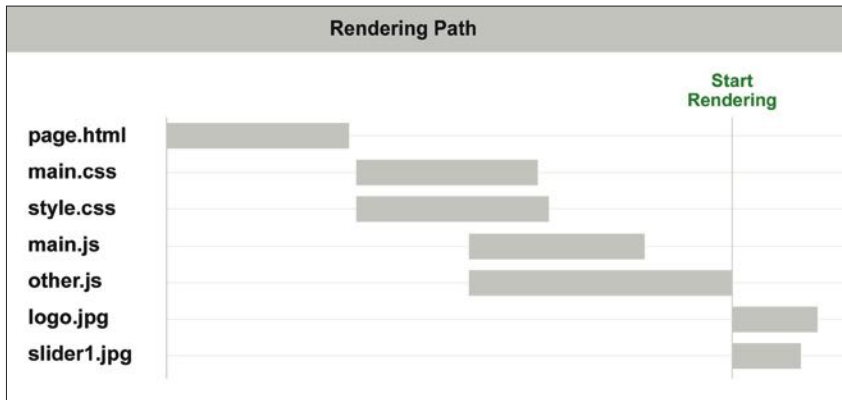


Abb. 3: Beispiel für Rendering-Verzögerung durch CSS und JavaScript

JavaScript- und CSS-Ressourcen, die das Rendering blockieren, in Inhalten "above the fold" (ohne Scrollen sichtbar) beseitigen

Ihre Seite enthält 25 blockierende Skript-Ressourcen und 7 blockierende CSS-Ressourcen. Dies verursacht eine Verzögerung beim Rendern Ihrer Seite.

Keine der Seiteninhalte "above the fold" (ohne Scrollen sichtbar) konnten ohne Wartezeiten für das Laden der folgenden Ressourcen gerendert werden. Versuchen Sie, blockierende Ressourcen zu verschieben oder asynchron zu laden, oder laden Sie kritische Bereiche dieser Ressourcen direkt inline im HTML.

Entfernen Sie JavaScript, das das Rendering blockiert:

- <http://www.qvc.de/...store/DE/content/javascript/storetext.js>
- <http://e.monetate.net/js/2/a-d253ceb9/p/qvc.de/entry.js>
- <http://www.qvc.de/...re/DE/content/javascript/decustomdev0.js>
- <http://www.qvc.de/...ssetStore/javascript/jquery-1.9.1.min.js>

Abb. 4: PageSpeed Insights – JavaScript- und CSS-Rendern-Maßnahme bei qvc.de

JS- und CSS-Rendern optimieren

Werden JavaScript- und CSS-Dateien im Quellcode nicht richtig positioniert, kann es dazu kommen, dass diese Dateien das Rendern der Seite – also den Seitenaufbau – blockieren. Durch die Blockierung des Renderings verlängert sich die Zeit, in welcher sich der für den User sichtbare Bereich (above the fold) aufbaut. Im schlimmsten Fall erhält der Besucher beim Aufbau der Seite für die ersten Sekunden nur eine leere Seite. Erst danach werden Stück für Stück die Inhalte von oben nach unten geladen. Die Grafik in Abbildung 3 soll den sogenannten „Rendering Path“ nochmals verdeutlichen.

Zunächst wird die HTML-Datei vom Server angefordert und heruntergeladen. Danach beginnt der Browser, die HTML-Datei zu lesen, und bemerkt zwei CSS-Dateien. Das Rendering wird gestoppt, da die CSS-Dateien zunächst heruntergeladen werden müssen. Danach stößt der Browser auf zwei JavaScript-Dateien und muss auch diese

zunächst herunterladen. Erst danach lädt der Browser die Webseite weiter und beginnt somit, die ersten Bilder zu laden.

