

Better safe than sorry

Die Macht des Testens im technischen SEO

Hi *SEO*komm!

Chrissy Kunisch

Expert SEO Consultant

Tech SEO ♥

#screamingfrogultra



Hätte auch anders ausgehen können...

ANGEBOTE, PROSPEKTE UND KATALOGE FÜR MÜNCHEN

In **München** wird Dein Shoppingausflug zu einem richtigen Erlebnis, wenn Du durch die Neuhauser- und Kaufingerstraße, ihre Querstraßen und über den Marienplatz läufst. Geschäfte aller Art mit dem unterschiedlichsten Sortiment verwandeln die Altstadt in ein vielseitiges Shoppingviertel. In **München** wirst Du ganz sicher Einiges finden, wofür Du Geld ausgeben kannst. Große Shopping-Center wie das Olympia-Einkaufs-Zentrum, das Mira oder die Riem Arcaden um **Münchens** Zentrum herum überwältigen Dich außerdem mit einem vielfältigen Warenangebot – und der große Vorteil: egal ob es regnet oder schneit, Du bist immer im Trockenen.



Agenda



Warum lohnt es sich selbst zu testen und womit mache ich das?



Best of der letzten 12 Monate: Tests und was man damit tun kann



Fazit inkl. Risiken & Nebenwirkungen für dein SEO-Leben

Warum überhaupt selbst testen?



Die habt ihr auch schon alle öfter gehört

und vielleicht auch einfach mal so hingenommen, oder?

**“Google folgt maximal
5 Redirects”**

**“Content hinter Tabs
wird als weniger
relevant bewertet”**

**“Domain Age ist ein
Rankingfaktor”**

**“Seite sollte in
3 Sekunden laden”**

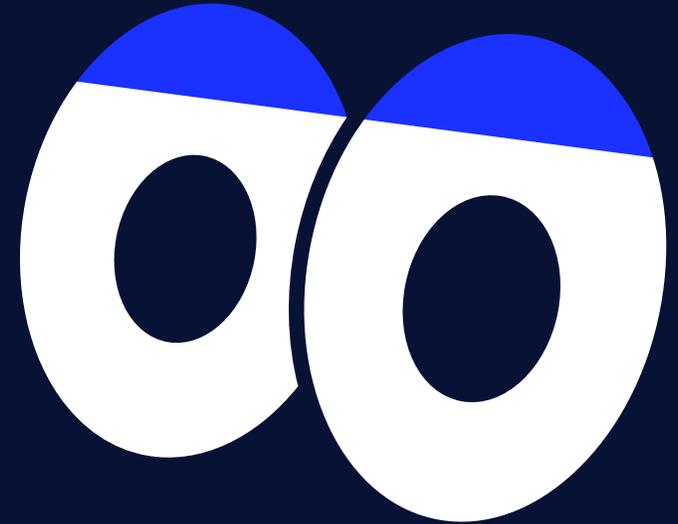
Spoiler:

Es ist oft
leichter die technische
Verarbeitung durch
Google als den Impact
aufs Ranking zu testen.



Was braucht man zum Testen? Wenig.

- Domain
- Webserver
- HTML / CSS / JavaScript Basics
- **PHP Basics** (hier kannst du natürlich auch auf andere Sprachen setzen)
- **Google Search Console registrieren** mit einer Methode deiner Wahl



Ich mache meine Tests konkret mit

- seo-playground.com
- Ubuntu / Apache Webserver
- HTML / CSS / JavaScript Basics
- WordPress Instanz auf einem Verzeichnis (/blog)
- PHP Basics
- Visual Studio Code / WinSCP FTP / Putty
- StackOverflow + ChatGPT

*Viel verrückter
muss es für den
Start auch gar
nicht sein*



*Best of Tests der
letzten 12 Monate*

Tests & Übungen der letzten 12 Monate

Eine Auswahl

Wie scraped man Google
Ergebnisse?

Wie vielen Redirects folgt
Google wirklich aktuell?

Validierung von Monitoring
Tools mit eigenen cURL
Skripten

...

Um wie viel Pixel verschiebt
sich Position 1, wenn ein
SGE-Ergebnis ausgespielt
wird?

Kann ich mit dem Entfernen
Tool das Crawling eines
URL-Patterns triggern?

Was passiert bei einem
dynamisch entfernten /
ergänzten "noindex"?

...

Was passiert, wenn man die
robots.txt in der robots.txt
ausschließt? Disallow: /*.txt\$

CTR steigern durch
abweichende Favicons auf
selber Domain einbinden?

Kann man die PPA Fragen
mittels Parameter-String
wie eine API abfragen?

...

Tests & Übungen der letzten 12 Monate

eine Auswahl

Wie scraped man Google
Ergebnisse?

Wie vielen Re...
Google v...

Validi... Monitoring
cURL

Wie viel Pixel verschiebt
sich Position 1, wenn ein
SGE Ergebnis ausgespielt
wird?

Kann ich mit d...
Tool das Cra...
URL-Pattern

Was passiert, wenn man die
robots.txt in der robots.txt
ausschließt? Disallow: /*.txt\$

CTR steigt
abweichende
selber Domai



**Übrige Antworten
bekommt ihr
gerne an der Bar
heute Abend ;-)**

How it started...

Deine Seite soll in 3 Sekunden laden ...



John Mueller (official) · #MaybeABot

@JohnMu · [Follow](#)

There's no limit per page. Make sure they load fast, for your users. I often check webpagetest.org and aim for <2-3 secs

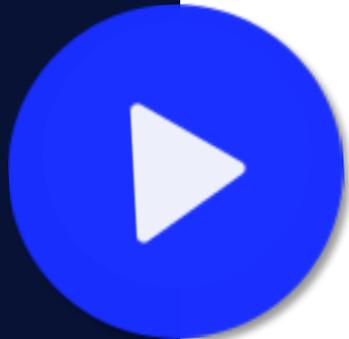


Peeling back the layers:

Stimmen die 3 Sekunden?

Falls ja, welche Speed
Metrik zählt?

3-Sekunden-Regel: Wahrheit oder Fiktion?



YouTube DE

Suchen

SEO Update

de

Wie viel Geduld hat ein Crawler?

So schnell müssen deine Inhalte laden, um indexiert zu werden

Wie viel Geduld hat ein Crawler? "3-Sekunden-Regel" im Page Speed Test • diva-e SEO Update

diva-e 255 Abonnenten Abonnieren

11 Teilen Herunterladen Clip

324 Aufrufe vor 10 Monaten #seo #google #onlinemarketing
Hast du dich schon einmal gefragt, ab wann deine Inhalte so langsam laden, dass sie möglicherweise nicht mehr indexiert werden?
Immer wieder kursiert die Aussage, dass laut Google eine Website in maximal 3 Sekunden laden sollte. Aber ist da wirklich was dran? Abgesehen von einer verbesserten User Experience und den ...mehr

Grundsätzliches Setup

```
<html>

<head>
</head>

<body>

<div id="text-container">
  </div>

<script>
<type="text/javascript"
src="timeout-
test.js"></script>

</body>

</html>
```

```
$(document).ready(function() {

    var textToAdd = "This is the content that
is added repeatedly every 100 ms.";
    var totalTime = 10000; // 10 Sekunden

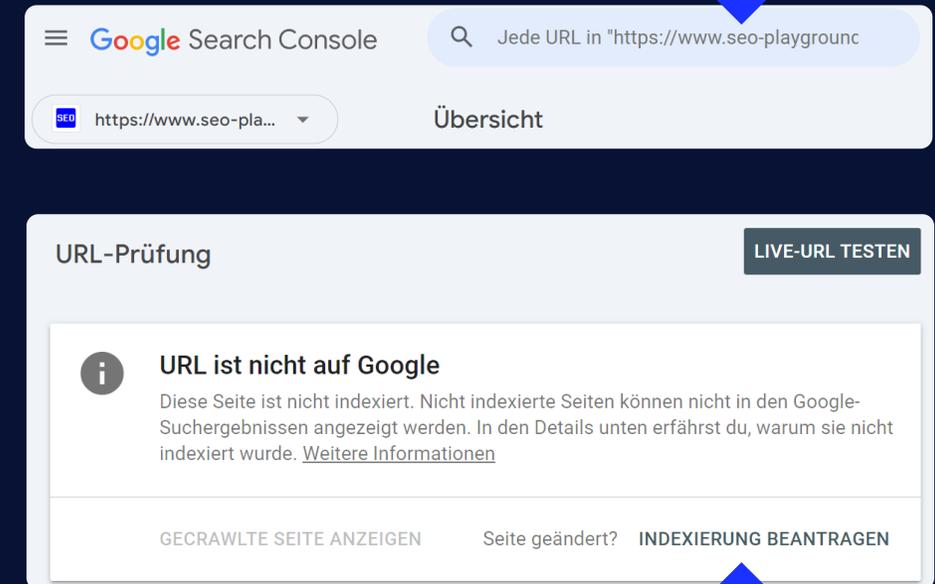
    var startTime = new Date().getTime();
    var currentIndex = 0;

    function insertNewEntry() {
        var currentTime = new Date().getTime();
        var elapsedTime = currentTime -
startTime;

        if (elapsedTime <= totalTime) {
            var newParagraph =
$("<p>").text("Added after " + elapsedTime + "
ms: " + textToAdd);
            $("#text-container").append(newParagraph);
            currentIndex++;

            setTimeout(insertNewEntry, 100);
        }
    }

    // Start
    insertNewEntry();
});
```



The screenshot shows the Google Search Console interface. At the top, there's a search bar with the text "Jede URL in 'https://www.seo-playgrunc'". Below that, the URL "https://www.seo-pla..." is visible. The main section is titled "URL-Prüfung" (URL Check) and contains a message: "URL ist nicht auf Google" (URL is not on Google). The message explains that the page is not indexed and provides a link for "Weitere Informationen" (More information). At the bottom of the message, there are three buttons: "GECRAWLTE SEITE ANZEIGEN" (Show crawled page), "Seite geändert?" (Page changed?), and "INDEXIERUNG BEANTRAGEN" (Request indexing).

*Beispielcodes
zum Download
bekommt ihr*

Wie sieht das im Browser aus?

SEO Playground

"3 SEKUNDEN REGEL": RENDERING TIMEOUT TEST

Alle 100 ms wird Text als Paragraph per JavaScript eingefügt bis 10 Sekunden abgelaufen sind. Wie viele der Einträge werden beim Headless Crawl berücksichtigt und indiziert?

ADDED AFTER 0 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 102 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 208 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 323 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 435 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 540 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 641 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 756 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 857 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 973 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 1074 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 1190 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 1305 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 1407 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 1507 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 1611 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 1721 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 1824 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 1939 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 2054 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 2156 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 2257 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.

RENDERING TIMEOUT TEST

SEO Playground

"3 SEKUNDEN REGEL": RENDERING TIMEOUT TEST FÜR BILDER

Alle 100 ms wird ein Bild per JavaScript eingefügt bis 10 Sekunden abgelaufen sind. Wie viele der Einträge werden beim Headless Crawl berücksichtigt und indiziert?



```
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  

```

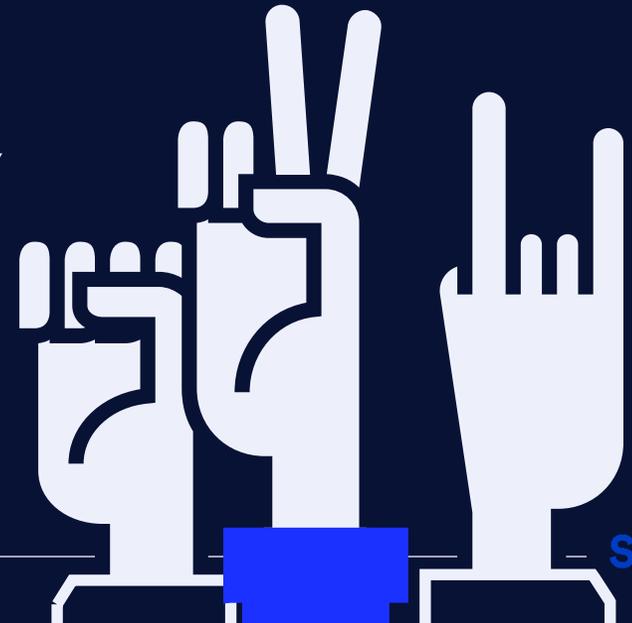
RENDERING TIMEOUT TEST

Sind es wirklich 3 Sekunden?

