

Page Speed Optimization

Only a fast website is a good website

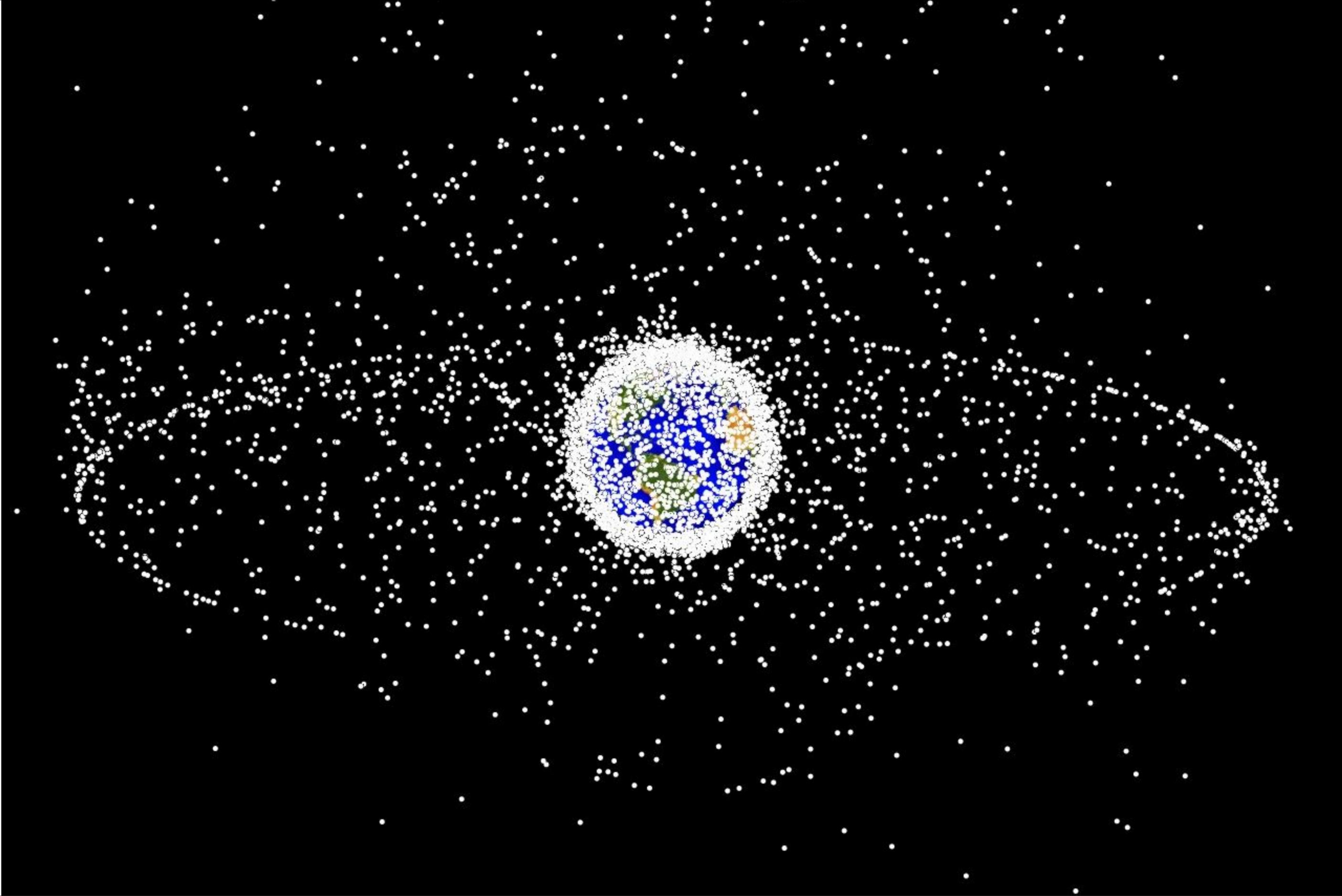
11.10.2019

MAG
MELL



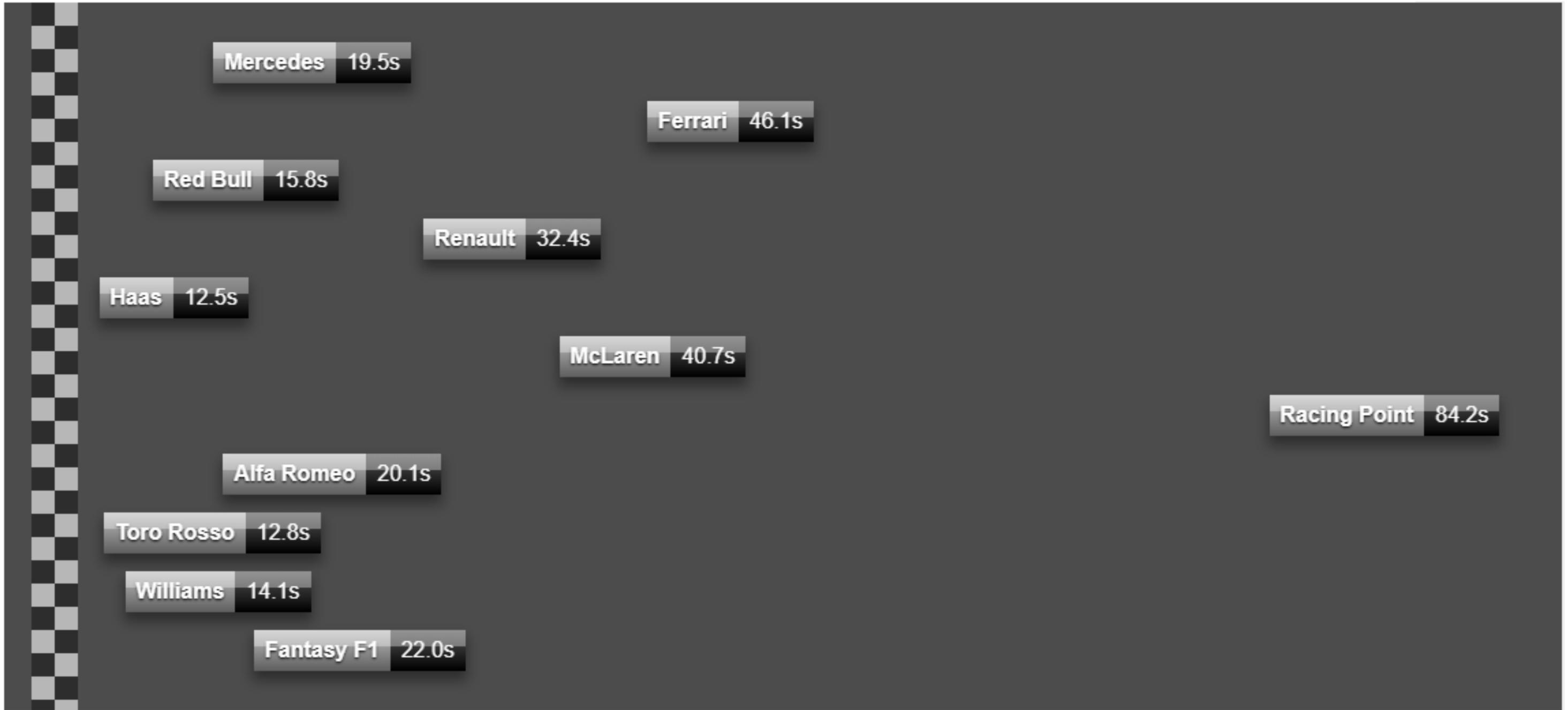
warum reden wir überhaupt über pagespeed?

- definition für »schnell«: 200ms (100ms interne Zielsetzung bei Google)
- definition »langsam« kommt aus Usability-Untersuchungen:
2s gilt als akzeptabel, Absprungrate massiv höher ab 3s
- Durchschnitt für die Top 1000 Webseiten ist bei 6.8 Sekunden
> da ist noch einiges an Nachholbedarf :)
- google.com bewegt sich aktuell bei rund 2s, weil sie nicht mehr "pure" sind



das moderne web ist komplex

- schnelle Webseiten sind vor allem statische, heftig gecachte Webseiten ohne Dynamik
- aktuell schnellste bekannte Kandidaten
400ms <https://www.hm.com>
580ms für [wikimedia.org](https://www.wikimedia.org)



ergebnis der komplexität

- Direktes Ergebnis dieser Komplexität? Sieht man schön an der Untersuchung von Jake Archibald: <https://jakearchibald.com/2019/f1-perf/>
- loading times of F1 pages
slowed down to 3G (384kbit/s to 7.2Mbit/sec)
- enorme Bandbreite an Geschwindigkeiten
- viele Webseiten so langsam, dass sie keinen Spass mehr machen