Das Rendering kann bereits bei einer geringen Anzahl an CSS- sowie JavaScript-Dateien enorm verzögert werden. Nicht selten verwenden Webseiten deutlich mehr Ressourcen und positionieren diese wild verteilt im Quellcode. Je nach Webseite muss hier der Rendering Path genau analysiert werden, um Optimierungsmöglichkeiten vorschlagen zu können.

Ziel hierbei ist es, nicht die gesamte Ladezeit zu optimieren, sondern das Anzeigen des sichtbaren Bereichs (above the fold) zu beschleunigen. Es müssen daher sämtliche Dateien, die für diesen Bereich zuständig sind, bevorzugt geladen werden. Dateien, die für die Darstellung des „Above the fold“-Bereichs nicht benötigt werden, sollten innerhalb des Quelltextes an das Ende verschoben oder im Hintergrund (asynchron) geladen werden.

Sieht man sich bspw. den Online-

Shop von QVC etwas genauer an, bemängelt Google, dass insgesamt 25 blockierende JavaScript-Ressourcen und sieben blockierende CSS-Ressourcen an einer Verzögerung des Renderings beteiligt sind.

Schwer ist es bei dieser Optimierungsmaßnahme, pauschale Aussagen zu treffen. Allgemein kann jedoch Folgendes gesagt werden:

- » JavaScript und CSS sollten stets nur notwendigen Code beinhalten, also aufgeräumt sein.
- » Sämtlicher Code sollte in möglichst wenig unterschiedlichen Dateien bereitgestellt werden. Eine Anzahl von 25 unterschiedlichen JavaScript-Dateien wie im oberen Beispiel sollte vermieden werden.
- » JavaScript sollte – wenn möglich – am Ende des Quellcodes geladen werden.
- » Das notwendige CSS für den sichtbaren Bereich der jeweiligen Seite sollte im Bestfall direkt im Head der Seite – wenn möglich auch inline (direkt im Quellcode) – geladen werden. Der restliche CSS-Code sollte am Ende des Quellcodes aufgerufen werden.

Es muss jedoch unbedingt erwähnt werden, dass die Umsetzung dieser Optimierungsmaßnahme bei Online-Shops oftmals sehr große technische Herausforderungen mit sich bringt. Sämtliche JavaScript- und CSS-Dateien müssen auseinandergenommen, auf Funktionalität überprüft und in kritische und unkritische Ressourcen aufgeteilt werden. Besonders für CSS-Dateien ist es ein komplexer Prozess, für jede Unterseite automatisiert festzustellen, welche Regeln gerade für den Content im sichtbaren Bereich (above the fold) benötigt werden und welche nicht.

Empfehlenswert ist es, die Optimierungsmaßnahme direkt in einen Shop-Relaunch zu integrieren und das

Rendering von Grund auf zu planen. Auf diesem Wege kann das Rendering der Seite grundlegend überarbeitet und nach einer ladezeitorientierten Logik neu aufgebaut werden.

Durch die Positionierung von Ressourcen kann das Rendering einer Seite deutlich verbessert werden. Je nach Online-Shop ist dies jedoch nur mit enormen technischen Herausforderungen umzusetzen.

Browser-Caching nutzen

Beim Besuch einer Seite speichert der Browser die Seite und deren einzelne Ressourcen (CSS, JS oder Bilder) im Zwischenspeicher (Cache) ab. Besucht man die Seite ein weiteres Mal, fragt der Browser beim Server nach, ob sich die jeweiligen Dateien geändert haben. Falls diese sich verändert haben, werden die Dateien neu angefordert. Haben sich die Dateien jedoch nicht verändert, lädt der Browser die Seite direkt aus dem Cache. Dadurch müssen diese Ressourcen nicht bei jedem Seitenaufruf neu vom Server angefordert werden, sondern werden direkt von der lokalen Festplatte des Nutzers geladen. Dies ermöglicht es, eine große Anzahl an PageSpeed raubenden Dateianforderungen (HTTP Requests) einzusparen.

