



WEBPERFORMANCE

Eine Präsentation von Sergej Fedorchenko, Sebastian Schmitt, Kevin Thielen und Michael Landreh

Agenda / Überblick

1

Einführung

2

Contentseitige Optimierung

3

Serverseitige Optimierung

4

Optimierung für mobile Endgeräte

Agenda / Kapitel 1 - Einführung

1

Einführung

2

Contentseitige Optimierung

3

Serverseitige Optimierung

4

Optimierung für mobile Endgeräte

Definition von Webperformance

Einführung / Definition Performance

„The accomplishment of a given task measured against preset known standards of accuracy, completeness, cost, and speed. *In a contract, performance is deemed to be the fulfillment of an obligation, in a manner that releases the performer from all liabilities under the contract.*”

Definition von Webperformance

Einführung | Definition „Accuracy“ und „Completeness“

- **„Accuracy“** (*Treffsicherheit; Genauigkeit*)
 - Wie genau wird die Aufgabe erledigt?
 - Beschreibt wie reproduzierbar das Ergebnis ist
 - Abhängig von der Aufgabe; manchmal irrelevant
 - Oft in Prozent oder Intervallen angegeben
- **„Completeness“** (*Vollständigkeit*)
 - Wie vollständig ist das Ergebnis?
 - Wie viel fehlt?
 - Wie viel gehört hinein (bei nicht absoluten Aufgaben)?
- Maximierbare Faktoren -> Keine Differenzierung möglich

Definition von Webperformance

Einführung | Definition „Costs“ und „Speed“

- **„Costs“** (*Kosten*)
 - Was kostet das Erledigen der Aufgabe?
 - Gegengewicht zu den Anderen Faktoren
 - Interessen Abwägung
 - Am Eindeutigsten bei gleichen sonstigen Faktoren
- **„Speed“** (*Geschwindigkeit*)
 - Wie schnell wird die Aufgabe oder Teilaufgabe erledigt?
 - Verschiedene Geschwindigkeiten („Load Time“, „Time-To-First-Byte“, „Response Time“ usw.)
 - Besonders Relevant: „Response Time“ und „Load Time“
 - Größter differenzierender Faktor (!)
 - Maximum ist nicht absolut

Identifikation schlechter Webperformance über „Bucket Tests“

Einführung | „Bucket Test“

- Testmethode: „Bucket Test“
 - Zwei verschiedene Versionen einer Seite
 - Versionen werden verschiedenen Gruppe („Buckets“) zugänglich gemacht
 - Unterschiede in den „Buckets“ werden gemessen.
- Probleme der Methode:
 - Negative Erfahrungen für einen Teil der Testgruppe -> Verlust von Usern
- Lösung:
 - Microsoft, Google, Yahoo haben Test veröffentlicht

Ein Beispiel schlechter Performance

Einführung | „Bucket Test“ – Beispiel „Bing“

	Distinct Queries/User	Query Refinement	Revenue/User	Any Clicks	Satisfaction	Time to Click (increase in ms)
50m s	-	-	-	-	-	-
200m s	-	-	-	-0.3%	-0.4%	500
500m s	-	-0.6%	-1.2%	-1.0%	-0.9%	1200
1000m s	-0.7%	-0.9%	-2.8%	-1.9%	-1.6%	1900
2000m s	-1.8%	-2.1%	-4.3%	-4.4%	-3.8%	3100

Ein Beispiel guter Performance

Einführung | „Bucket Test“ – Beispiel „Mozilla“

- Fallstudie „Mozilla“
 - Erreichten eine Verbesserung der Ladezeit ihrer Seite um 2,2 sek
 - Ergebnisse von „Bucket Test“ zeigt eine um 15,4% erhöhte Konversionsrate
 - 60 mil. mehr Downloads im Jahr
- Fallstudie „Shopzilla“
 - Verbesserten ihre Ladegeschwindigkeit von 6sec auf 1,2sec.
 - Festgestellte Verbesserungen :
 - 7% bis 12% höhere Konversionsrate
 - 25% mehr Seitenaufrufe
 - Durch Focus auf Performance bei Neuentwicklung Reduktion der Update-Kosten auf 80\$ von mehreren Tausend \$

Agenda / Kapitel 2 - Contentseitige Optimierung

1

Einführung

2

3

4

Es stehen etliche Methoden zur Verfügung, um die Performance von Seiten des Clients zu verbessern

Contentseitige Optimierung / Allgemein

- Preloading von Ressourcen
- Cookiegrösse reduzieren
- Weiterleitungen
- Parallelisierung vs. dns Lookup
 - Mehr einzigartige hostnames...
 - erhöhen Parallelität: Reduzierung „Response Time“
 - erhöhen DNS lookups: Erhöhung „Response Time“
 - Faustregel: 2-4 einzigartige hostnames