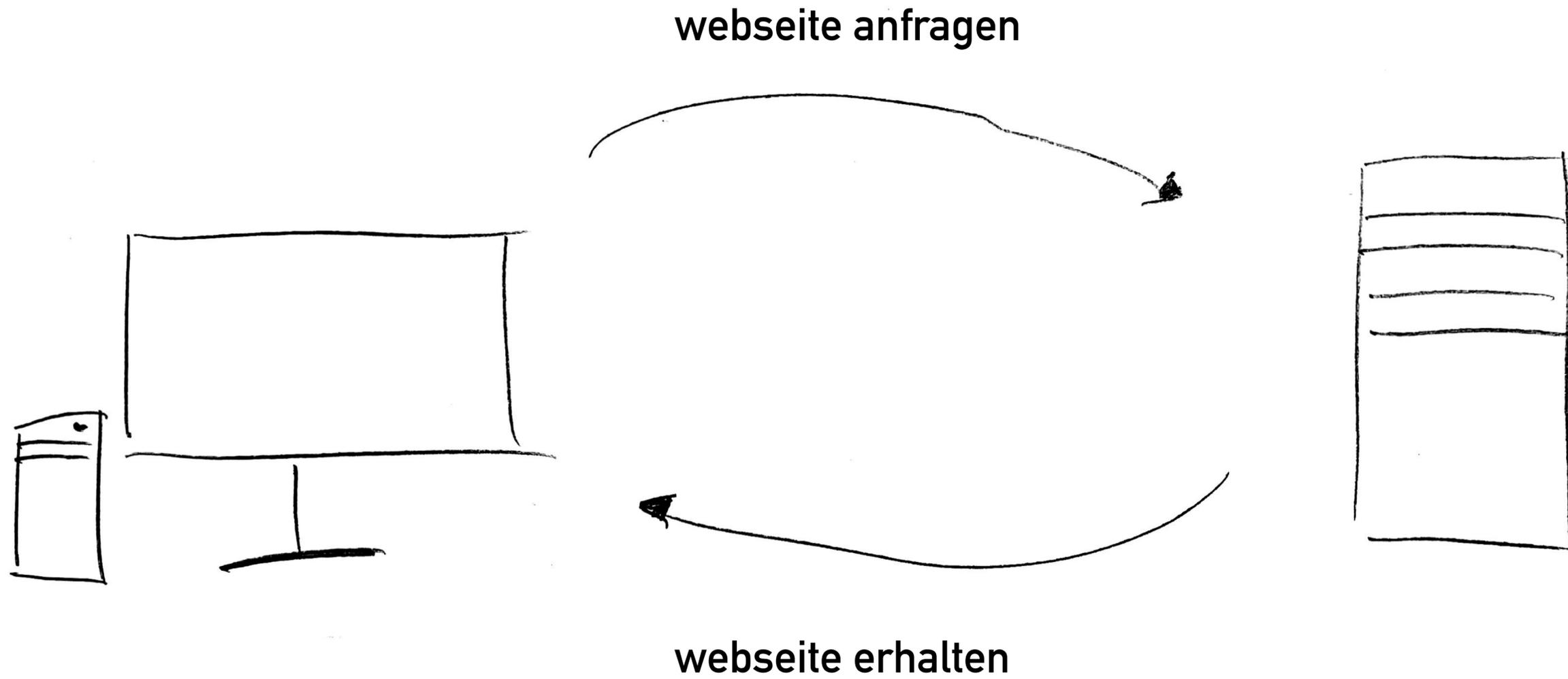


niemand wartet gerne

- Es wird ein bisschen besser wenn du genau weisst worauf du wartest - Information und Ablenkung kann helfen
- Wirklich schlecht ist es, wenn »langsam« dazu führt dass ein Service oder ein Shop garnicht mehr funktioniert weil es so langsam ist
- dabei sollte gerade das - im eigenen Interesse! - schnell und schmerzfrei sein

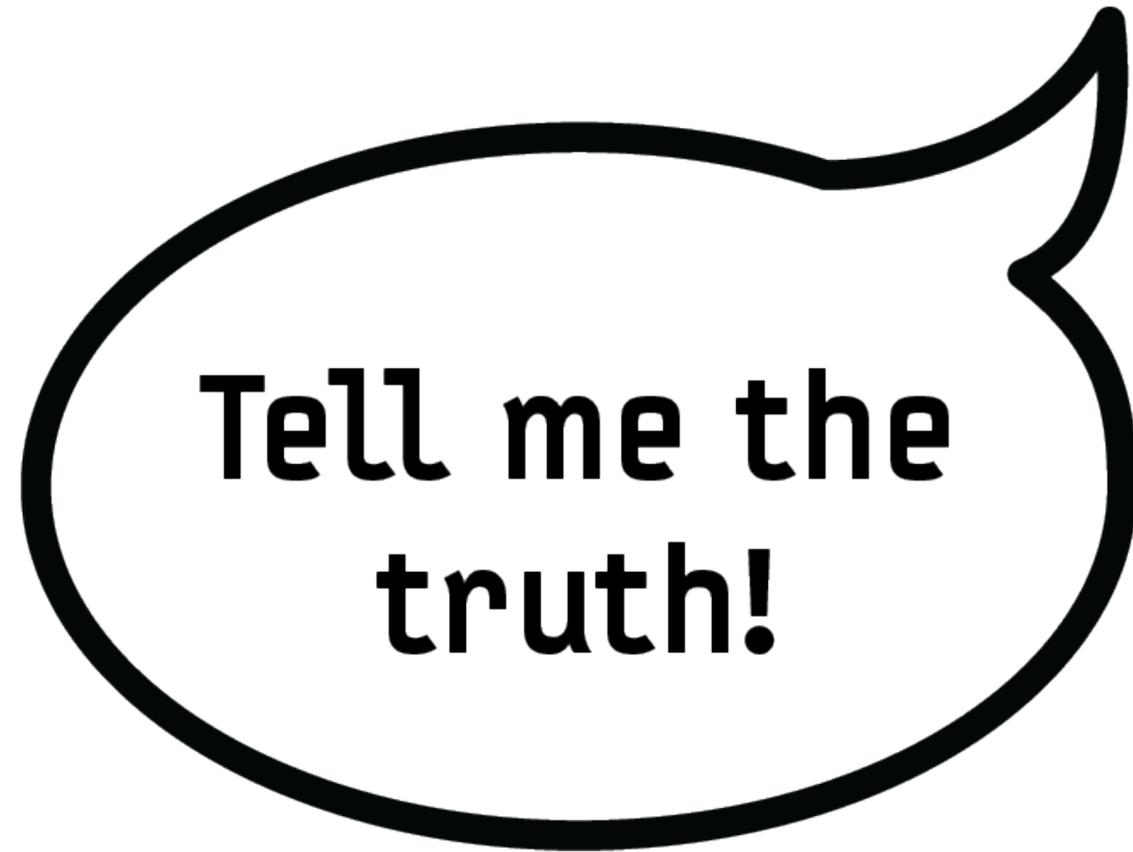
niemand wartet gerne

- Untersuchungen mit Usern und Statistiken haben gezeigt dass es mit allen wichtigen Metriken steil bergab geht, wenn die Seite zu langsam ist - User-Zufriedenheit, Interaktion, Conversions
- 50% erwarten ein funktionales Ergebnis in 2 Sekunden oder weniger
- 60% verlassen die Seite wenn 3 Sekunden überschritten werden -
75% der Abspringer kaufen dann woanders