Google PageSpeed empfiehlt für Ressourcen mindestens eine Haltbarkeit von einer Woche. Bei einigen Ressourcen kann es jedoch durchaus sinnvoll sein, höhere Zeiträume zu definieren.

Wichtig: Werden Änderungen innerhalb dieser Woche bspw. an CSS-Dateien durchgeführt, erhalten wiederkehrende Besucher diese im Zeitraum von einer Woche nicht. Dies kann man jedoch durch eine Versionierung der Dateinamen lösen. Anstatt die CSS-Datei nur style.css zu nennen, kann man diese bspw. nach dem letzten Änderungsdatum style-20-05-15.css benennen. Sobald der Name einer Datei geändert wird, greift die Caching-Zeit nicht mehr.

51 / 100 Übersicht über die Vorschläge

! Behebung erforderlich:

Browser-Caching nutzen

Das Festlegen eines Ablaufdatums oder eines Höchstalters in den HTTP-Headern für statische Ressourcen weist den Browser an, zuvor heruntergeladene Ressourcen über die lokale Festplatte anstatt über das Netzwerk zu laden.

Nutzen Sie Browser-Caching für die folgenden cachefähigen Ressourcen:

```
http://0815.mytoys.de/scripts/acfp.js (Ablaufdatum nicht festgelegt)
http://api.zanox.com/scripts/zanox.js (Ablaufdatum nicht festgelegt)
http://www.mytoys.de/...images/footer/footer_news_niam_button.png (Ablaufdatum nicht festgelegt)
http://www.mytoys.de/static/css/elements/mt_buttons.css (Ablaufdatum nicht festgelegt)
http://www.mytoys.de/static/css/elements/mt_forms.css (Ablaufdatum nicht festgelegt)
http://www.mytoys.de/static/css/elements/mt_sem.css (Ablaufdatum nicht festgelegt)
http://www.mytoys.de/static/css/feature.css (Ablaufdatum nicht festgelegt)
http://www.mytoys.de/static/css/font/fontello/fontello.ttf (Ablaufdatum nicht festgelegt)
http://www.mytoys.de/static/css/helpers/mt_clearfix.css (Ablaufdatum nicht festgelegt)
http://www.mytoys.de/static/css/helpers/mt_fonts.css (Ablaufdatum nicht festgelegt)
http://www.mytoys.de/static/css/helpers/mt_icons.css (Ablaufdatum nicht festgelegt)
```

Abb. 5: PageSpeed Insights – Browser-Caching-Maßnahme bei mytoys.de

```
<IfModule mod_expires.c>
  ExpiresActive On
  ExpiresByType text/html „access plus 1 week“
  ExpiresByType text/css „access plus 1 week“
  ExpiresByType application/javascript „access plus 1 month“
  ExpiresByType application/x-javascript „access plus 1 month“
  ExpiresByType image/jpeg „access plus 6 month“
  ExpiresByType image/png „access plus 6 month“
  ExpiresByType image/gif „access 1 year“
  ExpiresByType application/pdf „access 1 month“
  ExpiresByType image/x-icon „access plus 1 year“
</IfModule>
```

Code 1: Caching-Konfiguration in der .htaccess-Datei

Sieht man sich bspw. den Online-Shop von MyToys im PageSpeed Insights Tool an, fällt sofort auf, dass sämtlichen Dateiformaten kein bzw. nur ein kurzes Ablaufdatum gegeben wurde. Die Dateiformate besitzen eine Caching-Zeit von unter 8,5 Stunden (siehe Abbildung 5).

Mytoys könnte dieses Problem relativ einfach und ohne großen Aufwand mithilfe einer Anweisung innerhalb der .htaccess-Datei lösen. Das Browser-Caching lässt sich über die .htaccess-Datei mit dem Modul expires.c definieren (siehe Code 1).

Mithilfe diesen Codes erhalten die genannten Dateiformate verschiedene Caching-Laufzeiten, die mindestens über einer Woche liegen.