Komponentenübersicht

Contentseitige Optimierung / Überblick

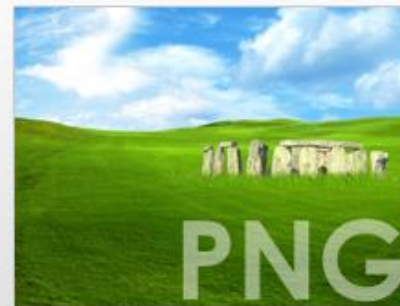
HTML



CSS



JS



Komponentenübersicht

Contentseitige Optimierung / Teil 1 - HTML

HTML



CSS



JS



PNG



HTML optimieren

Contentseitige Optimierung / HTML

- Optionale Tags vermeiden
- Default Attribute nutzen
- Anführungszeichen bei Attribute weglassen
- Dom-Tags reduzieren



Komponentenübersicht

Contentseitige Optimierung / Teil 2 - CSS

HTML



CSS



JS



CSS optimieren

Contentseitige Optimierung / CSS

- CSS nach oben weil..
 - der Browser auf die Stylesheets wartet bevor er rendert
 - Progressives Rendern für die User Experience wichtig ist:
 - Feedback: Seite „hat sich nicht aufgehängt“
 - Progress: Der User sieht wie weit die Seite geladen wurde
 - Der User hat etwas worauf er schauen kann
- <link..> statt @import
- Expressions vermeiden



Komponentenübersicht

Contentseitige Optimierung / Teil 3 - JS

HTML



CSS



JS



JavaScript optimieren

Contentseitige Optimierung / JS

- Smarte Event Handler nutzen
- Duplikate vermeiden
- Skripte ans Ende des HTML-Dokuments
 - Alles unter dem Skript wird erst gerendert, wenn das Skript fertig ist
 - Komponenten unter dem Skript werden erst geladen, wenn Skript fertig ist



Beispiel: JavaScript in der Mitte eines HTML-Dokuments...

Contentseitige Optimierung / JS

Rule 6 - Put Scripts at the Bottom

Example 1 - Scripts in the Middle

This page contains a `<SCRIPT SRC="...">` tag in the middle of the page that takes 10 seconds to load. This illustrates two problems.

PROBLEM 1: Everything below the script won't render until the script is loaded.

The `<SCRIPT SRC="...">` tag is here.

...und 10 Sekunden später

Contentseitige Optimierung / JS

Rule 6 - Put Scripts at the Bottom

Example 1 - Scripts in the Middle

This page contains a `<SCRIPT SRC="...">` tag in the middle of the page that takes 10 seconds to load. This illustrates two problems.

PROBLEM 1: Everything below the script won't render until the script is loaded.

The `<SCRIPT SRC="...">` tag is here.

PROBLEM 2: All components below the script don't start downloading until the script is done.

The script blocks these images from being downloaded, so they're not available until much later during rendering.



It's even worse if scripts are in the document HEAD. Look at [Example 2 - Scripts Top vs. Bottom](#) to see a comparison of scripts at the top vs. the bottom of the page.



This is the companion website for [High Performance Web Sites](#) by Steve Souders.

Komponentenübersicht

Contentseitige Optimierung / Teil 4 - Skripte

HTML



CSS



JS



Skripte optimieren

Contentseitige Optimierung / Skripte

- Externalisieren vs. Inlining
 - Inline-Skripte sind schneller
 - Externe Skripte können gecached werden
- Skripte bündeln
 - Skripte zusammenfassen soweit es Sinn macht
 - Wiederverwendbar
 - Reduzierung der HTTP-Requests



Minifying

Contentseitige Optimierung / Skripte

- Entfernung von Leerzeichen, Kommentaren und weiteren optionalem Text
- große Scripts bis zu 30% schneller