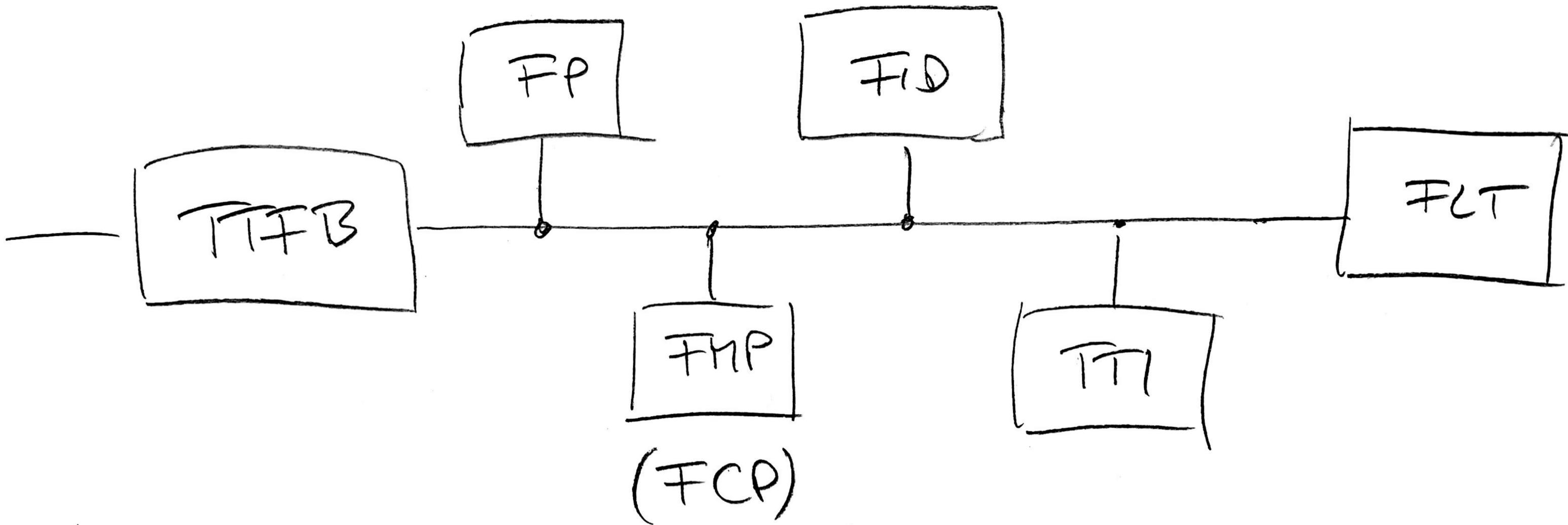
back to basics?

- Dabei könnte alles so einfach sein ;)
Isses aber nicht.
- Auf Server-Seite: Webserver, Programming, Templates, Datenbank, externe Werkzeuge, APIs, verschiedene Caching-Mechanismen
- Unterwegs: Routing, Netzwerk-Wechsel, Load Balancer, Firewalls, Proxy Server and Proxy Caches
- Client: Firewall, Virens scanner, Browser, Plugins, lokales Caching



don't trust the numbers

- Dabei hilft es nicht nur auf die Zahlen zu schauen
- Zahlen sind komplett neutral, was wir daraus ableiten ist wichtig - es heisst nicht umsonst »data science«, und nicht »wir schauen auf Zahlen«
- Verschiedenste Faktoren beeinflussen das User-Erlebnis:
Content- und Bildqualität, Videoinhalte, Animationen, Interaktivität,
akkurate und/oder aktuelle Daten (Stauübersicht, Unwetterwarnung ...)



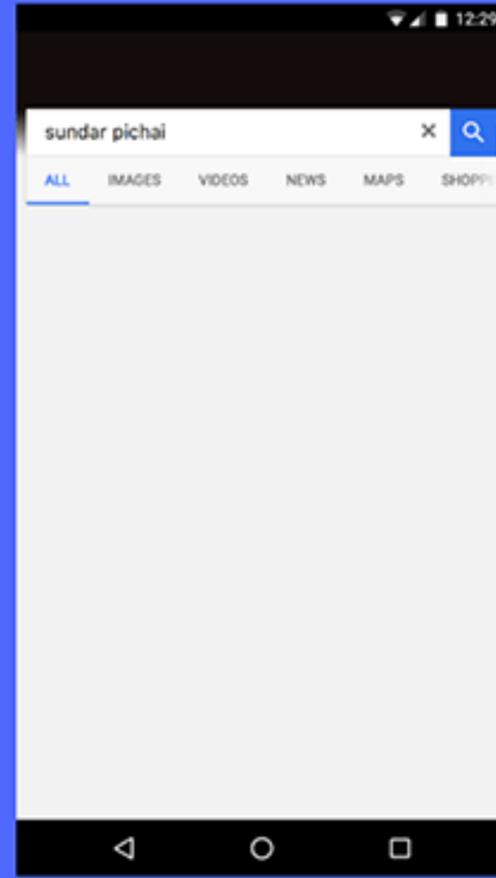
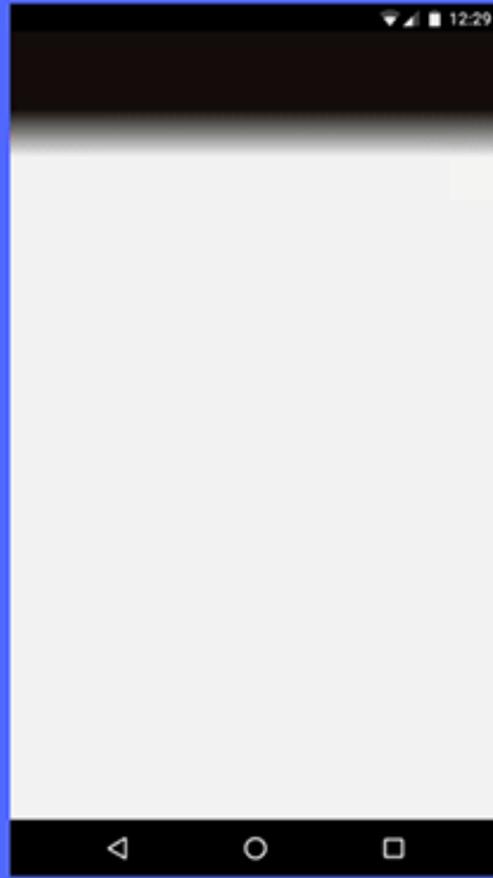
es gibt trotzdem ein paar zahlen, deren namen wir zumindest kennen sollten