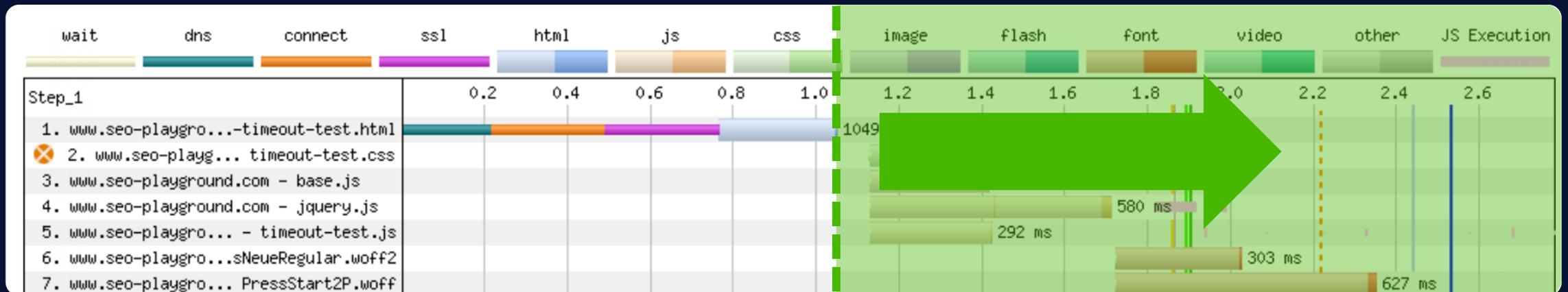
Nein.

Eher so

5 Sekunden!

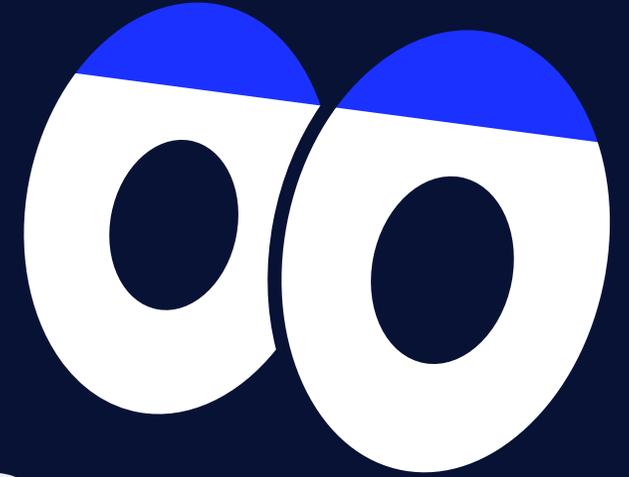


Ab wann zählen die 5 Sekunden?



Das 5-Sekunden-Zeitfenster startet *nach* dem initialen Laden des Index-HTML-Documents.

Welche Metrik zählt jetzt?



Keine explizite Metrik.

Etwa 5 Sekunden nach dem Laden des HTML, werden Inhalte, die innerhalb dieses Zeitfensters geladen werden, wahrscheinlich indexiert; andernfalls eher nicht.

Aufpassen beim Live URL Test

Bei der Nutzung der Google Search Console URL-Inspection solltest du beachten, dass Google deine Seite im Live-URL-Test nicht unbedingt jedes Mal vollständig rendert.

✓ Page indexing Page is indexed

Discovery

Sitemaps No referring sitemaps detected

Referring page None detected

URL might be known from other sources that are currently not reported

Crawl

Last crawl Sep 13, 2023, 7:39:28 AM

Crawled as Googlebot smartphone

Crawl allowed? Yes

Page fetch Successful

Indexing allowed? Yes

Indexing

User-declared canonical None

Google-selected canonical Inspected URL

LEARN MORE OPEN REPORT

GOOGLE-INDEX LIVE-TEST

VS.

✓ Page availability Page can be indexed

URL will be indexed only if [certain conditions](#) are met

Discovery

Not checked in live tests

Crawl

Time Nov 16, 2023, 7:50:37 PM

Crawled as Google Inspection Tool smartphone

Crawl allowed? Yes

Page fetch Successful

Indexing allowed? Yes

Indexing

User-declared canonical None

Google-selected canonical Only determined after indexing

LEARN MORE

GOOGLE-INDEX LIVE-TEST

Am besten verlässt du dich auf die "GOOGLE-INDEX"-Ergebnisse!

Friendly Reminder:

JavaScript ausgeführt
on Document Ready -
keine User Interaction.



Google klickt nicht.

Wie sieht es denn mit Time to First Byte aus?

Relevanteste ▾



Johan v. Hülsen · 1.

7 Monate ...

Founder & SEO at Wingmen Online Marketing

Danke fürs Testen: Wir SEOs sind leider viel zu Testf...
Und die, die testen, teilen zu wenig davon.
Daher doppelt danke.

Soweit ich weiß hat Google schon mal dokument...
Rendering-Zeit Pseudo-Random ist.

Ich erinnere mich auch an einen Test, der gezeigt hatte, dass der
Googlebot an der Uhr dreht und Zeit schneller läuft.

Was auch noch mal spannend wäre (ich schreib das auch in mein
Test-Backlog): Hat die TTFB / Übertragung des HTML, oder
Ankunft des JavaScript einen Impact auf die Anzahl der
gerenderten Zeilen?

Zum Live-Rendering: Einen großen Unterschied macht es, ob
wirklich alle Ressourcen geladen werden. Der Live-Request fragt
ja wirklich alles Live ab. Während der Googlebot bei der
Indexierung ja auch auf gecachte Ressourcen zugreifen kann und
daher nur selten das Problem hat, dass Ressourcen (JavaScripte,
CSS, XHR-Requests) fehlen.

Gefällt mir · 6 | Antworten · 1 Antwort



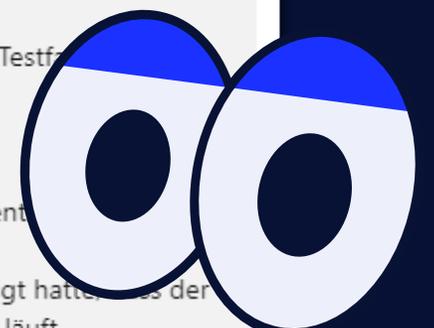
Christiane Kunisch **Autor:in**

7 Monate ...

Expert SEO Consultant bei diva-e

[Johan v. Hülsen](#) danke für dein cooles Feedback und den
zusätzlichen Input. Schreib ich auch mal mit auf die Liste
:-)

Gefällt mir · 4 | Antworten



Einbremsen des initialen HTML load

Methode: PHP Sleep, um das Laden der benötigten Ressourcen zu verzögern

5 Sekunden
delay "TTFB"

alle 100 ms
neuer Text-
paragraph mit
Timestamp

"3 SEKUNDEN REGEL": TTFB RENDERING TIMEOUT TEST

Alle 100 ms wird Text als Paragraph per JavaScript eingefügt bis 10 Sekunden abgelaufen sind. Wie viele der Einträge werden beim Headless Crawl berücksichtigt und indexiert?

NEU: Time to First Byte bzw. initial HTML Download wird um 5 Sekunden verzögert.

ADDED AFTER 1 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 101 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 225 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 327 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 445 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 546 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 646 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 747 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 861 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 1031 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 1144 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 1245 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.

"3 SEKUNDEN REGEL": TTFB RENDERING TIMEOUT TEST FÜR BILDER

Alle 100 ms wird ein Bild per JavaScript eingefügt bis 10 Sekunden abgelaufen sind. Wie viele der Einträge werden beim Headless Crawl berücksichtigt und indexiert?

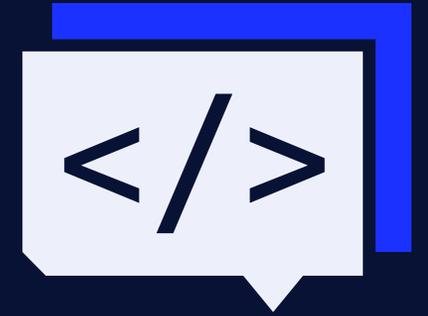
NEU: Time to First Byte bzw. initial HTML Download wird um 5 Sekunden verzögert.



5 Sekunden
delay "TTFB"

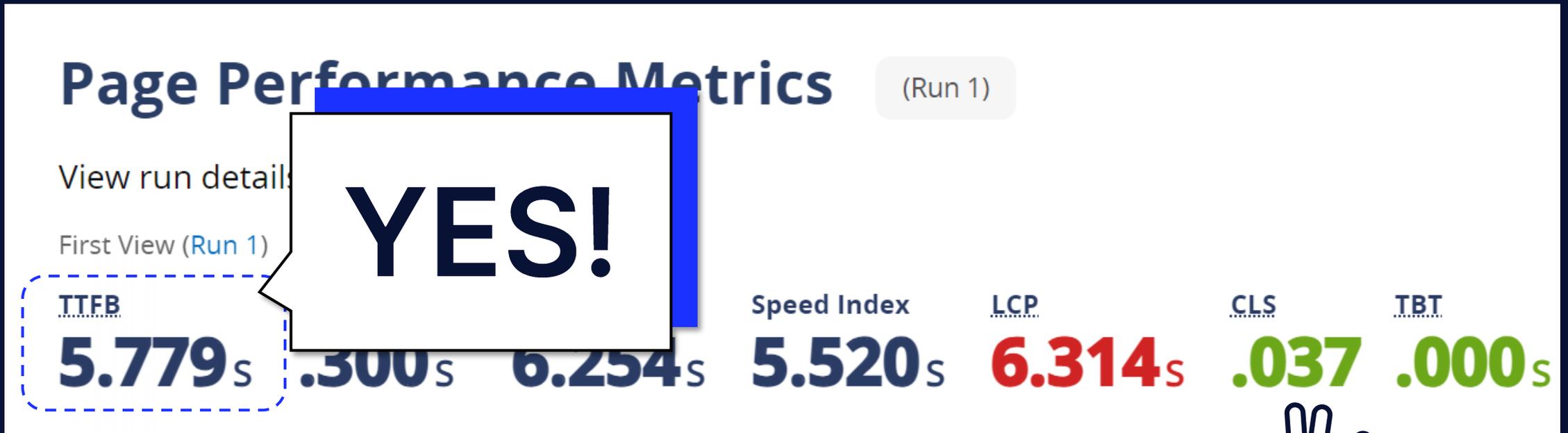
alle 100 ms
neues Bild mit
Timestamp im
Alt Text

Code Beispiel



```
1  <?php
2  // 5 Sekunden warten bis irgendwas geladen wird
3  sleep(5);
4  ?>
5
6  <!DOCTYPE html>
7  <html>
8  <head>
9  |   <title>TTFB Rendering Timeout Test</title>
10 |   <meta charset="utf-8">
```

Hat das PHP den gewünschten Effekt?



webpagetest.org



Wenn man "Time to JavaScript" verzögert?

Methode: JS via PHP mit Sleep laden, um Download des notwendigen JavaScripts zu verzögern

5 Sekunden
delay des JS

alle 100 ms
neuer Text-
paragraph
mit Timestamp

"3 SEKUNDEN REGEL": JS DELAY RENDERING TIMEOUT TEST

Alle 100 ms wird Text als Paragraph per JavaScript eingefügt bis 10 Sekunden abgelaufen sind. Wie viele der Einträge werden beim Headless Crawl berücksichtigt und indexiert?