The screenshot shows a side-by-side comparison of two CSS files in a code editor. The left pane, titled 'styles.css', displays the minified version of the CSS, which is a single block of text with all spaces, line breaks, and comments removed. The right pane, titled 'styles uncompressed.css', displays the original, uncompressed version of the CSS, which is formatted with spaces, line breaks, and comments for readability. The minified version is significantly shorter and more compact than the uncompressed version.

```
1 *{margin:0;padding:0}body{background-color:white;font:12px/1.55
2 Arial,Helvetica,sans-serif;color:#2f2f2f;text-align:center}img{border:0;vertical-align
3 :top}a{color:#1e7ec8;text-decoration:underline}a:hover{text-decoration:none}a:focus{
4 outline:0}h1,h2,h3,h4,h5,h6{margin:0 0 5px;line-height:1.35;color:#0a263e}h1{font-size:
5 20px;font-weight:normal}h2{font-size:18px;font-weight:normal}h3{font-size:16px;
6 font-weight:bold}h4{font-size:14px;font-weight:bold}h5{font-size:12px;font-weight:bold}
7 h6{font-size:11px;font-weight:bold}form{display:inline}fieldset{border:0}legend{display:
8 none}table{border:0;border-spacing:0;empty-cells:show;font-size:100%}caption,th,td{
9 vertical-align:top;text-align:left;font-weight:normal}p{margin:0 0 10px}strong{
10 font-weight:bold}address{font-style:normal;line-height:1.35}cite{font-style:normal}q,
11 blockquote{quotes:none}q:before,q:after{content:''}small,big{font-size:1em}ul,ol{
12 list-style:none}.hidden{display:block!important;border:0!important;margin:0!important;
13 padding:0!important;font-size:0!important;line-height:0!important;width:0!important;
14 height:0!important;overflow:hidden!important}.clear{clear:both}.hide{display:none}.
15 nobr{white-space:nowrap!important}.wrap{white-space:normal!important}.a-left{
16 text-align:left!important}.a-center{text-align:center!important}.a-right{text-align:
17 right!important}.v-top{vertical-align:top}.v-middle{vertical-align:middle}.f-left,.
18 left{float:left!important}.f-right,.right{float:right!important}.f-none{float:none!
19 important}.f-fix{float:left;width:100%}.no-display{display:none}.no-margin{margin:0!
20 important}.no-padding{padding:0!important}.no-bg{background:none!important}.wrapper{
21 min-width:1000px}.page-print{background:#fff;padding:25px 30px;text-align:left}.
22 page-empty{background:#fff;padding:20px;text-align:left}.page-popup{background:#fff;
23 padding:25px 30px;text-align:left}.main{width:980px;margin:0 auto;background-color:
24 white;min-height:400px;padding:10px 10px 10px;text-align:left}.col-left{float:left;
25 width:195px;padding:0 0 1px}.col-main{float:left;width:770px;padding:0 0 1px}.
26 col-right{float:right;width:195px;padding:0 0 1px}.col1-layout.col-main{float:none;
27 width:auto}.col2-left-layout.col-main{float:right}.col3-layout.col-main{width:475px;
28 margin-left:17px}.col3-layout.col-wrapper{float:left;width:687px}.col3-layout.
29 col-wrapper.col-main{float:right}.col2-set.col-1{float:left;width:48.5%}.col2-set.
30 col-2{float:right;width:48.5%}.col2-set.col-narrow{width:32%}.col2-set.col-wide{width
31 :65%}.col3-set.col-1{float:left;width:32%}.col3-set.col-2{float:left;width:32%}.
32 margin-left:2%}.col3-set.col-3{float:right;width:32%}.col4-set.col-1{float:left;width
33 :23.5%}.col4-set.col-2{float:left;width:23.5%;margin:0 2%}.col4-set.col-3{float:left;
34 width:23.5%}.col4-set.col-4{float:right;width:23.5%}input,select,textarea,button{font:
35 12px/15px Arial,Helvetica,sans-serif;vertical-align:middle;color:#2f2f2f}input,
36 input-text,select,textarea{background:#fff;border:1px solid #b6b6b6}input,input-text,
37 textarea{padding:2px}select{padding:1px}select option{padding-right:10px}select.multiselect
38 option{border-bottom:1px solid #b6b6b6;padding:2px 5px}select.multiselect option:
39 last-child{border-bottom:0}textarea{overflow:auto}input.radio{margin-right:3px}input,
40 checkbox{margin-right:10px}input.qty{width:2.5em!important;text-align:center}button,
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
```



Komponentenübersicht

Contentseitige Optimierung / Teil 5 - Bilder

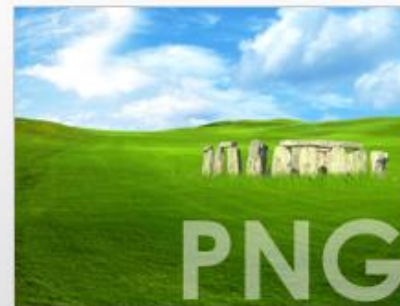
HTML



CSS



JS



Bilder optimieren

Contentseitige Optimierung / Skripte

- Richtiges Dateiformat wählen
- Nicht relevante Informationen entfernen:
 - Meta-Daten des Bildbearbeitungsprogramms
 - Thumbnails
- Optimierungssoftware wie „GifSicle“ nutzen
- Wiederverwendung von Bildern
- Bilder nicht skalieren

Inline Images / Data URL


Contentseitige Optimierung / Skripte

- Bilddaten werden direkt HTML angegeben
 - Keine zusätzlichen HTTP-Requests
 - Können nicht gecached werden