- time to first byte - Anfrage am Server, Verbindungsaufbau, Backend, zurück zum Browser
- first paint - erste sichtbare Änderung (Hintergrundfarbe, Logo)
- first contentful paint - das erste Mal, dass Inhalt erscheint
- first meaningful paint - der erste nutzbare Inhalt, hier könnten wir anfangen zu lesen
- first interactive delay - die Seite ist bereit für Interaktionen, wenn sie auch nicht sofort durchgeführt werden
- time to interactive - Webseite ist komplett interaktiv
- fully loaded time - wenn für rund 2 Sekunden nix mehr passiert, dann ist die Seite »done«

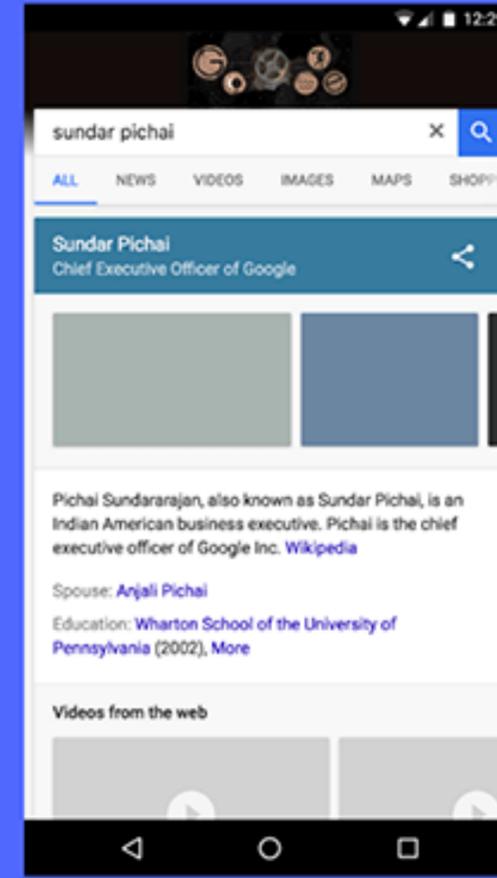
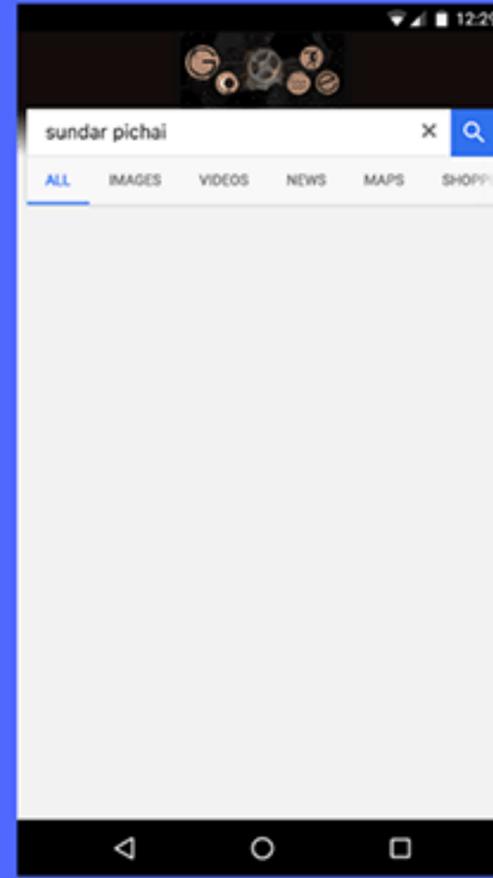
Bild: aus der Google Hilfe, so ähnlich auch bei Pagespeed Insights eingebunden



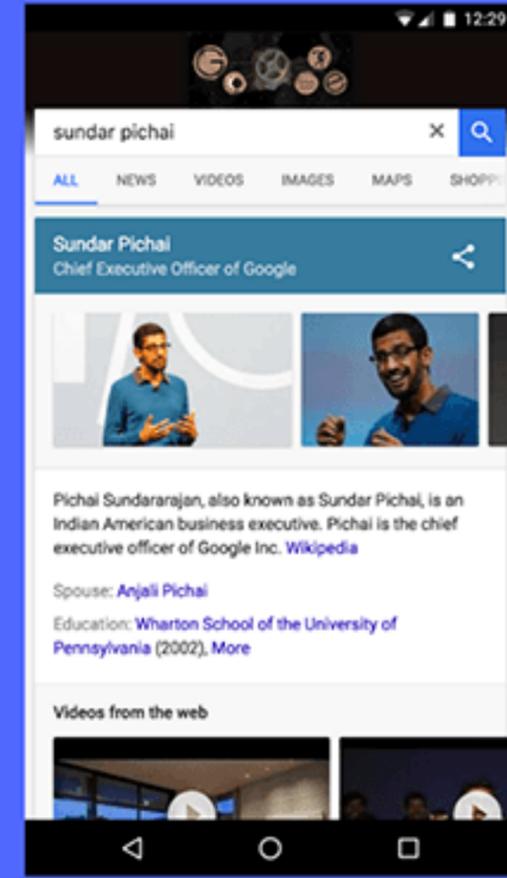
First Paint
(FP)



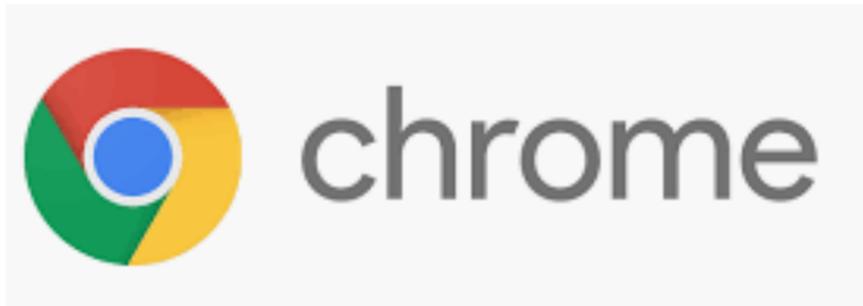
First Contentful
Paint (FCP)