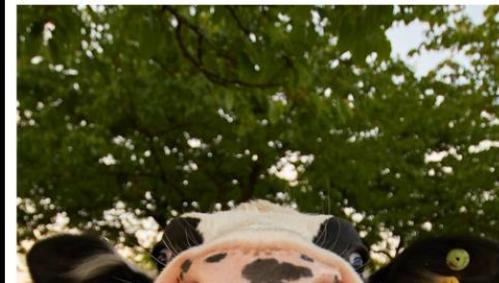
NEU: Zeit bis zum Download des externen JS wird um 5 Sekunden verzögert.

ADDED AFTER 5000 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 5107 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 5210 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 5322 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 5434 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 5536 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 5642 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 5754 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 5871 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 5972 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 6081 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.

"3 SEKUNDEN REGEL": JS DELAY RENDERING TIMEOUT TEST FÜR BILDER

Alle 100 ms wird ein Bild per JavaScript eingefügt bis 10 Sekunden abgelaufen sind. Wie viele der Einträge werden beim Headless Crawl berücksichtigt und indexiert?

NEU: Zeit bis zum Download des externen JS wird um 5 Sekunden verzögert.



5 Sekunden
delay des JS

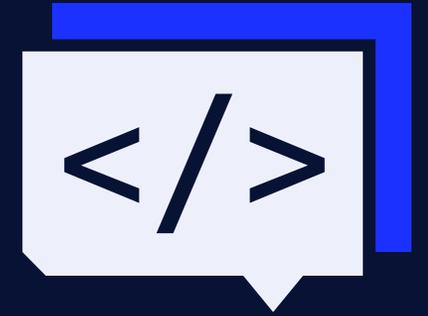
alle 100 ms
neues Bild mit
Timestamp im
Alt Text

Code Beispiel

```
57 <script type="text/javascript" src="base.js"></script>
58 <script type="text/javascript" src="jquery.js"></script>
59 <script type="text/javascript" src="delayed-js.php"></script>
60
61 </body>
62 </html>
```



```
1 <?php
2 header('Content-Type: application/javascript');
3 // 5 Sekunden warten bis irgendwas geladen wird
4 sleep(5);
5 ?>
6
7
8 > $(document).ready(function() { ...
31 });
```



PHP als JavaScript laden?

YEP! Weil der "Type", mit dem du Ressourcen einbindest, sagt "als was soll der Browser das interpretieren" – hier als JavaScript, weil das ja auch das ist, was das PHP "rüberschickt"

Hat das PHP den gewünschten Effekt?

Request #5 Details Request Response Raw Details Object Request Blocking

URL: <https://www.seo-playground.com/tsm/chrissy/speedtest/delayed-js.php>
Loaded By: <https://www.seo-playground.com/tsm/chrissy/speedtest/js-delay-rendering.php>
Document: <https://www.seo-playground.com/tsm/chrissy/speedtest/js-delay-rendering.php>

Request
Host: www.seo-playground.com
IP: 69.163.178.252
Error/Status Code: 200
Priority: High
Protocol: HTTP/2
HTTP/2 Stream: 9, weight 220, depends on 7, EXCLUSIVE
Request ID: 52429.5
Render Blocking Status: in_body

Timing
Discovered: 1.572 s
Request Start: 1.577 s
Time to First Byte: 5263 ms
Content Download: 1 ms

Timing

Phase	Duration
Resource Scheduling	DURATION
Queueing	37.42 ms
Connection Start	DURATION
Stalled	23.72 ms
Request/Response	DURATION
Request sent	0.21 ms
Waiting for server response	5.16 s
Content Download	0.82 ms

Chrome Dev Tools

YES!

YES!



webpagetest.org

Ändert sich hier irgendwas beim Rendern?

Rendering
Timeout bei
"Paragraph added
after
5.398 ms"

*Trotzdem
5 Sekunden!*

Rendering
Timeout bei
"Image added
after
5.398 ms"

```
Getestete Seite
HTML
SCRIPT
...
</p></div><b>
52 </b></div><b>
53 </b></div><b>
54 </b></div><b>
55 </b></div><b>
56 <script type="text/javascript" src="base.js"></script>
57 <script type="text/javascript" src="jquery.js"></script>
58 <script type="text/javascript" src="timeout-test.js"></script>
59
60
61 </b></body></html>
62
```

Gehen sogar 10 Sekunden Verzögerung?

Ist Google sogar so geduldig?

10 Sekunden
delay "TTFB"

alle 100 ms
neuer Text-
paragraph
mit Timestamp

SEO Playground

"3 SEKUNDEN REGEL": TTFB RENDERING TIMEOUT TEST

Alle 100 ms wird Text als Paragraph per JavaScript eingefügt bis 10 Sekunden abgelaufen sind. Wie viele der Einträge werden beim Headless Crawl berücksichtigt und indiziert?

NEU: Time to First Byte bzw. initial HTML Download wird um 10 Sekunden verzögert.

ADDED AFTER 0 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 114 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 243 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 345 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 448 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 549 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 656 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 765 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 868 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 984 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 1098 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 1215 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 1316 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.

TTFB RENDERING TIMEOUT TEST

SEO Playground

"3 SEKUNDEN REGEL": JS DELAY RENDERING TIMEOUT TEST

Alle 100 ms wird Text als Paragraph per JavaScript eingefügt bis 10 Sekunden abgelaufen sind. Wie viele der Einträge werden beim Headless Crawl berücksichtigt und indiziert?

NEU: Zeit bis zum Download des externen JS wird um 10 Sekunden verzögert.

ADDED AFTER 10000 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 10101 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 10201 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 10311 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 10426 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 10527 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 10629 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 10744 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 10845 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 10959 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 11061 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 11162 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 11263 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 11375 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.

JS DELAY RENDERING TIMEOUT TEST

10 Sekunden
delay des JS

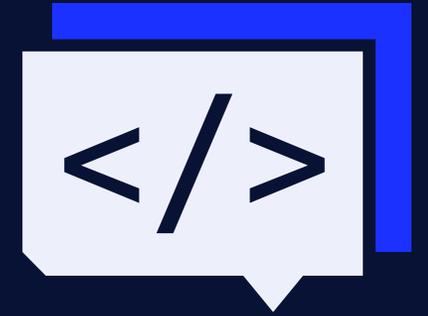
alle 100 ms
neuer Text-
paragraph
mit Timestamp

Wie kann ich mir eigentlich sicher sein, dass der "Googlebot mit JavaScript" gecrawlt hat und nicht wegen einem statischen Crawl meine JS Inhalte auf der Strecke geblieben sind?



Code Beispiel

Mini-Testskript einbauen, um zu prüfen, ob JS ausgeführt wurde



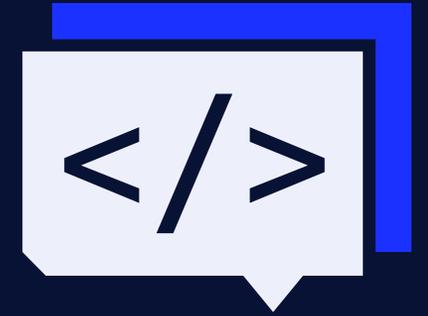
```
57 <script type="text/javascript" src="base.js"></script>  
58 <script type="text/javascript" src="jquery.js"></script>  
59 <script type="text/javascript" src="js-test.js"></script>
```



```
1 $(document).ready(function() {  
2       
3     console.log ("Yes");  
4     $("#text-container").append("JS wurde ausgeführt")  
5 });
```

Code Beispiel

Mini-Testskript einbauen, um zu prüfen, ob JS ausgeführt wurde



SEO Playground

DER PROBATO-NAUTUS IST EIN VIELSEITIGES GERÄT, DAS SOWOHL ALS WERKZEUG ALS AUCH ALS BEGLEITER DIENT. AUSGESTATTET MIT KÜNSTLICHER INTELLIGENZ UND EINER REIHE VON SENSOREN, KANN ES IN UNTERSCHIEDLICHSTEN UMGEBUNGEN EINGESATZT WERDEN - SEI ES IN DER LANDWIRTSCHAFT, IM BAUWESEN, IN DER EXPLORATION ODER Sogar IM ALLTAG EINES JEDEN EINZELNEN. VON DER ANALYSE VON BODENPROBEN BIS HIN ZUR ÜBERWACHUNG DER GESUNDHEIT VON NUTZTIEREN, VON DER KARTIERUNG UNBEKANNTER GEBIETE BIS HIN ZUR UNTERSTÜTZUNG BEI TÄGLICHEN AUFGABEN IM HAUSHALT - DER PROBATO-NAUTUS IST EIN WAHRES MULTITALENT.

DESIGN UND ERGONOMIE

EIN SCHLÜSSELMERKMAL DES PROBATO-NAUTUS IST SEIN DURCHDACHTES DESIGN, DAS SOWOHL FUNKTIONALITÄT ALS AUCH BENUTZERFREUNDLICHKEIT IN DEN VORDERGRUND STELLT. MIT EINEM ERGONOMISCHEN GRIFF UND INTUITIVEN BEDIENELEMENTEN LIEGT DAS GERÄT ANGENEHM IN DER HAND UND ERMÖGLICHT PRÄZISES ARBEITEN ÜBER LÄNGERE ZEITRÄUME HINWEG. SEIN ROBUSTES GEHÄUSE IST SO KONSTRUIERT, DASS ES DEN HERAUSFORDERUNGEN VERSCHIEDENER ARBEITSUMGEBUNGEN STANDHÄLT, SEI ES UNTER EXTREMEN WETTERBEDINGUNGEN IM FREIEN ODER ANSPRUCHSVOLLEN INDUSTRIELLEN ANWENDUNGEN.

INNOVATION UND FORTSCHRITT

DER PROBATO-NAUTUS STEHT EXEMPLARISCH FÜR DIE UERMÜDLICHE SUCHE NACH INNOVATION UND FORTSCHRITT IN DER MENSCHLICHEN GESCHICHTE. MIT SEINER FÄHIGKEIT, KOMPLEXE PROBLEME ZU LÖSEN UND NEUE HORIZONTE ZU ERSCHLIEßEN, HAT ER BEREITS DIE GRENZEN DESSEN, WAS MÖGLICH IST, ERWEITERT UND DIE MENSCHHEIT IN EINE AUFREGENDE ZUKUNFT GEFÜHRT. DOCH TROTZ ALL SEINER LEISTUNGSFÄHIGKEIT BLEIBT DER PROBATO-NAUTUS STETS EIN WERKZEUG IN DEN HÄNDEN SEINER BENUTZER, BEREIT, IHRE TRÄUME UND VISIONEN ZU UNTERSTÜTZEN UND ZU VERWIRKLICHEN.

JS WURDE AUSGEFÜHRT

IMAGE TEST NO IMG

INNOVATION UND FORTSCHRITT

DER PROBATO-NAUTUS STEHT EXEMPLARISCH FÜR DIE UERMÜDLICHE SUCHE NACH INNOVATION UND FORTSCHRITT IN DER MENSCHLICHEN GESCHICHTE. MIT SEINER FÄHIGKEIT, KOMPLEXE PROBLEME ZU LÖSEN UND NEUE HORIZONTE ZU ERSCHLIEßEN, HAT ER BEREITS DIE GRENZEN DESSEN, WAS MÖGLICH IST, ERWEITERT UND DIE MENSCHHEIT IN EINE AUFREGENDE ZUKUNFT GEFÜHRT. DOCH TROTZ ALL SEINER LEISTUNGSFÄHIGKEIT BLEIBT DER PROBATO-NAUTUS STETS EIN WERKZEUG IN DEN HÄNDEN SEINER BENUTZER, BEREIT, IHRE TRÄUME UND VISIONEN ZU UNTERSTÜTZEN UND ZU VERWIRKLICHEN.

