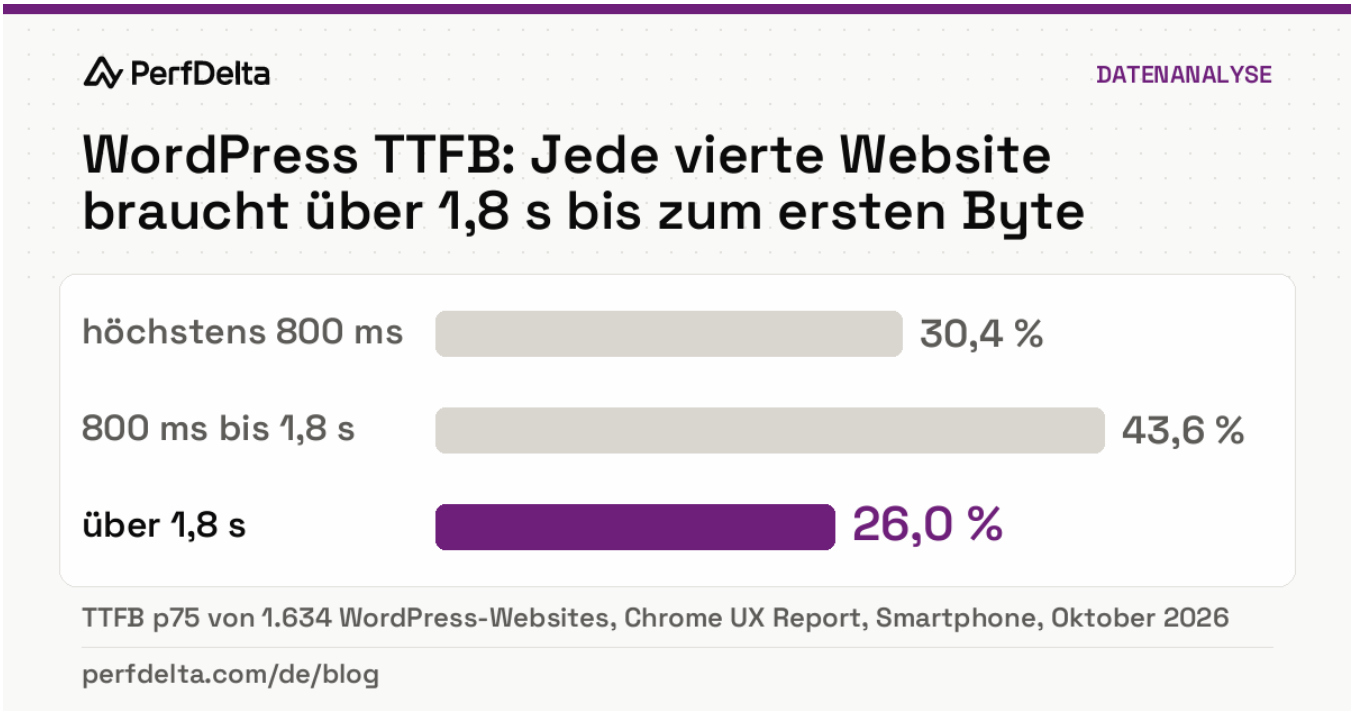
WordPress TTFB: Jede vierte WordPress-Website braucht über 1,8 Sekunden bis zum ersten Byte

Veröffentlicht am 04.11.2026 von Christian Schreiber



Bei langsamen WordPress-Websites fällt der Verdacht meist auf zu viele Plugins und schwere Themes im Browser. Unsere Felddaten zeigen auf eine andere Stelle, den Server. Wir haben die Chrome-Felddaten von 1.634 WordPress-Websites ausgewertet. Bei jeder vierten lag das 75. Perzentil der Time to First Byte (TTFB) über 1,8 Sekunden, ab dieser Grenze bewertet Google den Wert als schlecht. Über alle 11.376 Websites unserer Stichprobe mit TTFB-Daten traf das auf jede zehnte zu.