First Meaningful
Paint (FMP)



Time to
Interactive (TTI)



kleine werkzeug-übersicht

- Chrome - Performance Tab in den Developer Tools
- Webpagetest <https://www.webpagetest.org/>
- Pingdom Page Speed <https://tools.pingdom.com/>
- PageSpeed Insights <https://developers.google.com/speed/pagespeed/insights/>
- GTMetrix <https://gtmetrix.com/>

kleine werkzeug-übersicht

- alle Werkzeuge testen ähnlich, auf ähnlichen Grundlagen, aber:
- unterschiedliche technische Stände
- unterschiedliche Schwerpunkte
- schönste Aufbereitung für den Einstieg: GTMetrix

- Beispiele mit bild.de für alle Tools



Latest Performance Report for: https://www.blic.rs/

Report generated: Mon, Jun 3, 2019 8:36 PM +0200

Test Server Region: London, UK

Using: Firefox (Desktop) 54.0.1, PageSpeed 1.15-gt1.2, YSlow 3.1.8

Additional tips: Use a CDN

Performance Scores

PageSpeed Score

F (40%) ▾

YSlow Score

F (42%) ▾

Page Details

Fully Loaded Time

29.0s ▾

Total Page Size

4.96MB ▾

Requests

457 ▾

PageSpeed

YSlow

Waterfall

Timings

Video

History

RECOMMENDATION	GRADE	TYPE	PRIORITY
▾ Minimize redirects	F (0) ▾	CONTENT	HIGH
▾ Serve scaled images	F (0) ▾	IMAGES	HIGH
▾ Enable gzip compression	F (0) ▾	SERVER	HIGH
▾ Leverage browser caching	F (2) ▾	SERVER	HIGH
▾ Serve resources from a consistent URL	F (12) ▾	CONTENT	HIGH
▾ Enable Keep-Alive	F (37) ▾	SERVER	HIGH



Latest Performance Report for: <http://bild.de/>

Report generated: Thu, Oct 10, 2019 2:50 PM -0700

Test Server Region: Vancouver, Canada

Using: Chrome (Desktop) 75.0.3770.100, PageSpeed 1.15-gt1.2, YSlow 3.1.8

Looks like you might not be using a CDN [Why should I use a CDN? »](#)

Performance Scores

PageSpeed Score F (40%) ▾	YSlow Score F (45%) ▾
-------------------------------------	---------------------------------

Page Details

Fully Loaded Time 20.9s ▾	Total Page Size 5.64MB ▾	Requests 424 ▾
-------------------------------------	------------------------------------	--------------------------

- PageSpeed**
- YSlow
- Waterfall
- Timings
- Video
- History

RECOMMENDATION	GRADE	TYPE	PRIORITY
Minimize redirects	F (0) ▾	CONTENT	HIGH
Leverage browser caching	F (0) ▾	SERVER	HIGH
Defer parsing of JavaScript	F (0) ▾	JS	HIGH
Optimize images	F (12) ▾	IMAGES	HIGH
Minimize request size	F (16) ▾	CONTENT	HIGH
Avoid landing page redirects	F (45) ▾	SERVER	HIGH
Minify JavaScript	C (79) ▾	JS	HIGH
Inline small CSS	B (83) ▾	CSS	HIGH

What do my scores mean?

Rules are sorted in order of impact upon score
Optimizing rules at the top of the list can greatly improve your overall score.

Not every recommendation will apply to your page
The recommendations are meant to be generic, best practices; some things will be out of your control (eg. external resources) or may not apply to your page.

[Learn more about PageSpeed/YSlow scores and how they affect performance.](#)



http://bild.de/

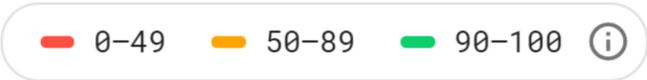
ANALYZE

MOBILE

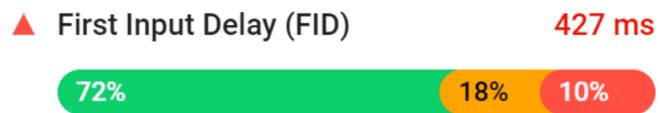
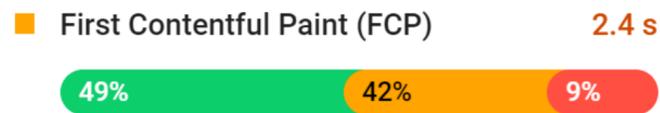
DESKTOP



https://www.bild.de/



Field Data — Over the last 30 days, the field data shows that this page has a **Slow** speed compared to other pages in the [Chrome User Experience Report](#). We are showing [the 90th percentile of FCP](#) and [the 95th percentile of FID](#).



Show Origin Summary

Lab Data



▲ First Contentful Paint 6.0 s

▲ First Meaningful Paint 6.1 s

▲ Speed Index 8.6 s

▲ First CPU Idle 15.2 s

▲ Time to Interactive 20.7 s

▲ Max Potential First Input Delay 490 ms

Web Page Performance Test for bild.de

From: Frankfurt, Germany - Dynatrace - Chrome - Cable
10.10.2019, 23:58:14

[Need help improving?](#)

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

Tester: i-082cde686f21ceaf0

Test runs: 3

[Re-run the test](#)

[Raw page data](#) - [Raw object data](#)

[Export HTTP Archive \(.har\)](#)

[View Test Log](#)

Performance Results (Median Run)

	Load Time	First Byte	Start Render	First Contentful Paint	Speed Index	Last Painted Hero	First CPU Idle	Document Complete			Fully Loaded			
								Time	Requests	Bytes In	Time	Requests	Bytes In	Cost
First View (Run 1)	9.072s	0.404s	1.700s	1.681s	3.184s	8.500s	> 8.734s	9.072s	239	3,148 KB	10.643s	260	3,421 KB	\$\$\$\$\$
Repeat View (Run 1)	5.790s	0.230s	1.000s	1.038s	1.517s	2.100s	> 5.204s	5.790s	103	416 KB	7.744s	120	585 KB	

[Plot Full Results](#)

Test Results

Run 1:

Waterfall	Screenshot

Pingdom Website Speed Test

Enter a URL to test the page load time, analyze it, and find bottlenecks.