Zudem sollte man wissen, dass das Google PageSpeed Insights Tool auch Dateien von fremden Servern zeigt und

die Caching-Laufzeit bemängelt. Auf die Caching-Laufzeit von Fremddateien haben Sie jedoch keinen Zugriff und müssen daher damit leben.

Browser-Caching lässt sich relativ einfach und schnell über die .htaccess-Datei aktivieren und bietet somit die Möglichkeit, mit wenig Aufwand vor allem die Ladezeit für wiederkehrende Besucher zu reduzieren.

Bilder optimieren

Bilder sind eine der größten Ressourcen, welche eine Webseite zu laden hat. Laut httparchive.org/interesting.php verursachen Bilder über 60 % der Dateigröße aller Ressourcen. Zudem findet man vor allem in Online-Shops immer größere und aufgrund von Technikneuerungen wie Retina-Displays auch immer hochwertigere Bilder. Daher

bieten Bilder meist großes Potenzial bei der Optimierung der Ladezeit.

Der PageSpeed-Wert von *kfzteile24.de* entspricht für die Startseite nur 30 von 100. Das Optimieren von Bildern wird hierbei von Google als die wichtigste Maßnahme definiert – nicht verwunderlich, bei einer möglichen Einsparung von 1,2 MB (entspricht 40 %).

Bei *kfzteile24.de* stößt man gleich auf die erste Regel der Bilderoptimierung: Bilder müssen in der Bildabmessung abgespeichert werden, in welcher sie später Verwendung finden (nicht kleiner und nicht größer). Aktuell werden Bilder in einer Größe geladen, in welcher Sie später überhaupt nicht verwendet werden. Man lädt somit eine große Datei und braucht diese danach nur als kleines Bild.

Neben der Bildabmessung kann auch durch die Auswahl des richtigen Bildformates Dateigröße eingespart werden. Die zwei gängigsten Bildformate, welche man je nach Zweck im Internet verwendet, sind JPEG und PNG. Das Bildformat GIF wurde weitestgehend von PNG abgelöst, da es keinerlei Verbesserungen aufweist.

JPEG – Das Bildformat JPEG steht ausgeschrieben für *Joint Photographic Expert Group* und ist, wie der Name schon sagt, für Fotos vorgesehen. Der größte Vorteil ist die enthaltene, nicht verlustfreie Kompression. Dies hat den Vorteil, dass man den Kompromiss zwischen Qualität und Bildgröße komplett selbst steuern kann. JPEG eignet sich dadurch vor allem für Bilder, welche eine geringe Dateigröße aufweisen sollen. Der große Nachteil bei JPEG ist, dass die Qualität der Bilder durch jeden Bearbeitungsprozess abnimmt.

PNG – Das Bildformat PNG (*Portable Network Graphics*) ist im Gegensatz zu JPEG verlustlos. Will man Bilder in höchster Qualität darstellen, sollte man sich für PNG entscheiden. Der Nachteil ist jedoch, dass PNG deutlich mehr Speicherplatz benötigt als JPEG. Zudem

30 / 100 Übersicht über die Vorschläge

! Behebung erforderlich:

Bilder optimieren

Durch eine korrekte Formatierung und Komprimierung von Bildern können viele Datenbytes gespart werden.

Optimieren Sie die folgenden Bilder, um ihre Größe um 1,2 MB (40 %) zu reduzieren.

Durch die Komprimierung und Größenanpassung von https://www.kfzteile24.de/img/k24/banner/pc/newsletter_layer.png könnten 64,3 KB (66 %) eingespart werden.

Durch die Komprimierung und Größenanpassung von <https://www.kfzteile24.de/img/k24/imgnav/producers/meguiars.png> könnten 35,5 KB (71 %) eingespart werden.