JS WURDE AUSGEFÜHRT

Elements Console Sources Network >> 1

top | -url:https://ww | Default levels | No Issues | 1 hidden

Yes [image-test.js:3](#)

IMAGE TEST NO IMG

Learning

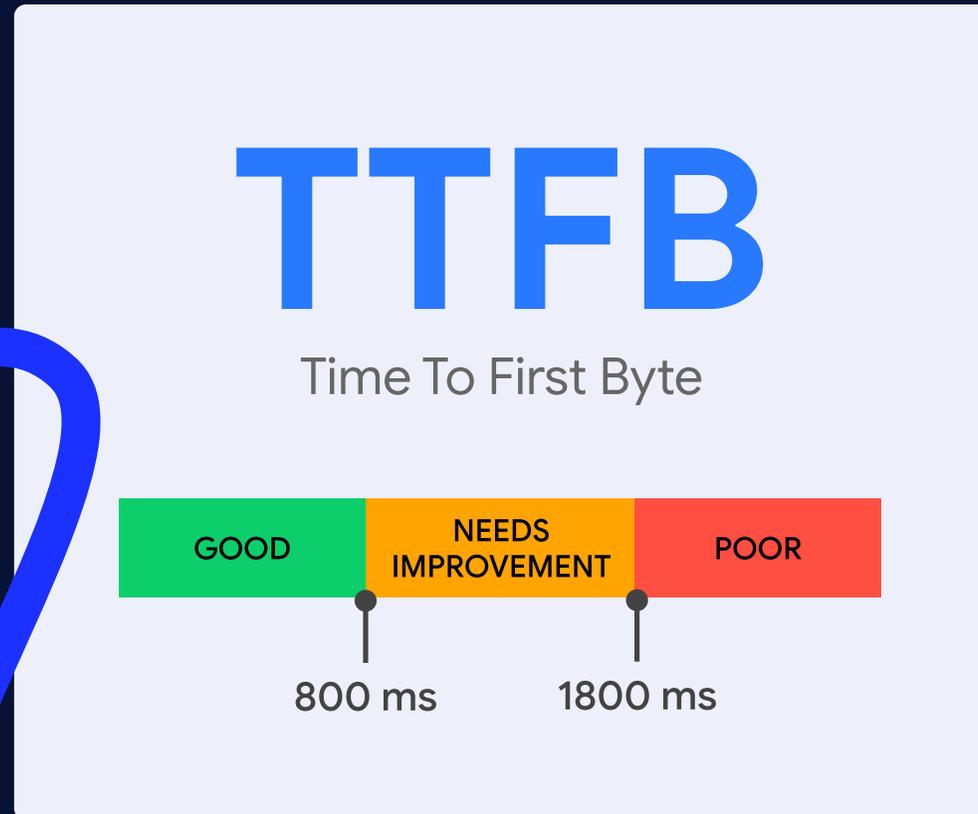
Bei Time to First
Byte gibt es
*“Bonus” von
bis zu 10 Sekunden.*



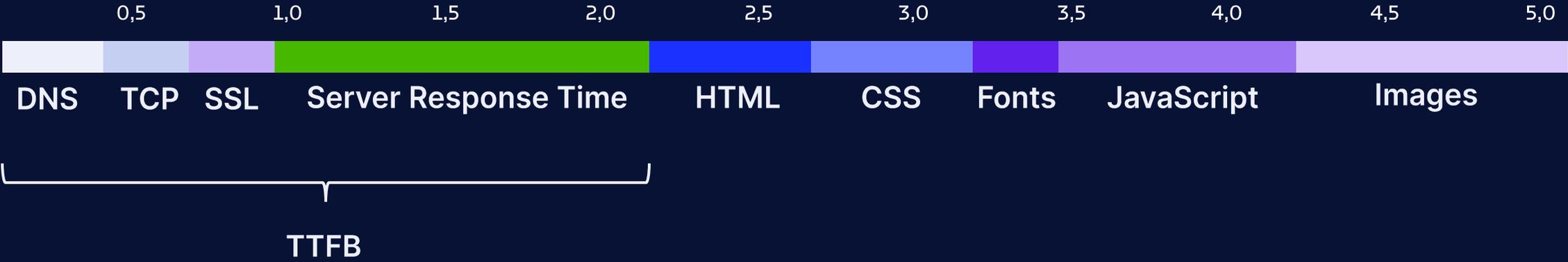
Kleines "ABER"

Die deutlich
langsameren Seiten
haben teilweise 1 bis 3
Wochen gebraucht bis
sie indexiert wurden.

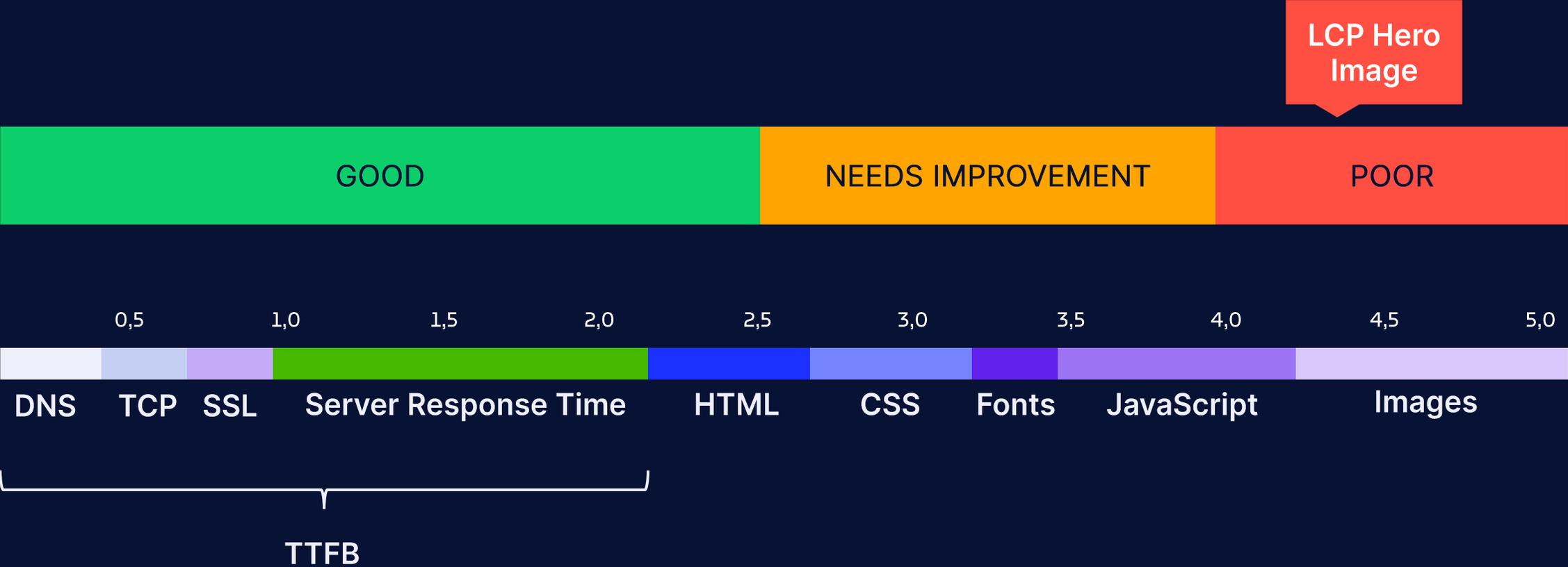
Vielleicht deswegen unambitionierte TTFB Richtwerte von Google?



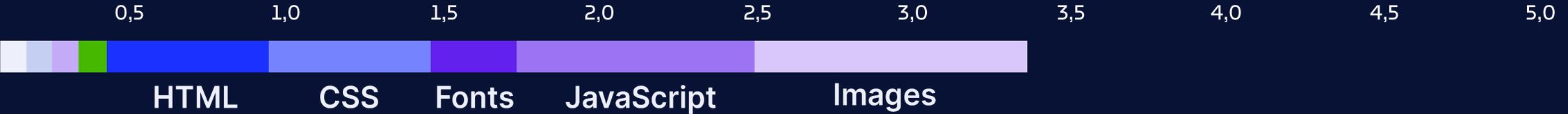
Time to First Byte: It's *magic!*



Time to First Byte: It's magic!



Time to First Byte: It's *magic!*



TTFB

Bei Delays von
JavaScript gibt es
*“Bonus” von
bis zu 5 Sekunden.*



5 Sekunden JS Execution sind manchmal schneller vorbei als du denkst

Beispiel

Google Tag Manager spät laden, an sich sinnvoll aus Page Speed Gründen.

ABER genau deshalb ein Grund mehr SEO-relevante Elemente nicht darüber zu laden (hreflang, Structured Data, ... alles schon gesehen).



Cool. Und jetzt?

Das Ergebnis kann man sich jetzt auf zwei Arten und Weisen zunutze machen

**Der White Hat SEO
geht jetzt auf die
Suche, welche Inhalte
vielleicht zu spät laden
und stellt sicher, dass
alles relevante vor
Ablauf der 5 Sekunden
geladen hat ...**



**Der Gray Hat SEO
geht jetzt auf die
Suche, welche Inhalte
er mit Absicht später
lädt, damit Inhalte
nicht indexiert
werden, die vielleicht
sonst DC wären ...**

Adding semantics:

Beeinflusst vielleicht
Structured Data die
Indexierung über die 5
Sekunden hinaus?

Programmieren
reicht langsam
wieder? Lass dir
von ChatGPT
helfen!



Testseite einfach generieren lassen...

Tipp: Bitte ChatGPT möglichst einfachen Code zu generieren, sonst wird es manchmal wild

ChatGPT 4o ▾

I am a technical SEO expert. I want to setup a landing page that has structured data markup according to schema.org. What I need is a HTML code snippet that I can copy into my existing HTML body tag. This HTML snippet should hold placeholder content that is arranged in a list of made-up technical SEO words. This item list with 15 items should then be marked with JSON-LD markup for the type "ItemList".

Sure, here is an example of an HTML snippet that you can insert into your existing HTML body tag. It includes placeholder content for a list of 15 made-up technical SEO words, marked up with JSON-LD structured data according to schema.org for the type "ItemList".

```
html Code kopieren
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>SEO Technical Terms</title>
  <script type="application/ld+json">
  {
    "@context": "https://schema.org",
    "@type": "ItemList",
    "itemListElement": [
      {
        "@type": "ListItem",
        "position": 1,
        "name": "SEOflux Analysis"
      },
      {
        "@type": "ListItem",
        "position": 2,
        "name": "KeywordMatrix"
      },
      {

```

Liste mit itemList Markup

Methode: dynamisch geladenen Inhalt zusätzlich semantisch mit Structured Data auszeichnen

Microdata

alle 100 ms
neuer Text-
paragraph mit
Timestamp

SEO Playground

"3 SEKUNDEN REGEL": ITEMLIST DELAY RENDERING TIMEOUT TEST

Alle 100 ms wird Text als Paragraph per JavaScript eingefügt bis 10 Sekunden abgelaufen sind. Wie viele der Einträge werden beim Headless Crawl berücksichtigt und indiziert, wenn sie mittels strukturierte Daten ausgezeichnet sind?

NEU: Zeit bis zum Download des externen JS mit Microdata Markup wird um 5 Sekunden verzögert.

ADDED AFTER 5001 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 5104 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 5211 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 5313 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 5427 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 5543 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 5644 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 5746 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 5861 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 5961 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.

ITEMLIST DELAY RENDERING TIMEOUT TEST

SEO Playground

"3 SEKUNDEN REGEL": JSON-LD ITEMLIST RENDERING TIMEOUT TEST

Alle 100 ms wird Text als Paragraph per JavaScript eingefügt bis 10 Sekunden abgelaufen sind. Wie viele der Einträge werden beim Headless Crawl berücksichtigt und indiziert?

NEU: Structured Data für ItemList mit JSON-LD wird über externes Script geladen.

ADDED AFTER 5000 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 5102 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 5203 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 5318 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 5431 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 5546 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 5648 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 5764 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 5880 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 5996 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.
ADDED AFTER 6100 MS: THIS IS THE CONTENT THAT IS ADDED REPEATEDLY EVERY 100 MS.