Bei diesen Websites wartet mindestens jeder vierte Seitenaufruf länger als 1,8 Sekunden auf das erste Byte HTML. Vorher kann nichts von der neuen Seite erscheinen, und keine Bildoptimierung und kein verzögertes Skript holt diese Zeit im selben Ladevorgang wieder auf.

Die TTFB misst die Zeit bis zum ersten Byte des HTML

Die Time to First Byte misst, wie lange es vom Start einer Navigation dauert, bis das erste Byte der Antwort im Browser ankommt. Laut Google gehören dazu Weiterleitungen, ein eventueller Service-Worker-Start, die DNS-Abfrage, Verbindungs- und TLS-Aufbau und die Zeit, bis der Server zu antworten beginnt (Time to First Byte, web.dev, zuletzt aktualisiert im November 2025). Als gut gelten 0,8 Sekunden oder weniger, als schlecht mehr als 1,8 Sekunden.

Das ist wie im Restaurant: Die TTFB ist die Zeit von der Bestellung, bis der erste Teller die Küche verlässt. Kocht die Küche jedes Gericht von Grund auf, hilft auch der schnellste Kellner nicht. Ein Seitencache ist das Gericht, das schon fertig bereitsteht.

Die TTFB ist selbst kein Core Web Vital, sie ist aber der erste Abschnitt des Largest Contentful Paint (LCP). Jede Millisekunde, die der Server braucht, verzögert den LCP. Unser Ratgeber zur Time to First Byte erklärt die Metrik im Detail.

WordPress-Websites antworten langsamer als der Rest der Stichprobe

Die Tabelle vergleicht die WordPress-Websites mit allen Websites unserer Stichprobe. Die Werte beruhen auf dem 75. Perzentil (p75) echter Smartphone-Besuche aus dem Chrome UX Report. Die erste Zeile ist der Median der p75-Werte aller Websites, die übrigen Zeilen zeigen den Anteil der Websites, deren p75 im genannten Bereich liegt:

Felddaten Smartphone	WordPress	alle Websites
TTFB, Median der p75-Werte	1.158 ms	841 ms
gute TTFB (höchstens 0,8 s)	30,4 %	46,8 %
schlechte TTFB (über 1,8 s)	26,0 %	10,4 %
guter LCP (höchstens 2,5 s)	48,4 %	59,0 %
guter INP (höchstens 200 ms)	71,7 %	60,2 %
guter CLS (höchstens 0,1)	82,8 %	76,6 %
alle drei Core Web Vitals gut	34,5 %	36,4 %

Bei Interaktivität (INP) und Layoutstabilität (CLS) schneiden WordPress-Websites besser ab als der Durchschnitt. Beim Laden verlieren sie, und der Rückstand beginnt am Server.

Bei 62 Prozent der langsamen WordPress-Websites dauert die Serverantwort am längsten

CrUX zerlegt den LCP in vier Phasen: die TTFB, die Verzögerung, bis das LCP-Bild zu laden beginnt, seinen Download und die Zeit bis zur Darstellung. Diese Phasen weist CrUX nur für Seitenaufrufe aus, bei denen das LCP-Element ein Bild ist. Für die 829 WordPress-Websites mit einem LCP-p75 über 2,5 Sekunden und Phasendaten haben wir geprüft, welche Phase das höchste p75 hatte:

Phase mit dem höchsten p75	WordPress	alle Websites
Time to First Byte	62,5 %	44,5 %

Phase mit dem höchsten p75	WordPress	alle Websites
Ladeverzögerung	29,4 %	44,0 %
Renderverzögerung	7,2 %	9,9 %
Ladedauer (Download)	0,8 %	1,5 %

Bei diesen Websites lag das p75 der TTFB-Phase bei 1.732 ms (Median über die Websites). Im langsamsten Viertel der Seitenaufrufe verbraucht damit allein die Serverantwort mehr als zwei Drittel der 2,5 Sekunden, die für einen guten LCP zur Verfügung stehen. Über alle Websites teilen sich Ladeverzögerung und Serverantwort die Spitze fast gleichmäßig, wie unsere Auswertung zum langsamen LCP gezeigt hat. Bei WordPress dominiert klar der Server.