Contentseitige Optimierung / Skripte

Web **Images** Videos Shopping Maps More Search tools

Related searches: [hundebabys](#) [katzenbilder](#) [süße katzen](#) [hunde](#)



src="data:image/jpeg; Base64/9j...rw7iU2Iv0SSS FWXXeiSSS6MP/9k="

```
<div id="center_col">
  <div data-jibp="h" data-jiis="uc" id="taw">_</div>
  <div class="med" id="res" role="main">
    <div data-jibp="h" data-jiis="uc" id="topstuff">_</div>
    <div data-jibp="h" data-jiis="uc" id="search">
      <style>_</style>
      <!--a-->
      <h2 class="hd">Search Results</h2>
      <div data-async-context="query:katzenbabys" id="ires">
        <ol eid="4IiVU9wVH6es7QbN-oCgAg" id="rso">
          <li>
            <div id="rg" style="width: 1651px; height: 11016px;">
              <div id="rg_s" style="visibility: visible;">
                <!--m-->
                <div class="rg_di" data-ved="0CDEQmyGAMAA" data-ri="0" style="width: 239px; height:
                  <a href="http://www.google.de/imgres?imgurl=http%3A%2F%2Fwww.wunderweib.de%2Fmed_A
                    <img data-sz="f" name="ZK2_xwH2jBXXrM:" onload="google.stb.csi.onTbn(1, this)" cl
                      <div class="_i1 rg_ilm" data-ved="0CDMQ9wswAA">_</div>
                    </a>
                  <div class="rg_meta">_</div>
                </div>
              </li>
            </ol>
          </div>
        </div>
      </div>
    </div>
  </div>
```

Agenda / Kapitel 3 - Serverseitige Optimierung

1

2 Contentseitige Optimierung

3

4



Themenübersicht

Serverseitige Optimierung - Überblick

Content Delivery
Network

Expires
Header

Komprimierungs-
möglichkeiten

Ausblick

Definition des Content Delivery Networks

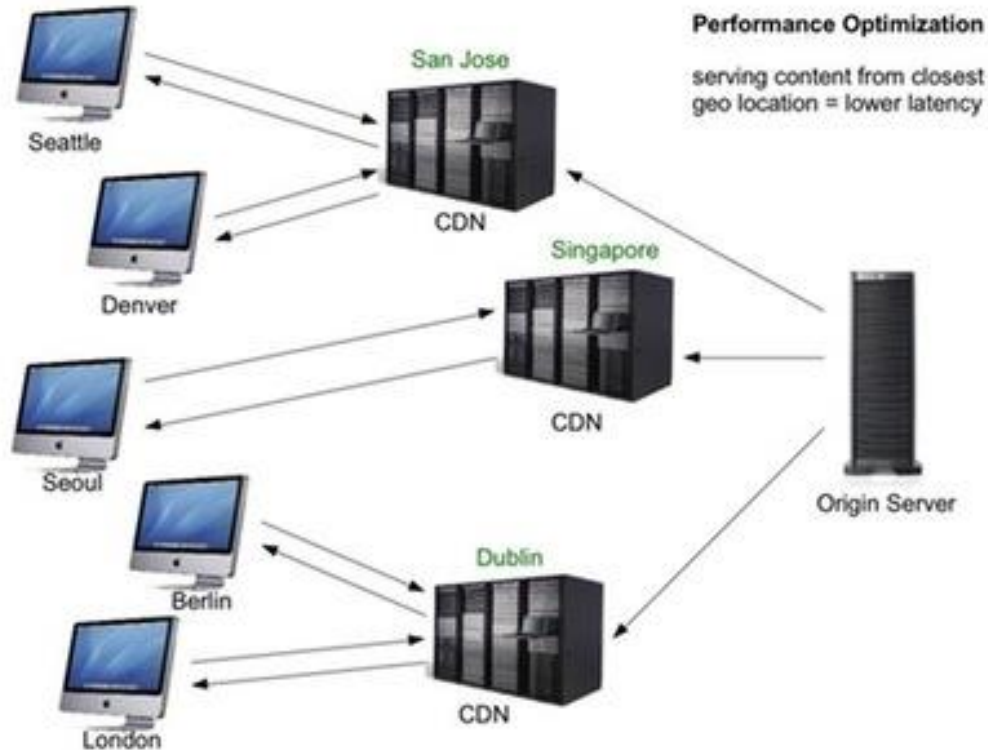
Serverseitige Optimierung / CDN - Definition

„Ein **Content Delivery Network (CDN)**, oder auch **Content Distribution Network** genannt, ist ein Netz lokal verteilter und über das Internet verbundener Server, mit dem Inhalte (insbesondere große Mediendateien) ausgeliefert werden.“