RENDERING TIMEOUT TEST

JSON-LD

alle 100 ms
neues Bild mit
Timestamp im
Alt Text

Was ist dann passiert?

Rendering
Timeout bei
"List Item added
after
5.400 ms"

Nix!

Rendering
Timeout bei
"List Item added
after
5.400 ms"

```
Ge crawlte Seite
HTML
WEITERE INFORMATIONEN
...
<script type="text/javascript" src="base.js"></script>
<script type="text/javascript" src="jquery.js"></script>
<script type="text/javascript" src="delayed-itemlist-10.php">
</script>
</body></html>
...
<div class="wrapper-nav">
<nav>
<ul>
<li><a href="index.html">Home</a></li>
<li><a href="home/seo-tools.html">SEO Tools</a></li>

```

Level up: Product Markup

JSON-LD
Product Markup

Alle 100ms neuer
Eintrag für
1 Produkt mit
Bild, Preis, etc.

SEO Playground

PRODUCT 1

PRODUCT WAS ADDED AFTER 0 MS



PRICE: \$101

PRODUCT 2

PRODUCT WAS ADDED AFTER 111 MS



PRICE: \$102

PRODUCT 3

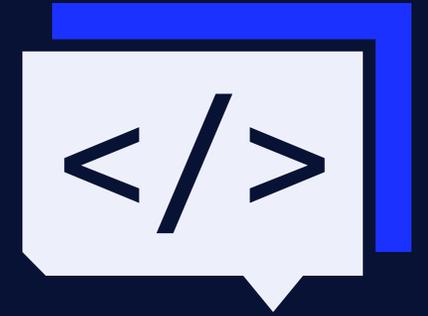
PRODUCT WAS ADDED AFTER 220 MS

RENDERING TIMEOUT TEST

*Vielleicht gibt es
ja nur einen
"Bonus" bei
"relevanterem"
Markup...*

Code Beispiel

Langsam wird es komplexer, aber machbar



```
1 $(document).ready(function() {
2   let productCount = 0;
3   const stopTime = 10000; // 10 seconds
4   const interval = 100; // 100 ms interval for generating products
5   const startTime = new Date().getTime(); // Record the time when the script starts
6
7   function generateProduct() {
8     const currentTime = new Date().getTime();
9     const elapsedTime = currentTime - startTime; // Time
10
11     // Check if 10 seconds have passed
12     if (elapsedTime < stopTime) {
13       productCount++;
14       const productId = `product${productCount}`;
15       const productPrice = (100 + productCount); // Dyn
16       const productImage = `product-image.png`; // Fixe
17
18       // Creating a new product entry
19       const productDiv = $('<div></div>').attr('id', 'product-list');
20       productDiv.html(
21         <h2>Product ${productCount}</h2>
22         <p>Product was added after ${elapsedTime} ms</p>
23         
24         <p>Price: $$${productPrice}</p>
25       );
26
27       // Inserting the JSON-LD product markup dynamically
28       const productMarkup = $('<script></script>')
29         .attr('type', 'application/ld+json')
30         .text(JSON.stringify({
31           "@context": "https://schema.org/",
32           "@type": "Product",
33           "name": `Product ${productCount}`,
34           "description": `Product was added after ${elapsedTime} ms`,
35           "image": productImage,
36           "sku": `SKU${productCount}`,
37           "brand": {
38             "@type": "Brand",
39             "name": "Example Brand"
40           },
41           "offers": {
42             "@type": "Offer",
43             "priceCurrency": "USD",
44             "price": productPrice,
45             "availability": "https://schema.org/InStock"
46           }
47         }));
48
49       // Appending product and JSON-LD to the DOM
50       productDiv.append(productMarkup);
51       $('#product-list').append(productDiv);
52
53       // Continue generating products every 100 ms
54       setTimeout(generateProduct, interval);
55     } else {
56       console.log('Product generation stopped after 10 seconds.');
```



*Nicht vergessen: ChatGPT is your friend here :-)
und Beispielcode gibt's zum Download*

Funktioniert das Skript wie gedacht?

Test für Rich Results



55 gültige Elemente erkannt

Für gültige Elemente sind Rich-Suchergebnisse in der Google Suche möglich.
[Weitere Informationen](#)

GETESTETE SEITE ANZEIGEN

VORSCHAU DER ERGEBNISSE ANZEIGEN

YES!

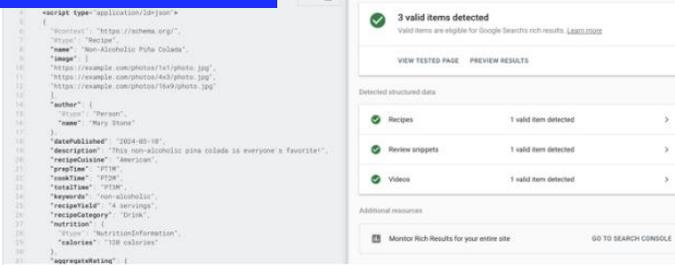


Nice to know: Structured Data Testing Tool Besonderheiten

Strukturierte Daten testen

Wir empfehlen, mit dem [Test für Rich-Suchergebnisse](#) zu beginnen, um zu sehen, welche Rich-Suchergebnisse für deine Seite generiert werden können. Nutze für die allgemeine Schemavalidierung den [Schema Markup Validator](#), um alle Arten von schema.org-Markup ohne Google-spezifische Validierung zu testen.

Test für Rich Results

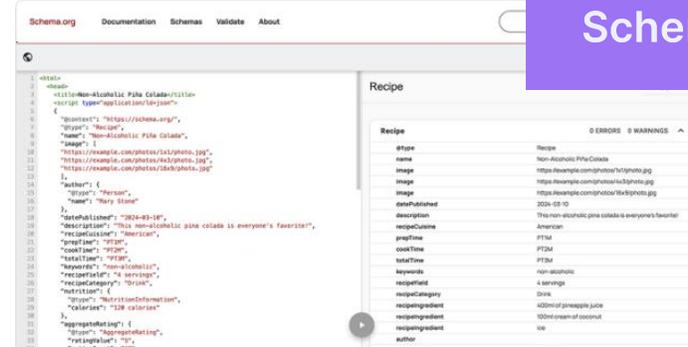


Test für Rich-Suchergebnisse

Das offizielle Tool von Google, mit dem du deine strukturierten Daten testest, um herauszufinden, welche Rich-Suchergebnisse über die [strukturierten Daten](#) auf deiner Seite generiert werden können. Du kannst dir auch als Vorschau ansehen, wie Rich-Suchergebnisse in der Google Suche aussehen können.

[Test für Rich-Suchergebnisse aufrufen](#)

Schema Markup Validator



Schema Markup Validator

Überprüfe alle auf Schema.org basierenden strukturierten Daten, die in Webseiten eingebettet sind, ohne dass Google spezifische Warnungen ausgibt.

[Schema Markup Validator aufrufen](#)

Nice to know: Structured Data Testing Tool Besonderheiten

Test für Rich Results

- Validiert, ob Markup für Rich Results vorhanden und korrekt implementiert ist
- **URL:** inline & externes JS wird geladen
- Markup Test rendert auch ~ 5 Sekunden

Schema Markup Validator

- Validiert, ob Syntax gemäß schema.org korrekt ist ("W3C Validator für Markup")
- **URL abrufen:** es wird kein JS geladen
- **Code Snippet:** gerenderten Code reinkopieren, dann klappt's

Nice to know: Structured Data Testing

<i>Beispiel</i>	Test für Rich Results		Schema Markup Validator	
	URL abrufen	Code Snippet einfügen	URL abrufen	Code Snippet einfügen
ItemList	<i>Keine Elemente erkannt</i>	<i>Keine Elemente erkannt</i>	<i>Keine Elemente erkannt</i>	ItemList All (1)
Product	110 gültige Elemente erkannt Produkt Snippets: 55 gültige Elemente Händlereinträge: 55 gültige Elemente	188 gültige Elemente erkannt Produkt Snippets: 94 gültige Elemente Händlereinträge: 94 gültige Elemente	<i>Keine Elemente erkannt</i>	Product All (94)

Quizfrage



*Wer die Zahlen
direkt versteht,
gewinnt ein
Tech-SEO-Expert
Sternchen.*

188 gültige Elemente
erkannt

Produkt Snippets: 94 gültige Elemente
Händlereinträge: 94 gültige Elemente

Product
All (94)

Upside: Auch aus Fehlern beim Testen kann man neue Erkenntnisse gewinnen!



Learning durch falsch gebautes Skript:

Wenn ein Markup davon abhängt, ob das Skript komplett durchläuft und das Skript nicht in 5 Sekunden komplett geladen wurde, dann wird das nix.

Szenarien, wann dir das passieren könnte:

- Wenn man zu viel über den GTM lädt
- Abfrage aus PIM für z.B. Preis dauert zu lang und davon hängt Product Markup ab
- ...

Wir fassen zusammen:

Structured Data

kann dir auch nicht
aus der Page-Speed-
Patsche helfen.

Don't get too excited about Page Speed:

Over-engineerern der
Performance kann
ernsthaft die Indexierung
von *Bildern* gefährden!

Das Picture Tag Setup, das den Test getriggert hat

Reguläres Picture Tag für verschiedene Source Sets mit erforderlichem Image Tag

```
<picture>

<source media="(min-width: 1200px)"
srcset="https://www.domain.com/media/content/img/image-name-16-
9_width_534_height_300.jpg,
https://www.domain.com/media/content/img/image-name-16-
9_width_800_height_450.jpg 2x">

<source media="(min-width: 960px)"
srcset="https://www.domain.com/media/content/img/image-name-16-
9_width_1200_height_675.jpg,
https://www.domain.com/media/content/img/image-name-16-
9_width_2400_height_1350.jpg 2x">

<source media="(min-width: 768px)"
srcset="https://www.domain.com/media/content/img/image-name-16-
9_width_960_height_540.jpg,
https://www.domain.com/media/content/img/image-name-16-
9_width_1920_height_1080.jpg 2x">

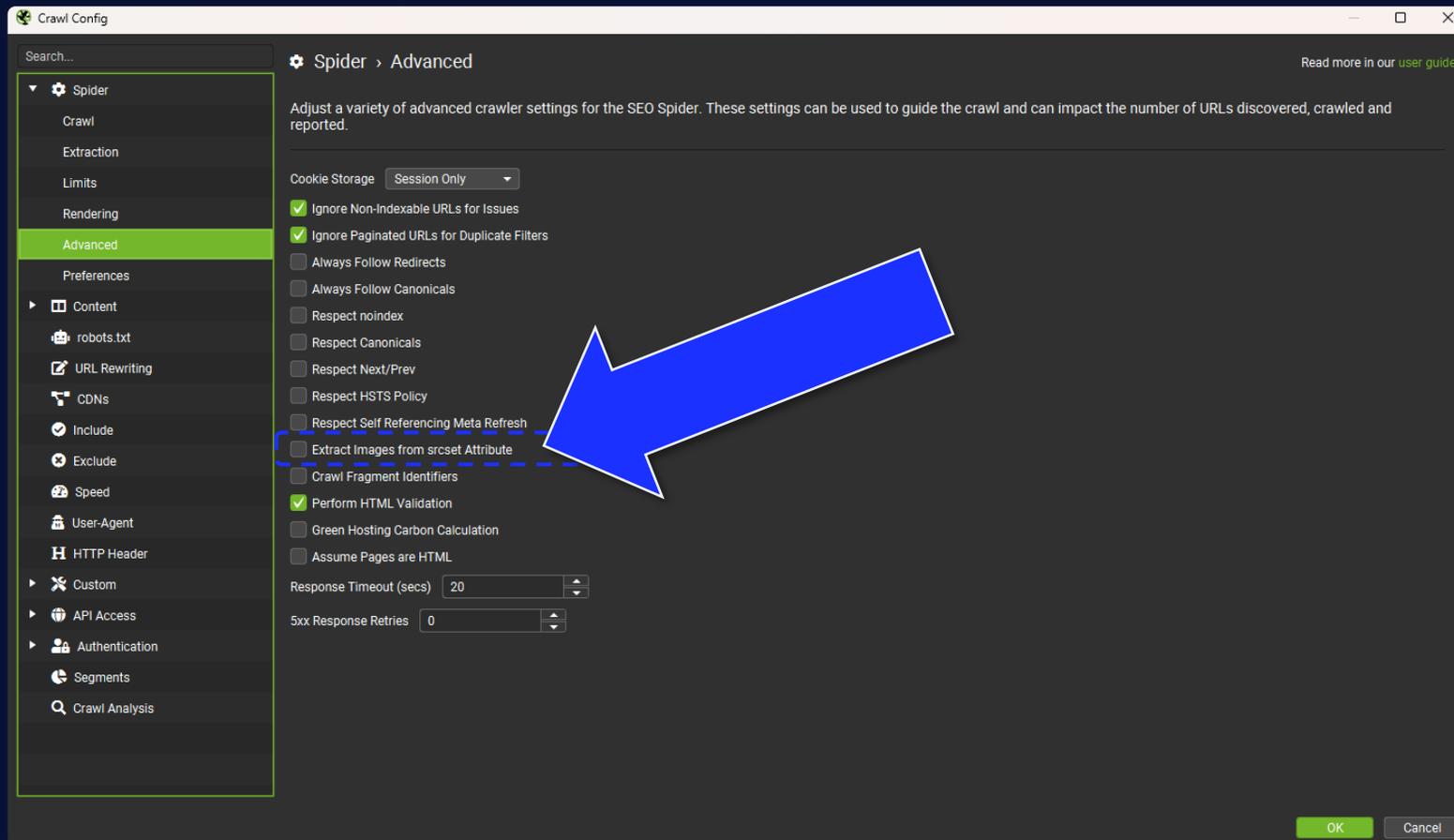
<source media="(min-width: 480px)"
srcset="https://www.domain.com/media/content/img/image-name-16-
9_width_800_height_450.jpg,
https://www.domain.com/media/content/img/image-name-16-
9_width_1536_height_864.jpg 2x">

<source media=""
srcset="https://www.domain.com/media/content/img/image-name-16-
9_width_480_height_270.jpg,
https://www.domain.com/media/content/img/image-name-16-
9_width_960_height_540.jpg 2x">



</picture>
```