URL

http://bild.de/

Test from

Europe - Germany - Frankfurt ▾

START TEST

The internet is fragile. Be the first to know when your site is in danger.

START YOUR FREE 14-DAY TRIAL



Your Results:

↓ DOWNLOAD HAR

↑ SHARE RESULT



Performance grade

D 65

Page size

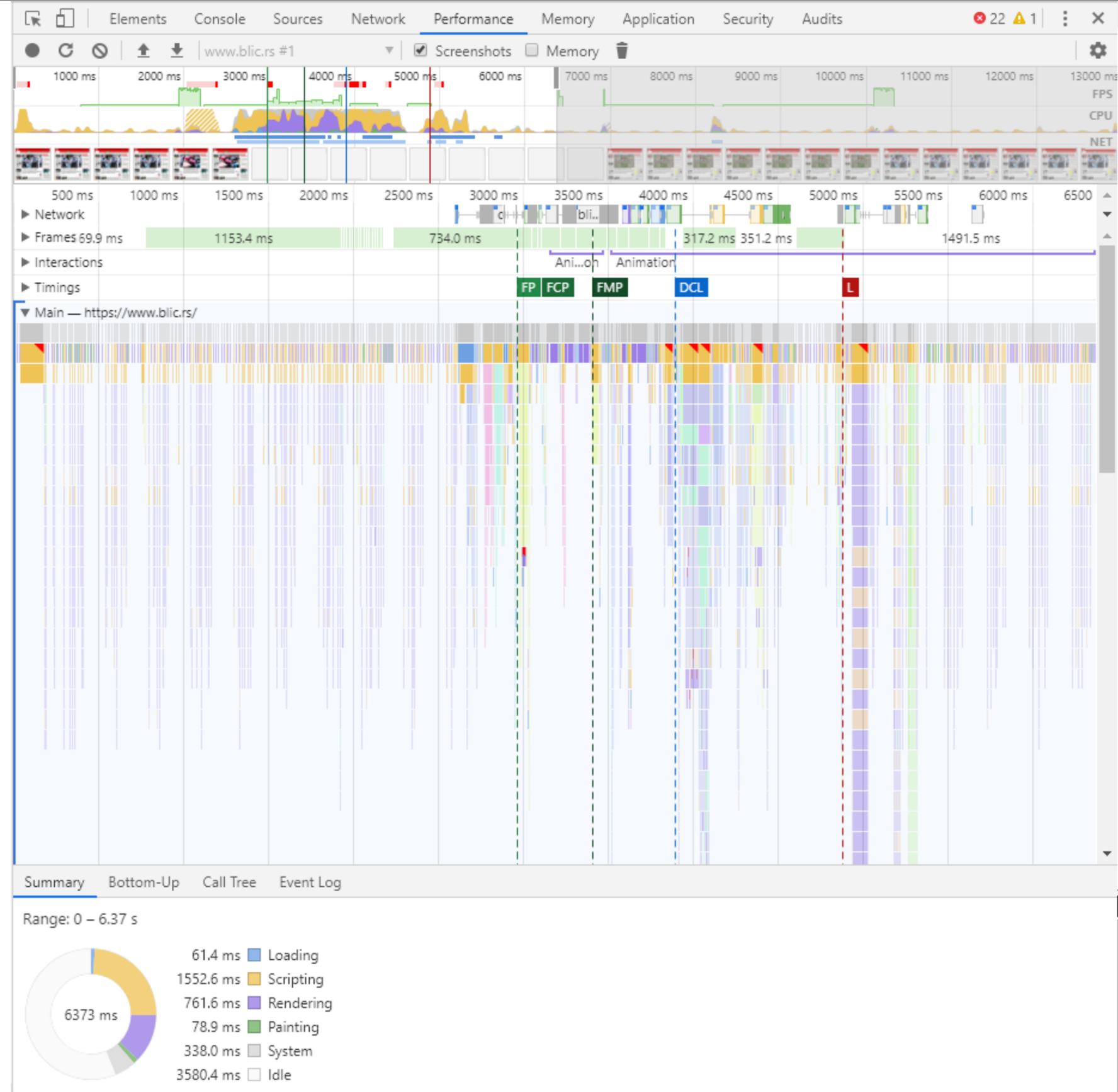
4.3 MB

Load time

2.35 s

Requests

248





RECOMMENDATION	GRADE		TYPE	PRIORITY
▼ Minimize redirects	F (0)	▼	CONTENT	HIGH
▼ Leverage browser caching	F (0)	▼	SERVER	HIGH
▼ Defer parsing of JavaScript	F (0)	▼	JS	HIGH
▼ Optimize images	F (12)	▼	IMAGES	HIGH
▼ Minimize request size	F (16)	▼	CONTENT	HIGH
▼ Avoid landing page redirects	F (45)	▼	SERVER	HIGH
▼ Minify JavaScript	C (79)	▼	JS	HIGH
▼ Inline small CSS	B (83)	▼	CSS	HIGH
▼ Specify a cache validator	B (87)	▼	SERVER	HIGH

was können wir tun? eine ganze menge.