Fehlendes Seitencaching ist eine wahrscheinliche Ursache, das Netz erklärt wenig

Die TTFB enthält auch das Netz. Die Daten erlauben eine grobe Prüfung: Die Umlaufzeit (RTT, p75) lag bei WordPress-Websites im Median bei 142 ms, nur 15 ms über den 127 ms aller Websites. Selbst mit drei bis vier Umläufen für DNS, Verbindung und TLS erklärt das nur einen kleinen Teil der 317 ms Abstand im TTFB-Median. Der größte Teil muss aus der Zeit stammen, die der Server zum Antworten braucht.

WordPress baut jede Seite mit PHP und Datenbankabfragen neu auf, solange kein Cache eine fertige Kopie ausliefert. Eine gecachte Seite kommt oft nach wenigen hundert Millisekunden an, eine nicht gecachte kann auf einem ausgelasteten Server Sekunden brauchen. Verfehlt mehr als ein Viertel der Aufrufe den Cache, bestimmt die Antwortzeit ohne Cache das p75. Unsere Daten enthalten keine Cache-Informationen, das ist also eine naheliegende Erklärung, keine gemessene. Typische Gründe für Fehltreffer:

- **Gar kein Seitencache** oder nur ein Cache im Browser statt auf dem Server.
- **URL-Parameter.** Server- und CDN-Caches mit dem Standard-Cache-Schlüssel behandeln `?utm_source=…`, `?gclid=…` oder `?fbclid=…` als neue Seite und bauen sie neu. Besucher aus Anzeigen und Newslettern, oft die wertvollsten, landen damit auf dem langsamsten Weg. Mehrere Caching-Plugins ignorieren diese Parameter bereits.
- **Cookies.** Angemeldete Nutzer, Warenkörbe und manche Consent-Werkzeuge setzen Cookies, bei denen der Cache für den Rest des Besuchs aussetzt.
- **Kurze Cache-Laufzeiten** und komplettes Leeren nach jeder kleinen Änderung, sodass der Cache auf Seiten mit wenig Verkehr selten gefüllt ist.

Weiterleitungen kommen hinzu. Ein Link auf `http://` oder auf den falschen Hostnamen und Klick-Tracking-Weiterleitungen vor einer Landingpage kosten zusätzliche Umläufe, bevor die eigentliche Anfrage beginnt. CrUX rechnet diese Zeit in die TTFB ein.