- *Wikipedia, Content Delivery Network*

Funktionsweise des Content Delivery Networks

Serverseitige Optimierung / CDN - Funktionsweise



- **Ohne CDN:** Ein Server ist für alle Clients zuständig
- **Mit CDN:** Jeder Client hat einen lokal zugehörigen Server

Einsatz:

- Internationale Unternehmen mit Kunden auf der ganzen Welt
- Webseiten mit hohem Anteil an statischen Elementen wie Bilder oder Videos

Verbesserung der Performance im Content Delivery Network

Serverseitige Optimierung / CDN – Performance Verbesserung

„ Only 10-20% of the end user response time is spent downloading the HTML document. The other 80-90% is spent downloading all the components in the page.“

- Steve Shoulders, *High Performance Websites*



Performanceoptimierung um bis zu 50%

Rule 2 - Use a Content Delivery Network

Example 2 - No CDN

page load time: **809 ms** [try it again](#)

Rule 2 - Use a Content Delivery Network

Example 1 - CDN

page load time: **484 ms** [try it again](#)

Expires Headers

Serverseitige Optimierung / Expires Headers - Beschreibung

- Optionaler Teil des HTTP-Headers
- Legt Dauer des Cachings fest
- Alternativen (erst ab HTTP/1.1)
 - Cache-Control

Antwort-Header

[Quelltext anzeigen](#)

```
Accept-Ranges bytes
Cache-Control max-age=2592000
Connection keep-alive
Content-Length 14054
Content-Type image/png
Date Fri, 13 Dec 2013 22:32:40 GMT
Etag "2066c2b-36e6-4ece6ea37c7c0"
Expires Sun, 12 Jan 2014 22:32:32 GMT
Last-Modified Sat, 07 Dec 2013 00:33:43 GMT
```



Expires Headers

Serverseitige Optimierung / Expires Headers - Nutzen

- „Empty Cache“
 - Es liegen keine Komponenten der Website im Cache vor
- „Primed Cache“
 - Alle Komponenten der Website liegen im Cache vor
- Anteil Seitenaufrufe mit „Primed Cache“: 75-85%
- Reduzierung der Antwortzeit um mehr als 50%

Rule 3 - Add an Expires Header

Example 1 - No Expires

Roundtrip response time: **729 ms** [try it again](#)

Rule 3 - Add an Expires Header

Example 2 - Far Future Expires

Roundtrip response time: **292 ms** [try it again](#)



Möglichkeiten zur Komprimierung

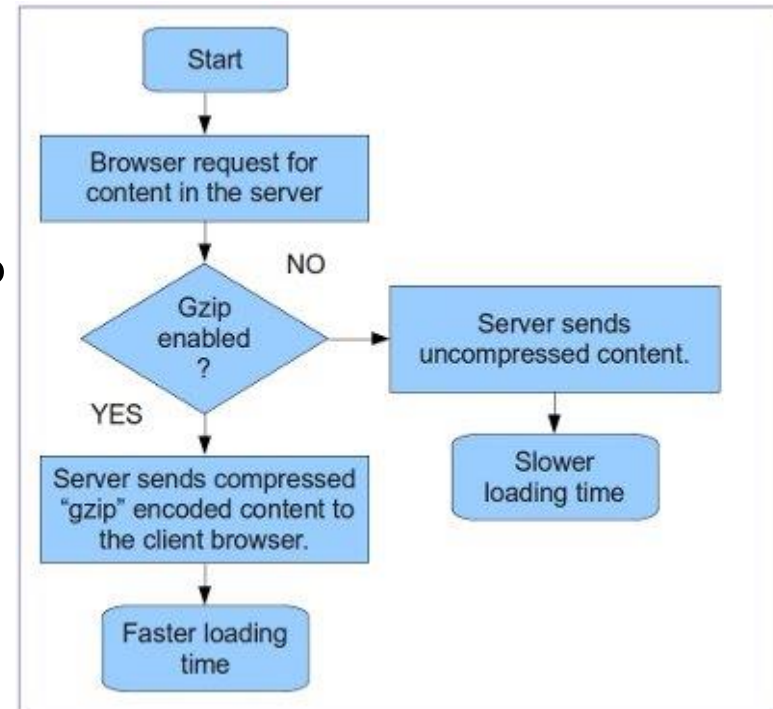
Serverseitige Optimierung / Komprimierungsmöglichkeiten

- Einfache, jedoch effektive Möglichkeit zur Performanceoptimierung
 - Keine Veränderung des Codes
 - Von fast allen Browsern unterstützt
- Reduzierung der Dateigröße im Durchschnitt um 70%
- „Gzip“ als Standard
 - Herausforderungen
 - Beide Seiten müssen „Gzip“ unterstützen
 - Umgang mit hohem Server Load