Durch die Komprimierung von <https://www.kfzteile24.de/img/k24/banner/pc/teststieger.jpg> könnten 31,8 KB (25 %) ohne Verluste eingespart werden.

Durch die Komprimierung und Größenanpassung von <https://www.kfzteile24.de/img/k24/imgnav/producers/fom.png> könnten 28,9 KB (76 %) eingespart werden.

Durch die Komprimierung und Größenanpassung von <https://www.kfzteile24.de/img/k24/imgnav/producers/armorAll.png> könnten 27 KB (73 %) eingespart werden.

Durch die Komprimierung und Größenanpassung von <https://www.kfzteile24.de/img/k24/imgnav/producers/hella.png> könnten 26,3 KB (67 %) eingespart werden.

Durch die Komprimierung und Größenanpassung von <https://www.kfzteile24.de/img/k24/imgnav/producers/interos.png> könnten 26,3 KB (67 %) eingespart werden.

[Details ausblenden](#)

Abb. 6: PageSpeed Insights – Bilder-optimieren-Maßnahme bei *kfzteile24.de*

Bildformate im Vergleich		
JPEG	JPEG für Web	JPEG tinyjpg
		
Dateigröße: 229 KB	Dateigröße: 67 KB Ersparnis: 162 KB	Dateigröße: 63 KB Ersparnis: 4 KB
PNG	PNG für Web	PNG tinypng
		
Dateigröße: 265 KB	Dateigröße: 196 KB Ersparnis: 69 KB	Dateigröße: 52 KB Ersparnis: 144 KB

Abb. 7: Bildformate im Vergleich

Komprimierung aktivieren

Durch die Komprimierung der Ressourcen mit "gzip" oder "deflate" kann die Anzahl der über das Netzwerk gesendeten Bytes reduziert werden.

Ermöglichen Sie die Komprimierung der folgenden Ressourcen, um die Übertragungsgröße um 196,8 KB (72 %) zu reduzieren.

Durch die Komprimierung von http://www.hornbach.de/...s/media/_technik/js/player_3/jwplayer.js könnten 110,2 KB (72 %) eingespart werden.

Durch die Komprimierung von http://www.hornbach.de/...nik/css/modules/inspirations_element.css könnten 20,2 KB (76 %) eingespart werden.

Durch die Komprimierung von http://www.hornbach.de/...chnik/css/modules/teaserelements_css.css könnten 20,1 KB (74 %) eingespart werden.

Durch die Komprimierung von http://www.hornbach.de/...media/_technik/js/jquerycarouselmin.js könnten 10,9 KB (71 %) eingespart werden.

Durch die Komprimierung von http://www.hornbach.de/...hnik/css/modules/standart_teaser_css.css könnten 7,5 KB (80 %) eingespart werden.

Durch die Komprimierung von http://www.hornbach.de/...media/_technik/js/jquerytouchswipe.min.js könnten 7,4 KB (66 %) eingespart werden.

Durch die Komprimierung von http://www.hornbach.de/...media/_technik/js/jquerycontrol.js könnten 6,6 KB (76 %) eingespart werden.

[Details ausblenden](#)

Abb. 8: PageSpeed Insights – Komprimierung-aktivieren-Maßnahme bei *hornbach.de*

wird PNG bevorzugt von Webdesignern eingesetzt, da das Bildformat die Möglichkeit der Transparenz bietet.

Es sollte also stets überdacht werden, ob PNG-Bilder notwendig sind. Jedoch sollte auch bei JPEG-Bildern an die Optimierung gedacht werden.

Bildkomprimierung – Bei der Bildkomprimierung verfolgt man das Ziel, das Originalbild bei möglichst gleichbleibender Bildqualität zu komprimieren, um möglichst viel Dateigröße einzusparen. Zunächst wird die Komprimierung mit dem Bildbearbeitungsprogramm Adobe Photoshop vorgenommen.

Des Weiteren werden *Image-Crush-Tools* verwendet, welche durch zusätzliche Methoden nochmals Dateigröße einsparen können.