PSA: Screaming Frog kann natürlich Source Sets auslesen



*oder Handarbeit per
Custom Extraction*

Ist dir schon was aufgefallen?

Vielleicht hast du dich schon gewundert ...

Reguläres Picture Tag
für verschiedene
Source Sets mit
erforderlichem
Image Tag

```
<picture>

<source media="(min-width: 1200px)"
srcset="https://www.domain.com/media/content/img/image-name-16-
9_width_534_height_300.jpg,
https://www.domain.com/media/content/img/image-name-16-
9_width_800_height_450.jpg 2x">

<source media="(min-width: 960px)"
srcset="https://www.domain.com/media/content/img/image-name-16-
9_width_1200_height_675.jpg,
https://www.domain.com/media/content/img/image-name-16-
9_width_2400_height_1350.jpg 2x">

<source media="(min-width: 768px)"
srcset="https://www.domain.com/media/content/img/image-name-16-
9_width_960_height_540.jpg,
https://www.domain.com/media/content/img/image-name-16-
9_width_1920_height_1080.jpg 2x">

<source media="(min-width: 480px)"
srcset="https://www.domain.com/media/content/img/image-name-16-
9_width_800_height_450.jpg,
https://www.domain.com/media/content/img/image-name-16-
width_1536_height_864.jpg 2x">

<source media=""
srcset="https://www.domain.com/media/content/img/image-name-16-
width_480_height_270.jpg,
https://www.domain.com/media/content/img/image-name-16-
width_960_height_540.jpg 2x">



</picture>
```



Scheint ja ein
ziemlich kleines
Bild zu sein ^^

Decoding Base64 string

Wie die meisten vermutlich schon erwartet haben ...

Base64-String in Online Decoder deiner Wahl werfen oder Vorschau der Bilddatei in Chrome Dev Tools prüfen

```
<picture>

<source media="(min-width: 1200px)"
srcset="https://www.domain.com/media/content/img/image-name-16-
9_width_534_height_300.jpg,
https://www.domain.com/media/content/img/image-name-16-
9_width_800_height_450.jpg 2x">

<source media="(min-width: 960px)"
srcset="https://www.domain.com/media/content/img/image-name-16-
9_width_1200_height_675.jpg,
https://www.domain.com/media/content/img/image-name-16-
9_width_2400_height_1350.jpg 2x">

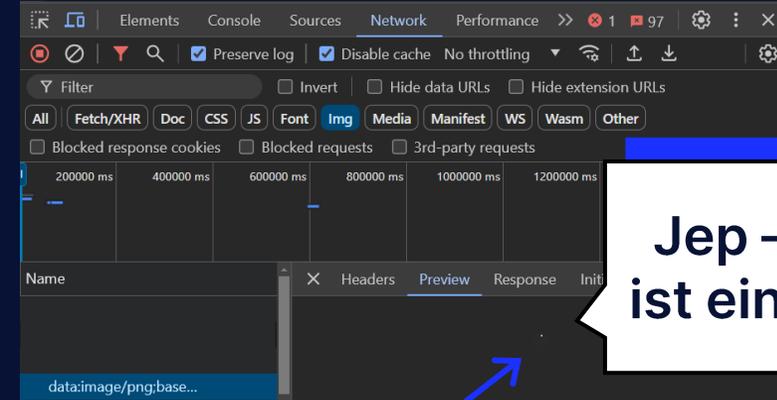
<source media="(min-width: 768px)"
srcset="https://www.domain.com/media/content/img/image-name-16-
9_width_960_height_540.jpg,
https://www.domain.com/media/content/img/image-name-16-
9_width_1920_height_1080.jpg 2x">

<source media="(min-width: 480px)"
srcset="https://www.domain.com/media/content/img/image-name-16-
9_width_800_height_450.jpg,
https://www.domain.com/media/content/img/image-name-16-
9_width_1536_height_864.jpg 2x">

<source media=""
srcset="https://www.domain.com/media/content/img/image-name-16-
9_width_480_height_270.jpg,
https://www.domain.com/media/content/img/image-name-16-
9_width_960_height_540.jpg 2x">



</picture>
```



Jep – das ist ein Pixel

Get your lazy load freak on: lazysizes

Wie findet der Googlebot dieses Picture Setup?

About

High performance and **SEO friendly** lazy loader for images (responsive and normal), iframes and more, that detects any visibility changes triggered through user interaction, CSS or JavaScript without configuration.

javascript

performance

lazyload

lazy-evaluation

lazysizes

responsive-images

<https://github.com/aFarkas/lazysizes>

Vorbereitung der Test Assets

Challenge: Sich völlig neue Themen ausdenken, damit man später leichter nach Inhalten googeln kann

Ein reguläres Picture Tag hat natürlich verschiedene Größen des identischen Bildes

Im Test verschiedene Bilder genutzt, um die Identifizierung der Dateien zu erleichtern



Knaxtilie-1024x1024.jpg



Knaxtilie-750x750.jpg



Knaxtilie-485x485.jpg



Knaxtilie-fallback.jpg



Knaxtilie-image.jpg



Knaxtilie-1024x1024.jpg



Knaxtilie-750x750.jpg



Knaxtilie-485x485.jpg



Knaxtilie-fallback.jpg



Knaxtilie-image.jpg

<picture>

```
<source media="(min-width: 1024px)" srcset="Knaxtilie-1024x1024.jpg">
```

```
<source media="(min-width: 748px)" srcset="Knaxtilie-750x750.jpg">
```

```
<source media="(min-width: 480px)" srcset="Knaxtilie-485x485.jpg">
```

```
<source media="" srcset="Knaxtilie-fallback.jpg">
```

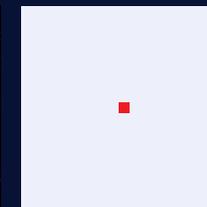
```

```

</picture>



Base64 encoded



Placeholder Image

Reguläres Picture Tag mit / ohne base64

Als Basis, erstmal ein Standard Setup nur mit tatsächlichen Bildern (keinen Platzhaltern)

4 Sourcesets und Image Tag alles JPGs

```
<picture>  
<source media="(min-width: 1024px)" srcset="Knaxtilie-1024x1024.jpg">  
<source media="(min-width: 748px)" srcset="Knaxtilie-750x750.jpg">  
<source media="(min-width: 480px)" srcset="Knaxtilie-485x485.jpg">  
<source media="" srcset="Knaxtilie-fallback.jpg">  
  
</picture>
```

DIE KNAXTILIE: EINE FANTASIEBLUME AUS DEN TIEFEN DER VORSTELLUNGSKRAFT

DIE KNAXTILIE IST EINE FIKTIVE BLUME, DIE IN DER WELT DER FANTASIE UND VORSTELLUNGSKRAFT WURZELT. OBWOHL SIE NICHT IN DER REALEN WELT EXISTIERT, HAT SIE IN DER KREATIVEN SPHÄRE EINEN BESONDEREN PLATZ EINGENOMMEN. DIESE SELTENE UND FASZINIERENDE BLUME WIRD VON KUNSTSCHAFFENDEN, SCHRIFTSTELLERIN UND TRÄUMERIN GLEICHERMÄßEN GESCHÄTZT UND INSPIRIERT.



HERKUNFT & BEDEUTUNG

DER BEGRIFF "KNAXTILIE" IST EINE KREATION, DIE SICH AUS DER VERSCHMELZUNG VERSCHIEDENER WÖRTER UND KLÄNGE ERGIBT. DIE "KNAXTILIE" IST EINE ZUSAMMENFÜHRUNG VON "KNACKS", WAS AUF DAS KNISTERNDE GERÄUSCH ANSPIELT, DAS DIE BLUME BEIM BERÜHREN ERZEUGEN SOLL, UND "LILIE", WAS AUF IHRE ELEGANTE UND ZARTE ERSCHEINUNG HINWEIST. DIESE BLUME EXISTIERT NUR IN DER FANTASIE, DOCH IHRE SYMBOLISCHE BEDEUTUNG ERSTRECKT SICH ÜBER DIE GRENZEN DER VORSTELLUNGSKRAFT HINAUS.