Sechs Maßnahmen senken die TTFB von WordPress oder ihre Wirkung

1. **HTML aus einem Seitencache ausliefern.** Auf Serverebene (etwa Varnish oder ein nginx-Cache) oder per Caching-Plugin. Das WordPress-Handbuch beschreibt beide Wege und erklärt, warum der Seitencache die größte Wirkung hat (Cache, Advanced Administration Handbook). Die Trefferquote zeigen Antwort-Header wie `x-cache`, `age` oder `cf-cache-status`.
2. **Marketing-Parameter im Cache-Schlüssel ignorieren** und Cookies, die den Cache umgehen, auf die Seiten beschränken, die sie wirklich brauchen, etwa Warenkorb und Kasse.
3. **HTML am Rand des Netzes cachen.** Ein CDN, das auch das HTML nah beim Besucher vorhält, spart bei gecachten Seiten den Weg zum Ursprungsserver und dessen Rechenzeit. Viele CDNs, auch Cloudflare, cachen HTML standardmäßig nicht. Nötig ist eine ausdrückliche Cache-Regel mit Ausnahme für Cookies und einem an WordPress gekoppelten Leeren des Caches. Das zählt besonders, wenn der Server in Europa steht und die Besucher in den USA sitzen oder umgekehrt.
4. **Nicht gecachte Anfragen beschleunigen.** Ein persistenter Objektcache wie Redis oder Memcached hält Datenbankergebnisse zwischen den Anfragen vor (gleiche Quelle). Eine aktuelle PHP-Version und wenige automatisch geladene Optionen verringern die Arbeit je Seite. Query Monitor zeigt langsame Datenbankabfragen und Hooks, ein Profiler wie Xdebug oder SPX zeigt, wo PHP seine Zeit verbringt.
5. **Weiterleitungen entfernen und das Backend sichtbar machen.** Verlinken Sie direkt die endgültige URL und nutzen Sie HSTS für den Wechsel auf HTTPS. Ein `Server-Timing`-Header zeigt im eigenen Real User Monitoring, wie viel der TTFB in PHP steckt und wie viel im Netz.
6. **Die Wartezeit nutzen.** Mit 103 Early Hints kann der Browser Stylesheet und LCP-Bild schon laden, während der Server die Seite noch baut. Speculation Rules laden die nächste Seite vor dem Klick. Seit Version 6.8 lädt WordPress eine verlinkte Seite vorab, sobald ein Besucher sie anzuklicken oder anzutippen beginnt (Eagerness „conservative“, Speculative Loading in 6.8, Make WordPress Core, März 2025). Beim Vorabrendern fällt die Serverantwort schon vor dem Klick an, und die TTFB dieses Seitenaufrufs liegt nahe null. Bei der mittleren WordPress-Website unserer Stichprobe lag der Anteil vorab gerenderter Seitenaufrufe bei 0,0 Prozent.

Ist die TTFB im Griff, folgt als Nächstes die Ladeverzögerung des LCP-Bilds. Der Ratgeber zu WordPress Core Web Vitals behandelt Bilder, Themes und Plugins.

Die TTFB Ihrer WordPress-Website prüfen

Der kostenlose TTFB Test zeigt den Feldwert Ihrer Seite und Ihrer Domain aus dem Chrome UX Report. Der Website Speed Test zerlegt den LCP Ihrer Website und Ihrer Wettbewerber in die vier Phasen. Wie WordPress-Websites insgesamt abschneiden, zeigt die WordPress Pagespeed-Statistik.

Methode und Grenzen dieser Auswertung

- **Daten:** 11.396 Smartphone-Datensätze aus dem Chrome UX Report, je Origin einmal gezählt, Datenzeitraum bis 31. Oktober 2026, abgerufen von PerfDelta. Jeder Datensatz fasst echte Chrome-Besuche aus einem gleitenden Zeitraum von 28 Tagen zusammen. TTFB-Werte lagen für 11.376 Origins vor, davon 1.634 mit WordPress. Anteile guter Werte beziehen sich auf die Origins mit Daten zur jeweiligen Metrik.
- **Plattform:** erkannt am Technologie-Stack der Startseite in unserer jüngsten Messung. Die Felddaten gelten für die gesamte Origin.
- **LCP-Phasen:** nur Origins mit einem LCP-p75 über 2,5 Sekunden, für die CrUX Phasendaten ausweist. Die Phasen erfassen nur Seitenaufrufe mit einem Bild als LCP-Element. Die längste Phase ist die Phase mit dem höchsten p75. Die p75-Werte der vier Phasen stammen aus unterschiedlichen Seitenaufrufen und ergeben zusammen nicht das LCP-p75.
- **Zusammenhang, keine Ursache:** Die Daten beschreiben Websites, die auf WordPress laufen, samt Hosting und Cache-Einrichtung. Sie zeigen nicht, dass WordPress an sich langsam ist. Eine gut gecachte WordPress-Website liefert ihr HTML so schnell aus wie jede andere.
- **Stichprobe:** Die Websites stammen aus Tests und Studien von PerfDelta, größere Marken sind überrepräsentiert. Die Ergebnisse sind nicht repräsentativ für das gesamte Web. Weitere Zahlen finden Sie in den Website Speed Statistiken.