Möglichkeiten zur Komprimierung

Serverseitige Optimierung / Komprimierungsmöglichkeiten - Einsatz

- Was sollte komprimiert werden?
 - Text basierte Dateien
 - HTML, CSS, Java-Script, XML
 - genug Serverkapazität
- Was sollte nicht komprimiert werden?
 - Standardmäßig komprimierte Dateien
 - Bilder, PDF
 - Server überlastet
- Lohnt sich die Komprimierung?
 - Standardregel: Lokale Komprimierung macht Datei 20-30% kleiner



Ausblick

Serverseitige Optimierung / Ausblick

Datenbank-
optimierungen

ETags

GET für AJAX-
Requests

Flush the
Buffer early

Agenda / Kapitel 4 - Optimierung für mobile Endgeräte

1

2

3

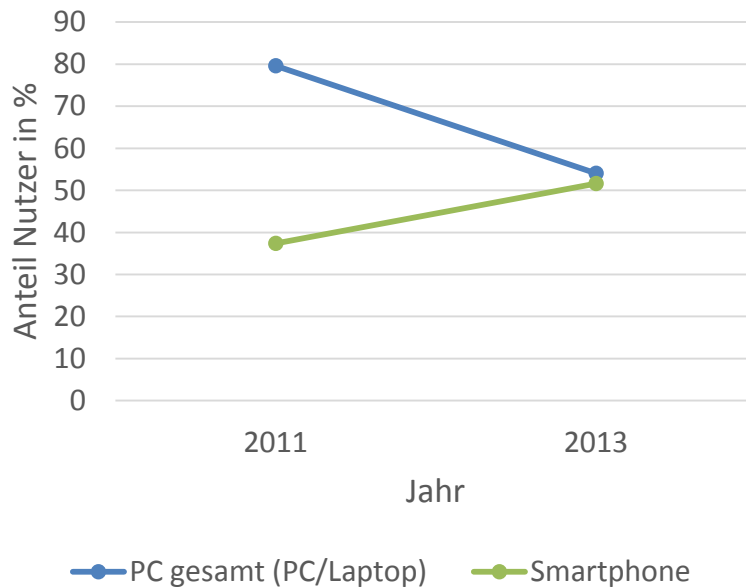
Serverseitige Optimierung

4

Die Wichtigkeit der Bereitstellung von mobile-optimierten Inhalten zeigt sich in den aktuellen Nutzertrends