4 Sourcesets und Image Tag mit base64 encodetem Bild

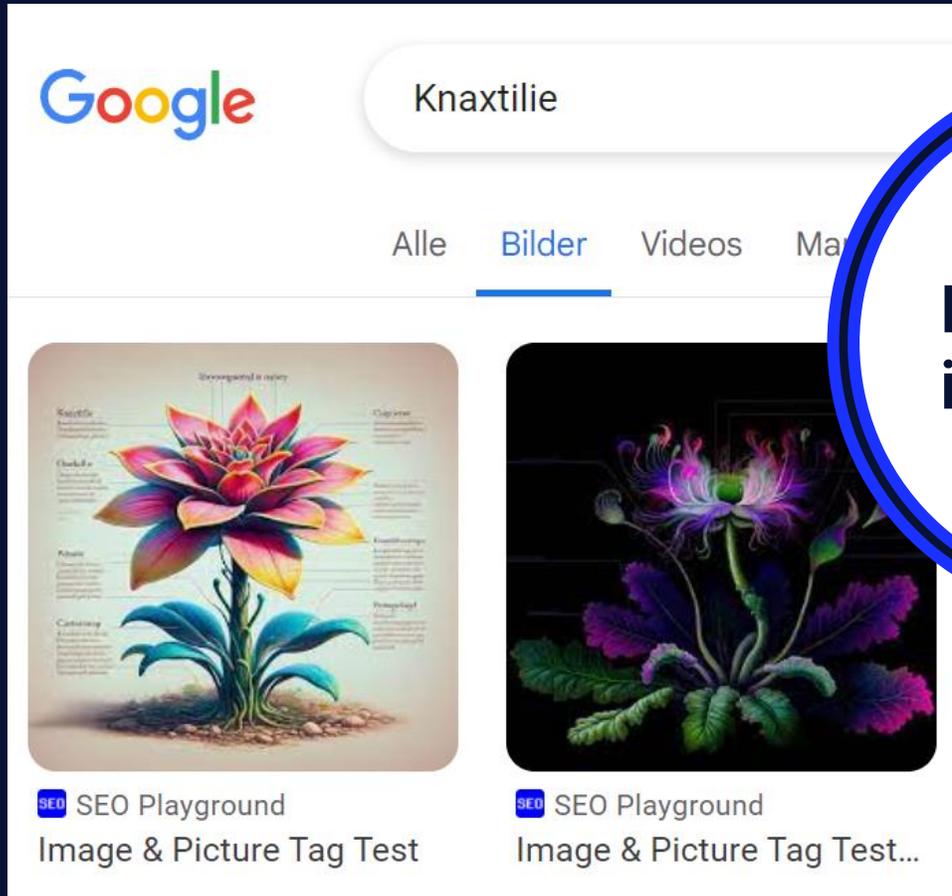
```
<picture>  
<source media="(min-width: 1024px)" srcset="Knaxtilie-1024x1024.jpg">  
<source media="(min-width: 748px)" srcset="Knaxtilie-750x750.jpg">  
<source media="(min-width: 480px)" srcset="Knaxtilie-485x485.jpg">  
<source media="" srcset="Knaxtilie-fallback.jpg">  

```



Stimmt auch, aber dauert deutlich länger

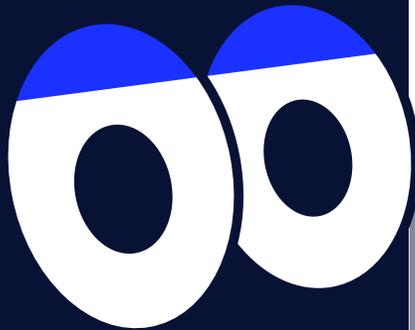
Base64 encodedes Bild aus Image Tag wurde erst knapp 3 Wochen später indexiert



**Indexed:
Image aus
image Tag
(Base64)**

Suche nach Alt-Text zeigt aber nur JPG

Suche nach dem Alt-Text findet nur eins der beiden Bilder



Google Darstellung einer Knaxtillie mit Beschreibung

Alle Bilder Videos Bücher News Finanzen Web Mehr Suchfilter

SEO Playground Image & Picture Tag T...

www.knorpelkompetenznetzwerk.de Allgemeine Informationen zu Knorpel...

Materialien für den Tec... Materialien für den Tec...

StudyHelp Nervenzellen [Aufbau & Funktion] - StudyHelp ...

LEIFiphsik Kontinuierliche Nebel...

OptiYummy Grundlagen: Entwurfsprozess – OptiYummy

Autodesk Wissenswertes über die berech...

Knorpelxperte.de Knie Modell > Aufbau des Knies einfach da...

BWL-Lexikon.de > Adjazenzmatrix » Definition, Erklär...

BWL-Lexikon.de Adjazenzmatrix bei ungerichteten Graphen

TechnikerMathe Seileckverfahren (Resultierende) Vide...

TechnikerMathe Seileckverfahren (Resultierende) Vide...

Studyflix Inzidenzmatrix & Inzidenzliste: Beispiel ...

YouTube Strecken teilen mit Lineal und ...

Quizlet Schaubild „Kontinuierliche u...

mativDigital Neuronale Netze - Funktionsweise...

Die übrigen Suchergebnisse enthalten vielleicht nicht das, wonach du gesucht hast. Trotzdem weitere Suchergebnisse anzeigen

*Verknüpfung
zwischen Alt-Text
und base64 Bild
wird nicht
hergestellt*

Chrome Dev Tools: Verwechslungsgefahr

```
<picture>  
  <source media="(min-width: 1024px)" srcset="Hyroglosithis-1024x1024.jpg">  
  <source media="(min-width: 748px)" srcset="Hyroglosithis-750x750.jpg">  
  <source media="(min-width: 480px)" srcset="Hyroglosithis-485x485.jpg">  
  <source media srcset="Hyroglosithis-fallback.jpg">  
   == $0  
</picture>  
<br>  
<br>  
<h2>Die ersten Anzeichen</h2>  
> <p>...</p>  
<h2>Die Ausbreitung der Krankheit</h2>  
> <p>...</p>  
<h2>Die Suche nach einer Heilung</h2>  
> <p>...</p>  
<h2>Ein unerwarteter Durchbruch</h2>  
> <p>...</p>  
<h2>Die Rettung von Elmbrook</h2>  
> <p>...</p>
```



Rendered size: 1024 × 1024 px
Rendered aspect ratio: 1:1
File size: 1.0 MB
Current source: <https://www.seo-playground.com/tsm/chrissey/imagetest/Hyroglosithis-1024x1024.jpg>

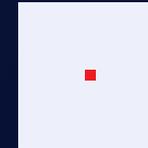
Chrome Dev Tools: Verwechslungsgefahr

```
<picture>  
  <source media="(min-width: 1024px)" srcset="Hyroglosithis-1024x1024.jpg">  
  <source media="(min-width: 748px)" srcset="Hyroglosithis-750x750.jpg">  
  <source media="(min-width: 480px)" srcset="Hyroglosithis-485x485.jpg">  
  <source media srcset="Hyroglosithis-fallback.jpg">  
 == $0  
</picture>  
<br>  
<br>  
<h2>Die erst  
<p>...</p>  
<h2>Die Au  
<p>...</p>  
<h2>Die Su  
<p>...</p>  
<h2>Ein ur  
<p>...</p>  
<h2>Die Re  
<p>...</p>
```

Popup: Hier wird immer das gerenderte Bild angezeigt (das ist keine Vorschau der Datei hinter dem Pfad)



Size: 1024 x 1024 px
Ratio: 1:1
Size: 1.0 MB
Source: https://www.seo-playground.com/tsm/chris.../imagetest/Hyroglosithis-1024x1024.jpg



Picture Tag mit Platzhalter in base64

Setup mit Pixel im Image Tag analog zu Laysizes Setup

SEO Platzgrund

DIE GEHEIMNISVOLLE KRANKHEIT HYROGLOSITHIS

IN EINER KLEINEN STADT NAMENS ELMBROOK BREITETE SICH EINE SELTSAME KRANKHEIT AUS, DIE DIE BEWOHNER IN ANGST UND VERWIRRUNG VERSETZTE. SIE NANNTEN SIE HYROGLOSITHIS, NACH DEM ALTEN WORT FÜR "SÜDER HUND" IN EINER LÄNGST VERGESSENEN SPRACHE. DIEJENIGEN, DIE VON DIESER KRANKHEIT BEFALLEN WAREN, SAHEN ÜBERALL SÜDE HUNDE, EGAL WOHN SIE BLICKTEN.



DIE ERSTEN ANZEICHEN

ALLES BEGANN MIT EINEM JUNGEN MANN NAMENS ETHAN, DER PLÖTZLICH BEHAUPTETE, KLEINE WELPEN AUF DEN STRAßEN VON ELMBROOK ZU SEHEN, WO EIGENTLICH KEINE WAREN. ZU BEGINN DACHTEN DIE LEUTE, ER MACHE NUR EINEN SCHERZ ODER SEI ÜBERMÜDET, ABER ALS IMMER MEHR MENSCHEN BEGANNEN, ÄHNLICHE DINGE ZU BERICHTEN, WUCHS DIE BEDORGNIS IN DER GEMEINSCHAFT.

DIE AUSBREITUNG DER KRANKHEIT

DIE KRANKHEIT VERBREITETE SICH WIE EIN LAUFFEUER DURCH DIE STADT. MENSCHEN ALLER ALTERSGRUPPEN BERICHTETEN

III

TEST PLATZHALTER DER BASE64

4 Sourcesets

Image Tag mit Platzhalter-Pixel in base64



Hyroglosithis-1024x1024.jpg



Hyroglosithis-750x750.jpg



Hyroglosithis-485x485.jpg



Hyroglosithis-fallback.jpg



Base64 encoded

Was passiert, wenn man das Image Tag komplett weglässt?

Jep – dann wird im Browser gar nichts dargestellt

PROBATO-NAUTUS: DAS REVOLUTIONÄRE WERKZEUG DER ZUKUNFT

INER WELT, DIE STÄNDIG NACH INNOVATIONEN STREBT, HAT EINE BAHNBRECHENDE ERFINDUNG DIE ART UND WEISE, WIE MENSCHEN ARBEITEN UND INTERAGIEREN, FÜR IMMER VERÄNDERT - DER PROBATO-NAUTUS. DIESSES REVOLUTIONÄRE WERKZEUG KOMBINIERT DIE FORTSCHRITTSCHNELLSTE TECHNOLOGIE MIT KREATIVEM DESIGN, UM EINE VIELZAHL VON AUFGABEN EINFACHER UND EFFIZIENTER ZU GESTALTEN.

FUNKTIONSWEISE UND ANWENDUNG

DER PROBATO-NAUTUS IST EIN VIELSEITIGES GERÄT, DAS SOWOHL ALS WERKZEUG ALS AUCH ALS BEGLEITER DIENT. AUSGESTATTET MIT KÜNSTLICHER INTELLIGENZ UND EINER REIHE VON SENSOREN, KANN ES IN UNTERSCHIEDLICHSTEN UMGEBUNGEN EINGESATZT WERDEN - SEI ES IN DER LANDWIRTSCHAFT, IM BAUWESEN, IN DER EXPLORATION ODER Sogar IM ALLTAG EINES JEDEN EINZELNEN. VON DER ANALYSE VON BODENPROBEN BIS HIN ZUR ÜBERWACHUNG DER GESUNDHEIT VON NUTZTIEREN, VON DER KARTIERUNG UNBEKANNTER GEBIETE BIS HIN ZUR UNTERSTÜTZUNG BEI TÄGLICHEN AUFGABEN IM HAUSHALT - DER PROBATO-NAUTUS IST EIN WAHRES MULTITALENT.

DESIGN UND ERGONOMIE

EIN SCHLÜSSELMERKMAL DES PROBATO-NAUTUS IST SEIN DURCHDACHTES DESIGN, DAS SOWOHL FUNKTIONALITÄT ALS AUCH BENUTZERFREUNDLICHKEIT IN DEN VORDERGRUND STELLT. MIT EINEM ERGONOMISCHEN GRIFF UND INTUITIVEN BEDIENELEMENTEN LIEGT DAS GERÄT ANGENEHM IN DER HAND UND ERMÖGLICHT PRÄZISES ARBEITEN ÜBER LÄNGERE ZEITRÄUME HINWEG. SEIN ROBUSTES GEHÄUSE IST SO KONSTRUIERT, DASS ES DEN HERAUSFORDERUNGEN VERSCHIEDENER ARBEITSUMGEBUNGEN STANDHÄLT, SEI ES UNTER EXTREMEN WETTERBEDINGUNGEN IM FREIEN ODER IN ANSPRUCHSVOLLEN INDUSTRIELLEN ANWENDUNGEN.