Das Bildbearbeitungsprogramm Adobe Photoshop bietet extra für das Web einen eigenen Menüpunkt namens „Für Web und Geräte abspeichern“ an. Dort hat man die Möglichkeit, Bilder/Grafiken als JPEG oder PNG abzuspeichern. Im Falle von JPEG kann man detailliert die Qualität des Bildes regeln und sieht durch eine „Vorher-Nachher-Ansicht“, wie sich die Qualität des Bildes verändert. Außerdem lässt sich erkennen, wie sich die Qualität auf die Dateigröße auswirkt. Soll das Bild hingegen als PNG-Format abgespeichert werden, sind die Einstellungsmöglichkeiten in Adobe Photoshop eingeschränkt. In diesem Format ist es nicht möglich, die Komprimierung des Bildes selbst zu regeln. Doch auch hier führt die Funktion „Für Web und Geräte abspeichern“ zu massiven Einsparungen bei der Dateigröße.

Durch die Bildoptimierung mit Photoshop können bereits sehr gute Ergebnisse erreicht werden. Bei einer zusätzlichen Bearbeitung der eigenen Bilder mit einem Image-Crush-Tool kann die Einsparungsgröße nochmal übertroffen werden. Diese Tools verwenden besondere Optimierungstechniken, um unnötige Bytes aus Bilddateien zu entfernen. Solche Tools sind bspw. tinypng.com für PNG und tinyjpg.com für JPEG. Jedoch sollte hier stets das Ergebnis im Auge behalten und nicht blind auf diese Tools vertraut werden.

In Abbildung 7 sieht man einen Überblick über die Bildformate JPEG und PNG sowie über die verschiedenen Optimierungsmethoden und deren Ergebnisse.

Wie schon angesprochen, sollte man im letzten Schritt bei der Bildkomprimierung mit Image-Crush-Tools auf die Qualität achten und bei Notwendigkeit diesen Punkt überspringen.

Bunte Mischung gesucht!



Für den Standort Augsburg:

» Team Leader SEO
(m/w)

» Team Leader
Web Development
(m/w)

MIT MEINEM TEAM
SETZE ICH INNOVATIVE,
INTERDISZIPLINÄRE
PROJEKTE FÜR UNSERE
KUNDEN UM

Christoph Mann



Alle Infos und weitere Jobs unter:

explido.de/karriere

```
<IfModule mod_deflate.c>
  AddOutputFilterByType DEFLATE application/javascript
  AddOutputFilterByType DEFLATE application/x-javascript
  AddOutputFilterByType DEFLATE application/xhtml+xml
  AddOutputFilterByType DEFLATE application/xml
  AddOutputFilterByType DEFLATE font/opentype
  AddOutputFilterByType DEFLATE font/otf
  AddOutputFilterByType DEFLATE font/ttf
  AddOutputFilterByType DEFLATE image/svg+xml
  AddOutputFilterByType DEFLATE image/x-icon
  AddOutputFilterByType DEFLATE text/css
  AddOutputFilterByType DEFLATE text/html
  AddOutputFilterByType DEFLATE text/javascript
  AddOutputFilterByType DEFLATE text/plain
  AddOutputFilterByType DEFLATE text/xml
</IfModule>
```

Code 2: GZip-Konfiguration in der .htaccess-Datei

Neben den genannten Maßnahmen der Bildoptimierung gibt es noch zwei weitere Methoden, welche in Verbindung mit Online-Shops unbedingt angesprochen werden müssen:

Verwendung von LazyLoading – Bei größeren Seiten mit vielen Bildern, wie bspw. Rasterseiten in einem Online-Shop, kann es auch Sinn machen, LazyLoading einzusetzen. Dies ist eine Technik, die Bilder erst vom Server anfordert, sobald der Benutzer diese sieht. Dies bedeutet, dass Bilder erst geladen werden, wenn sie in den sichtbaren Bereich gelangen. Normalerweise werden sämtliche Bilder der Seite beim ersten Aufruf geladen, egal ob sie überhaupt gesehen werden oder nicht.