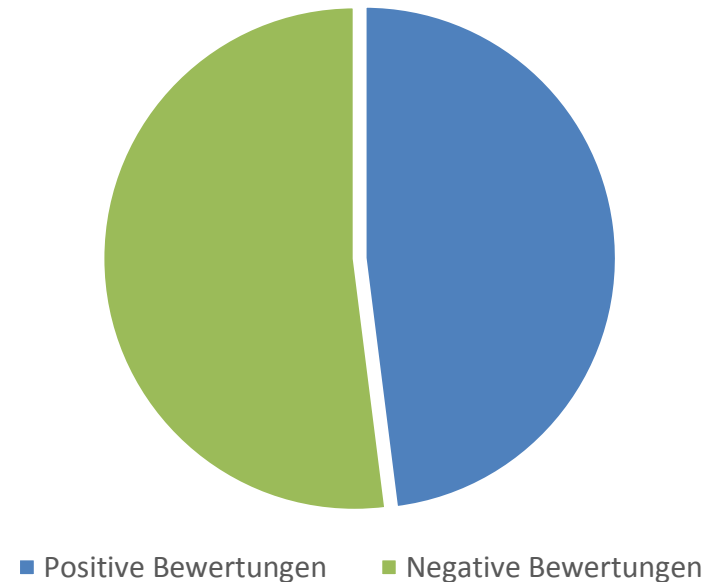
Mobile Optimierung / Relevanz

Tägliche Online-Nutzung nach Endgerät



Quelle: Google (2013), Connected Consumer Study

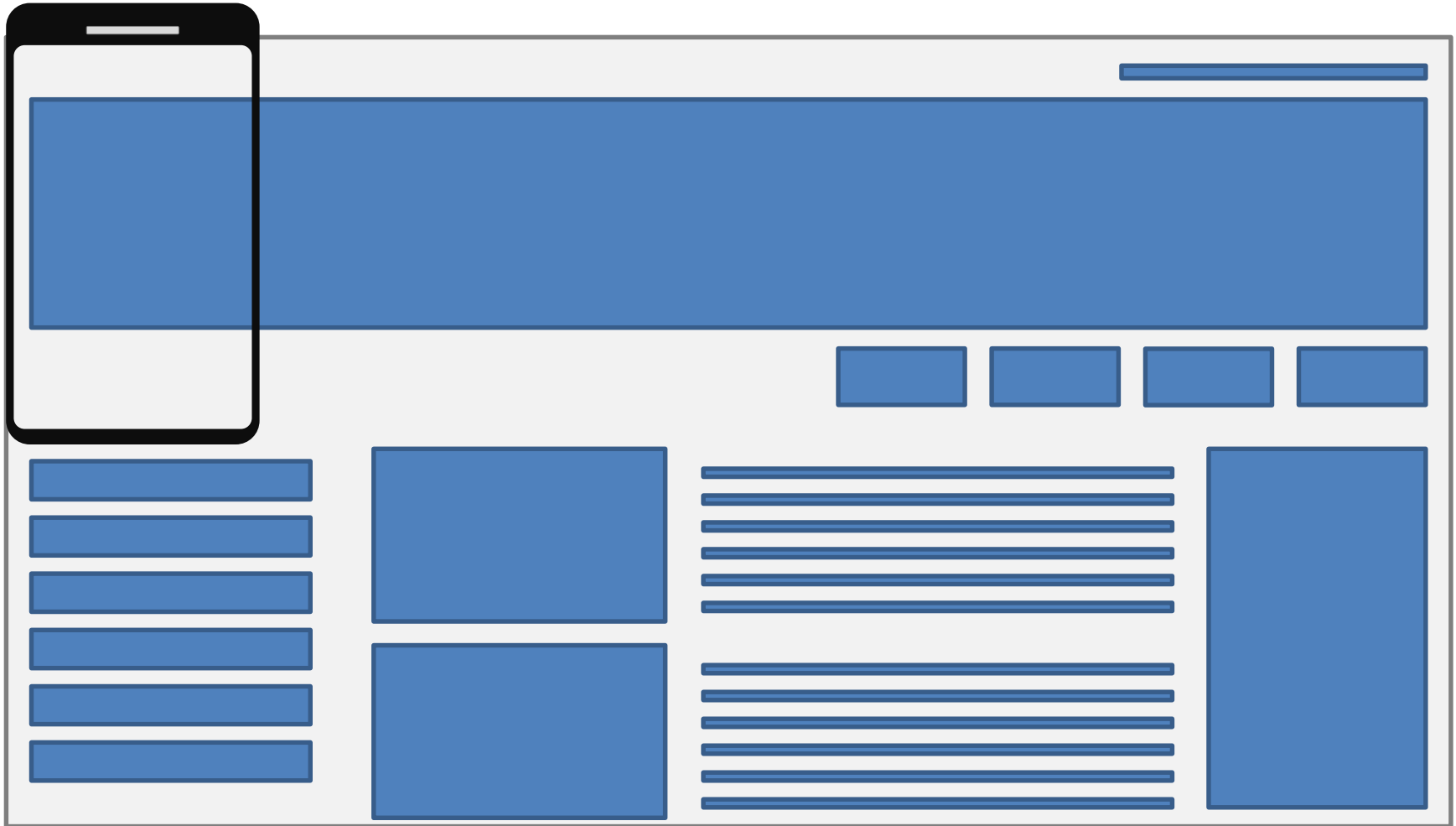
Anteil negativer Bewertungen von Mobile-Websites



Quelle: TNS Infratest (2013)

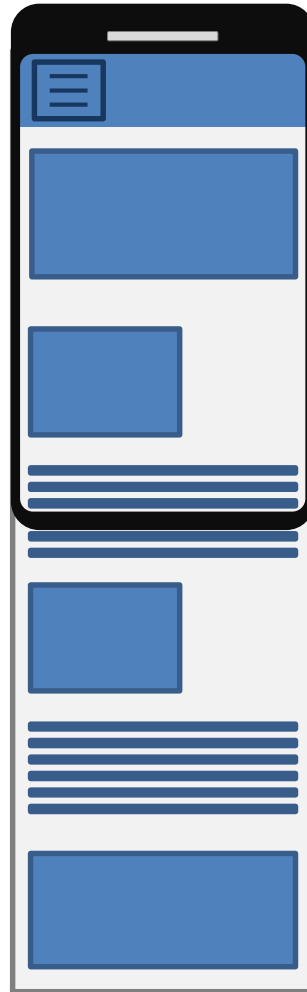
Durch nicht optimierte Websites machen Nutzer mobiler Endgeräte schlechte Erfahrungen beim Surfen

Mobile Optimierung / Negative User Experience



Eine für mobile Endgeräte optimierte Seite zeichnet sich durch ein kompaktes Layout und einfache Bedienung aus