- Dabei ist gar nicht interessant was wir in der Backend-Programmierung erreichen können - weil es meistens nicht die relevanteste Größe ist (Ausnahmen: überlastete Webseiten und Shops - anderes Thema)
- follow the (white) rabbit, als Beispiel von GTMetrix - jedes Werkzeug zeigt uns Handlungsempfehlungen, vorsortiert nach Relevanz
- Übung für zu Hause:
ein Projekt in allen Tools auf 100% und A+ bringen, zumindest einmal :)



Quelldatei: 800x1020px
139kB



verkleinert: 200x255px
14,1kB



optimiert: 200x255px
13,3kB

baustelle 1: bildgröße

- Größter Faktor ist in ganz vielen Fällen:
Bilder sauber verkleinern & optimieren
- Verkleinerung: idealerweise im CMS, über Erweiterungen, Plugins (adaptive image), externe Services (cloudimage, responsive.io etc.)
- Ganz toll in Contao: Steuerung über Bildregeln, Viewport-abhängig
- Optimierungs-Tools: tinypng.com, [imageoptim](https://imageoptim.com), squosh.app - und viele weitere externe Services und Plugins

baustelle 2: caching

- Caching in Contao verwenden: serverseitig, clientseitig
(im CMS einstellbar)
- Caching im Apache verwenden: über htaccess-Datei, gute
Vorlagen waren in Contao 3.5 enthalten
<https://github.com/contao/core/blob/3.5/.htaccess.default>



MAG
MELL

weitere möglichkeiten kurz angerissen

- geotargeting - Inhalte nah am Besucher ausliefern, entweder über einen Server, oder über Content Delivery Networks auch weltweit
- above the fold - Optimierung der sichtbaren Inhalte, andere Layouts, Inhalte und Stile werden später nachgeladen
- lazy loading - Inhalte werden generell erst nachgeladen wenn sie gebraucht werden (vor allem: Bilder, aber auch Videos)
- Unterstützung dabei: zum Beispiel mit dem Google Pagespeed Modul für Apache, diverse Web-Plugins zum Beispiel für Build-Werkzeuge wie npm, gulp, webpack usw.



dirty tricks - anbieter

- Eine Menge Firmen bieten »Pagespeed-Optimierung«, oder »Technisches SEO« an
- Meistens machen die eine banale Checkliste durch, wenden minimal Energie auf für maximales Ergebnis - nicht schlecht, aber oft überteuert, pur zahlenbasiert, nicht individuell

dirty tricks - optimierungen

- Es gibt eine Menge Tricks und Maßnahmen um noch ein bisschen mehr rauszuholen
- Hardcore-Optimierungen gehen meist zu Lasten der Qualität
- Die meisten dirty Tricks haben negative Auswirkungen
- Empfehlung: Standards verwenden - das Web entwickelt sich erfreulich schnell weiter, die Tricks sind kurzlebig



Quelldatei:
800x1020px
139kB



verkleinert und optimiert:
200x255px
13,3kB



trick - oversized 0%:
400x510px
8kB

dirty tricks - ein beispiel

- Ein Bild wird in doppelter Größe, aber mit 0% Qualität abgespeichert
- Es kommt ein bisschen auf das Bild an, aber hiermit können nochmal 25-50% gegenüber dem verkleinerten und optimierten Bild eingespart werden - der Browser übernimmt hier viel des Heavy Lifting
- Besserer Ansatz: moderne Werkzeuge und Formate im Rahmen der Standards nutzen -> JPEG 2000 und WebP als Bildformate bieten ähnliche Optimierungsraten, können z.B. mit squosh.app erzeugt werden

**thanks for listening
time for questions**

bonus content

aus der praxis: was bringt optimierung

- am Beispiel eines echten Kundenprojekts
- mit Screenshots aus GTMetrix
- künstlich alles deaktiviert was richtig gemacht werden kann
(Caching in Contao, am Server etc.)
- Optimierung nach Empfehlungen bis 85/90 Punkte erreicht
waren

testlauf 1: null optimierung

- Keinerlei Optimierung im Contao CMS
- Kein Caching intern oder extern
- Seiten mit HTTP/1.1 ausgeliefert
- Alle Bilder in Originalgröße
(meistens 1920px, manche auch nur 1100 oder 1200px)
- Keine Pagespeed-Erweiterungen aktiv

testlauf 1: null optimierung

- News-Übersichtsseite mit Archiv
- ca. 85 News online, alle mit Vorschaubild

Performance Scores

PageSpeed Score	YSlow Score
F (18%) ▾	D (62%) ▾

Page Details

Fully Loaded Time	Total Page Size	Requests
2.5s ^	23.8MB ▾	100 ▾

testlauf 2: bildgröße

- Allgemeine Bildregel mit maximaler Auflösung 1280px

Performance Scores

PageSpeed Score	YSlow Score
F (20%) ▾	D (62%) ▾

Page Details

Fully Loaded Time	Total Page Size	Requests
2.4s [▲]	16.9MB ▾	100 ▾

testlauf 3: echte bildregeln

- Gute Bildregeln mit responsiver Anpassung, zugeschnitten auf den Einsatz (News-Übersicht)

Performance Scores

PageSpeed Score	YSlow Score
D (63%) ▾	D (61%) ▾

Page Details

Fully Loaded Time	Total Page Size	Requests
3.7s ▲	1.76MB ▲	100 ▾

testlauf 4: caching aktiv

- Caching in Contao und in der .htaccess-Datei aktiviert
- Pagespeed-Erweiterungen für Contao installiert
- Auslieferung mit HTTP/2

Performance Scores

PageSpeed Score	YSlow Score
A (90%) ^	C (72%) ♦

Page Details

Fully Loaded Time	Total Page Size	Requests
2.1s ^	1.75MB ^	100 v

**Gute Basis-Optimierung
mit kleinem Aufwand**