INNOVATION UND FORTSCHRITT

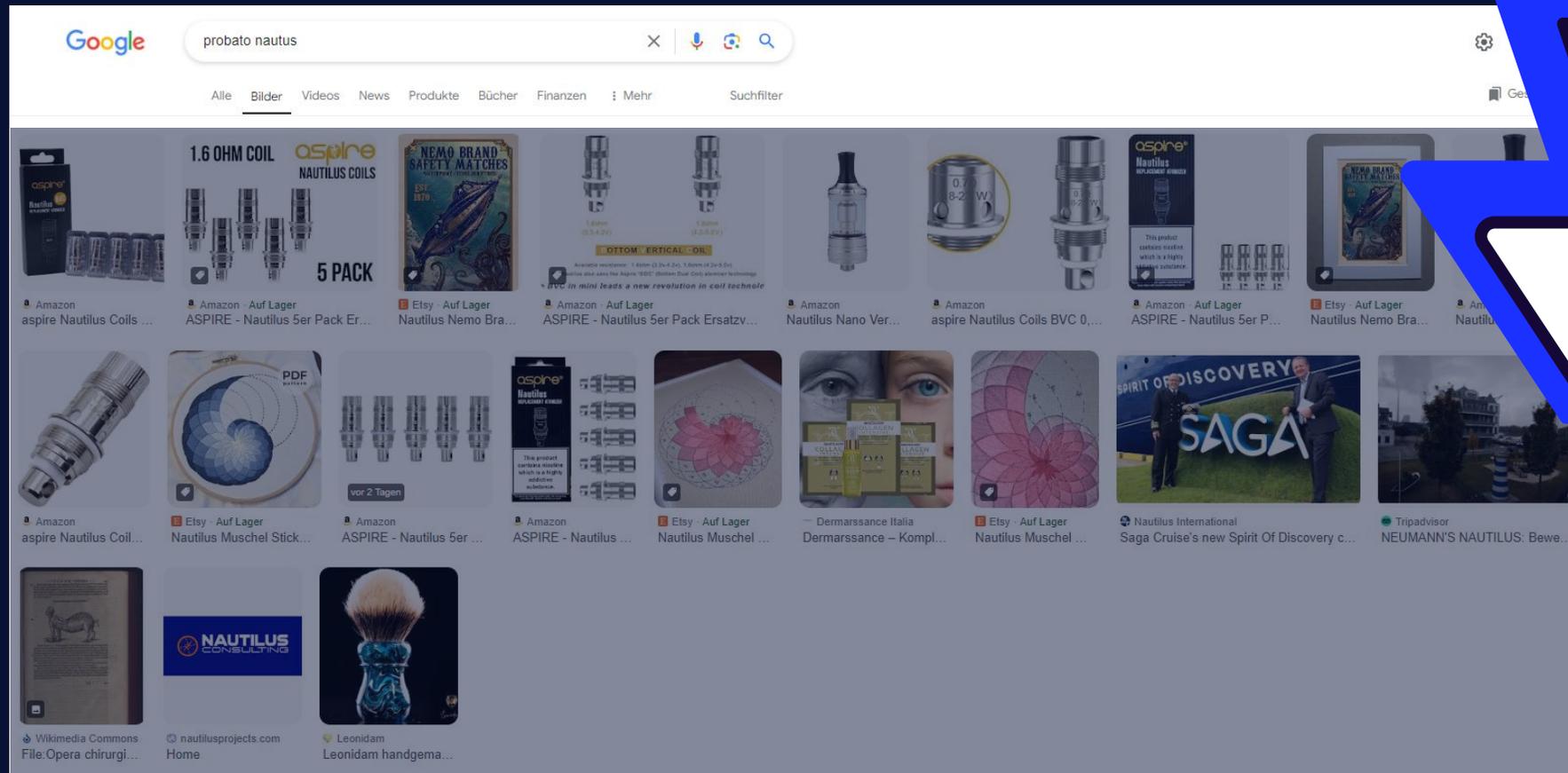
DER PROBATO-NAUTUS STEHT EXEMPLARISCH FÜR DIE UNERMÜDLICHE SUCHE NACH INNOVATION UND FORTSCHRITT IN DER MENSCHLICHEN GESCHICHTE. MIT SEINER FÄHIGKEIT, KOMPLEXE PROBLEME ZU LÖSEN UND NEUE HORIZONTE ZU ERSCHLIESSEN, HAT ER BEREITS DIE GRENZEN DESSEN, WAS MÖGLICH IST, ERWEITERT UND DIE MENSCHHEIT IN EINE AUFGEBENDE ZUKUNFT GEFÜHRT. DOCH TROTZ ALL SEINER LEISTUNGSFÄHIGKEIT BLEIBT DER PROBATO-NAUTUS STETS EIN WERKZEUG IN DEN HÄNDEN SEINER BENUTZER, BEREIT, IHRE TRÄUME UND VISIONEN ZU UNTERSTÜTZEN UND ZU VERWIRKLICHEN.

JS WURDE AUSGEFÜHRT

Picture Tag OHNE Image Tag

```
<picture>
<source media="(min-width: 1024px)" srcset="Probato-Nautus-1024x1024.jpg">
<source media="(min-width: 748px)" srcset="Probato-Nautus-750x750.jpg">
<source media="(min-width: 480px)" srcset="Probato-Nautus-485x485.jpg">
<source media="" srcset="Probato-Nautus-fallback.jpg">
</picture>
```

Das wird natürlich auch nix



Wenn dir die Indexierung deiner Bilder nicht egal ist oder du mindestens eine Bilder-Sitemap hast, würd ichs lassen!



Außerdem

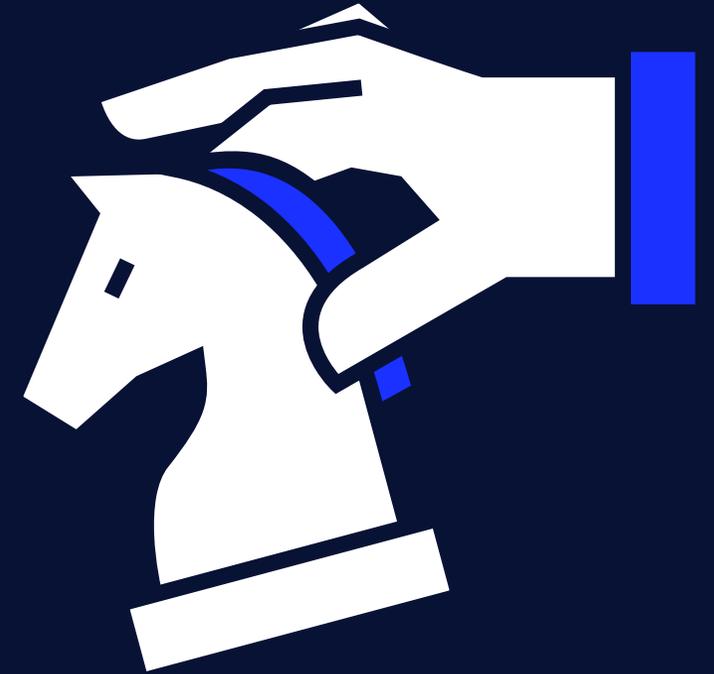
Oft ist das Thema

Compression,

Cropping & Caching

ein krasserer Hebel beim

Thema Page Speed & Bilder



*The last 30 Minutes
in a nutshell*

Uff ... das war viel Input

Zusammenfassung to go

1

Alle **SEO-relevanten Inhalte**, die indexiert werden sollen, **müssen innerhalb von 5 Sekunden** nach dem Download des initialen HTML-Dokuments **geladen haben**.

2

TTFB von bis zu **10 Sekunden** und **JavaScript Delays von bis zu 5 Sekunden** werden von Google **toleriert**, können aber zu langsameren Indexierung führen.

3

Bilder, die indexiert werden sollen, müssen in einem **Image Tag** vorliegen, oder mindestens über **Bilder Sitemap** erreichbar sein.

4

Base64-encodete Bilder werden **langsamer indexiert** und die **Verknüpfung zum Alt-Text wird nicht (zuverlässig) hergestellt**.

Motivationsschub gefällig?

*Auch in echten
Kunden-
projekten hat
es Impact*



Bei der anstehenden AEM-Migration wird das **Lazysizes Setup zurückgelassen** und dafür Image Performance über Edge Delivery Services neu angegangen



Cloud-Migration hat Anbindung ans PIM System vorübergehend beeinträchtigt – **Abruf der Produktnamen schlug fehl** → Markups waren down



Relevante Inhalte sind tatsächlich nicht indexiert, da Ladevorgang aufgrund von turbo-verschachtelten Shadow DOM Routes* zu lange war

So würde ich es auf Metaebene zusammenfassen

Vieles kann man mit wenigen Mitteln und Codingerfahrung selbst testen!

Over-engineeren im Tech SEO kann auch schnell mal ins Auge gehen!

→
Statisch Sachen laden ist und bleibt halt einfach geil für SEO!

Alles bleibt wie es ist:

*Je komplizierter du es
Google machst, desto eher
riskierst du deinen SEO-
Erfolg!*

*Noch
unentschlossen?*

Mini-*Lifehacks* für den Start

Live im Web testen

lokal verhalten sich Dinge anders und SEO passiert im Internet ^^

Automatic Upload

beim Speichern im Code Editor konfigurieren

Cleanen Browser nutzen

ohne Plugins / Extensions, Logins etc.

Basic Command Lines lernen

...bin anfangs schon am „Passwort reinkopieren“ gescheitert ^^

Browser Cache ist lästig

viel mehr als du denkst – nicht verwirren lassen

Safari hat eigene Regeln

wunder dich nicht, wenn plötzlich dein Text nicht angezeigt wird o.ä.

Test-Setup immer testen

ob du auch wirklich testest, was du denkst

Tests auch wieder löschen

zum Beispiel deine Cronjobs, die alle 5 Minuten URLs aufrufen o.ä.

Woran hat et jelegen?

Lieber gleich
angewöhnen Code
sinnvoll zu
kommentieren und
lernen Error Handling
einzubauen!



Einfach machen!

Risiken & Nebenwirkungen

Geheimwaffe gegen
Imposter Syndrom &
Mansplaining etc.

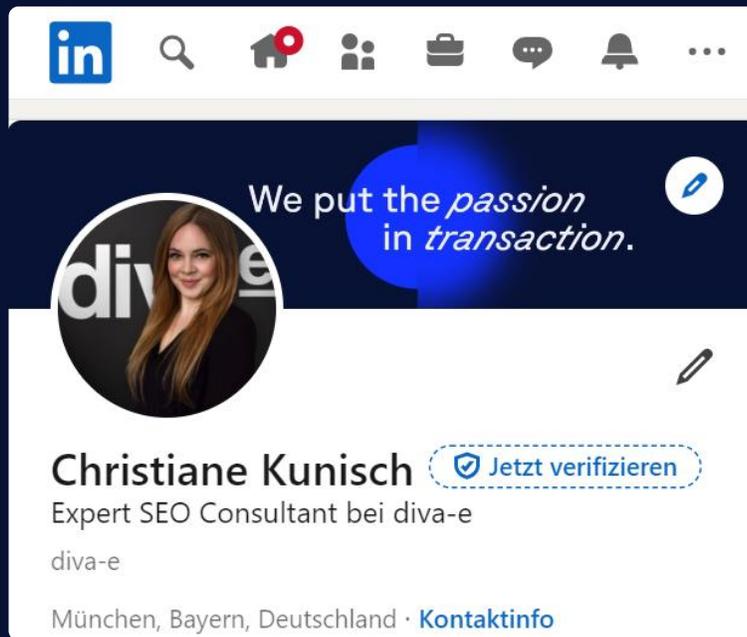
Sharing is Caring

Die SEO
Community ist
halt einfach geil!

Ganz wichtig:
nicht verzweifeln!



Happy to connect



LinkedIn profile of Christiane Kunisch. The profile includes a navigation bar with icons for home, search, network, work, messages, and notifications. The header features a profile picture of Christiane Kunisch, a blue speech bubble with the text "We put the passion in transaction.", and a "diva-e" logo. Below the header, the name "Christiane Kunisch" is displayed with a "Jetzt verifizieren" button. Her title is "Expert SEO Consultant bei diva-e" and her company is "diva-e". The location is "München, Bayern, Deutschland" and there is a "Kontaktinfo" link.



Code Beispiele, Links und Inspo für
weitere Tests gibt's hier zum Download



A QR code is displayed below the text, which likely links to the download mentioned in the text.

Danke.

Ihr seid
fantastisch!