Mobile Optimierung / Positive User Experience

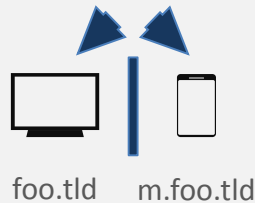


Zur Optimierung einer Website für die mobile Nutzung stehen grundsätzlich zwei Herangehensweisen zur Verfügung

Mobile Optimierung / Optimierungsansätze

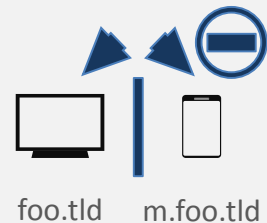
1 Mobile Seite

- Es wird eine zweite, für mobile Endgeräte optimierte Seite entwickelt
- Die mobile Seite ist unter einer eigenen Domian erreichbar
- Grundlage: UserAgent Redirect



1.1 Mobile Seite (reduzierter Inhalt)

- Die mobile Seite erzielt durch weniger Inhalt und Elemente einen Geschwindigkeitsvorteil



2 Responsive Design

- Es wird ein Design für alle Auflösungen erstellt
- Das Design passt sein Layout der Bildschirmgröße des Endgeräts an
- Grundlage: MediaQuerys



2.1 „Mobile-First“-Ansatz

- Beim Erstellen des Responsive Design wird vom mobilen Endgerät als Standard ausgegangen
- Konzentration auf das Wesentliche



Im Vergleich der Optimierungsoptionen hat der „Responsive Design“-Ansatz das größere Potenzial

Mobile Optimierung / Vergleich der Optimierungsansätze

	Vorteile	Nachteile
1 Mobile Seite	<ul style="list-style-type: none"> • Das mobile Design kann vollkommen unabhängig vom Standarddesign entwickelt werden 	<ul style="list-style-type: none"> • Erhöhter Entwicklungs- und Verwaltungsaufwand • Wiedererkennungswert der Seite wird u. U. geschmälert
1.1 Mobile Seite (reduzierter Inhalt)	<ul style="list-style-type: none"> • Weist zusätzlich durch Reduzierung des Inhaltes um große Daten einen (starken) Geschwindigkeitsvorteil vor 	<ul style="list-style-type: none"> • (Wie oben) • User können durch reduzierte Inhalte irritiert bzw. nicht fündig werden
2 Responsive Design	<ul style="list-style-type: none"> • Es muss nur ein Design entwickelt werden • Passt sich jeder Auflösung an 	<ul style="list-style-type: none"> • Erhöhter Konzeptionsaufwand • Befindet sich noch im Aufbau
2.1 „Mobile-First“-Ansatz	<ul style="list-style-type: none"> • Konzentration auf das Wesentliche erhöht die User Experience 	<p>Ø</p>

Vielen Dank für eure Aufmerksamkeit!

Noch Fragen?



Backup / Quellen

- <http://www.elmastudio.de/webdesign/webseiten-optimierung-fur-mobile-gerate-verschiedene-losungen-im-vergleich/>
- <http://www.elmastudio.de/webdesign/webdesign-goes-mobile-first-eine-kleine-einfuehrung-zum-neuen-webdesign-trend/>
- <http://de.slideshare.net/itnig/web-performance-optimization-for-everyone>
- <http://de.slideshare.net/MattKuklinski1/boosting-the-performance-of-your-rails-apps>
- <http://www.websitedesignwebsitedevelopment.com/wp-content/uploads/2012/12/wdwd-css-file.png>
- [http://upload.wikimedia.org/wikipedia/commons/9/99/Gioconda_\(copia_del_Museo_del_Prado_restaurada\).jpg](http://upload.wikimedia.org/wikipedia/commons/9/99/Gioconda_(copia_del_Museo_del_Prado_restaurada).jpg)
- http://connexo.de/img/logos/CSS3_Logo.png
- <http://adaptivemarketingsolutions.com/wp-content/uploads/2013/11/user-experience-adaptive.png>
- http://lghhttp.14677.nexcesscdn.net/806F00/Magento/media/wysiwyg/css_minify.jpg
- http://www.wunderweib.de/media/redaktionell/wunderweib/laura_9/katzewettbewerb/bewerber/babyfinalisten/katzenbaby-finale-01.jpg
- <http://loadstorm.com/wp-content/uploads/files/web-performance-optimization-CDN-400.jpeg>
- <http://cdn.blog-it-solutions.de/wp-content/uploads/2013/12/leverage-browser-caching-expire-header-beispiel.png>
- <http://images.devshed.com/sc/stories/.CodexM/wordpressfastloading1.jpg>
- <https://developer.yahoo.com/performance/rules.html>
- <http://it-ebooks.info/book/258/>