Verwendung von WebP – Unter WebP versteht man ein von Google entwickeltes Bildformat. Mit WebP kann man gegenüber JPEG bis zu 80 % der Dateigröße sparen, ohne dabei an Bildqualität einzubüßen. Einziger Nachteil von WebP ist, dass es nur unter Chrome und Opera funktioniert. Hat man jedoch einen hohen Anteil an Chrome-Usern, ist die Verwendung von WebP trotzdem lohnenswert. Hierbei muss man alle Bilder unter zwei Bildformaten zur Verfügung stellen, nämlich WebP für Chrome und Opera und bspw. JPEG für die restlichen Browser.

Die Zuordnung der Bilder steuert eine Regel innerhalb der .htaccess-Datei. Würde man das obere Bild in WebP umwandeln, erhält man eine Version mit einer Dateigröße von nur 26 KB.

Bilder sollten durchgehend optimiert und stets in der bestmöglichen Dateigröße abgespeichert und verwendet werden.

Komprimierung aktivieren (Gzip)

Gzip gilt als die einfachste Technik zur Reduzierung der Seitengröße, die gleichzeitig aber auch den stärksten positiven Einfluss auf den PageSpeed hat. Mithilfe von Gzip wird die Übertragungsgröße durch das Komprimieren von Dateien bei der Auslieferung vom Webserver verringert.

Wurde Gzip auf dem Server aktiviert, kann der Webserver die Datei in gepackter Version an den Browser ausliefern. Der Browser wiederum entpackt die Datei und öffnet sie.

Probleme im Bereich Komprimierung kann man bspw. bei Hornbach.de feststellen. Indem JavaScript- und CSS-Dateien mithilfe von Gzip komprimiert werden, können allein auf der Startseite ca. 197 KB (entspricht 72 %) eingespart werden.

Um Gzip bspw. mit Hilfe von mod_deflate (je nach Server) zu aktivieren,

ist folgender Code in der .htaccess-Datei notwendig (siehe Code 2).

Wichtig: Bereits komprimierte Dateien wie Bilder sollten von einer Gzip-Komprimierung ausgeschlossen werden. Durch eine weitere Komprimierung wird zusätzlicher Aufwand produziert und es kann sogar sein, dass die Dateien größer als vorher sind.

Die Aktivierung der Komprimierung mithilfe von Gzip gilt als die einfachste Technik zur Reduzierung der Seitengröße und bietet daher ein großes Verbesserungspotenzial.

Die Optimierungsmaßnahme „JS- und CSS-Rendering optimieren“ lässt sich bei vielen Shops nur schwer sinnvoll und kurzfristig umsetzen. Hierfür sollte ein Shop-Relaunch genutzt werden. Für die Maßnahmen „Browser-Caching“, „Bilder optimieren“ und „Komprimierung aktivieren“ gibt es so gut wie keine Ausreden, diese nicht umzusetzen. Trotzdem besteht weiterhin die Frage, warum dies ein so großer Anteil der Top-Shops aktuell nicht macht.

Es kann davon ausgegangen werden, dass das Thema PageSpeed in Zukunft einen noch höheren Stellenwert im Ranking-Algorithmus von Google haben wird – gerade in Hinsicht auf mobile Seiten, die langsam laden. Diese sind nicht nur schlecht aus SEO-Sicht – sie sind auch wenig benutzerfreundlich. Wer die Entwicklungen und Ankündigungen seitens Google regelmäßig beobachtet, wird seine eigenen Schlüsse ziehen können, wie wichtig das In-Einklang-Bringen des Nutzerinteresses mit den Anforderungen der Suchmaschinenoptimierung bereits ist und auch noch werden wird. ¶