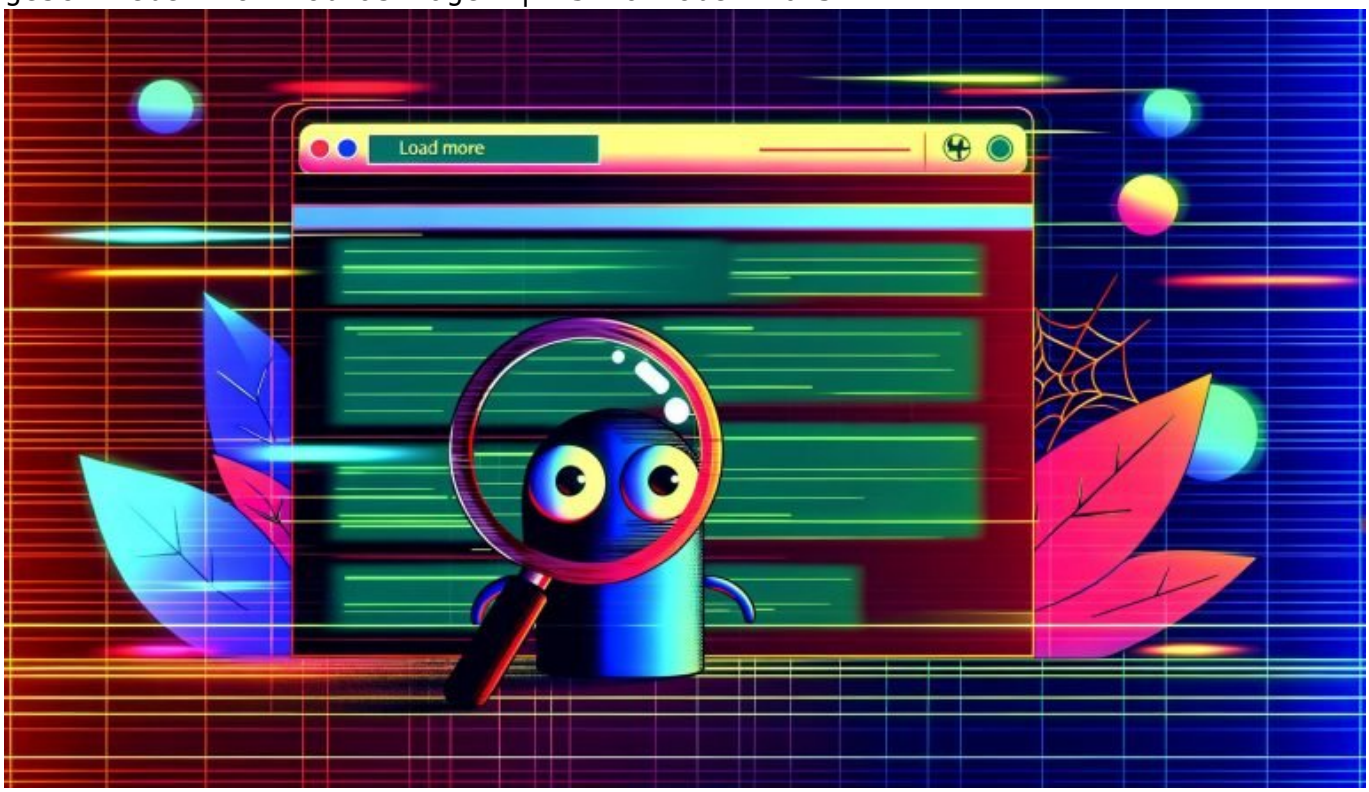
Load More Buttons crawlfähig machen: So geht’s richtig!

Category: SEO & SEM
geschrieben von Tobias Hager | 13. Oktober 2025



Load More Buttons crawlfähig machen: So geht’s richtig!

Du hast den “Load More”-Button auf deiner Seite – und wunderst dich, warum Google deinen genialen Content einfach nicht findet? Willkommen im Club der digitalen Selbstsabotage. Denn was für User cool aussieht, ist für Suchmaschinen oft unsichtbar. In diesem Artikel zerlegen wir gnadenlos, warum 99% aller “Load More”-Implementierungen SEO-technisch ins Leere laufen, zeigen dir, wie es besser geht und liefern dir den ultimativen Guide, damit Google & Co. endlich ALLE deine Inhalte verstehen. Keine Ausreden mehr, ab

heute wird gecrawlt, was das Zeug hält!

- Warum klassische “Load More Buttons” SEO-technisch eine Katastrophe sind
- Wie JavaScript und AJAX den Googlebot regelmäßig austricksen (und dich ins Ranking-Nirwana schicken)
- Was “Crawlability” und “Indexierbarkeit” im Kontext von nachladbaren Inhalten wirklich bedeuten
- Welche technischen Lösungen es gibt: von progressiver Anreicherung bis hin zu “History API” und serverseitigem Rendering
- Die besten Strategien für SEO-freundliche Pagination und Infinite Scroll
- Step-by-Step-Anleitung: So machen Profis ihren Load More Button crawlfähig
- Typische Fehler und wie du sie vermeidest – inklusive Praxisbeispielen
- Welche Tools und Tests dir zeigen, ob deine Lösung wirklich funktioniert
- Warum ohne technische Exzellenz dein “Load More”-Button schlichtweg Traffic killt

Der Load More Button ist das Paradebeispiel für modernen Web-Komfort und gleichzeitig der SEO-GAU in Reinform. Entwickelt für bessere User Experience und endlos scrollbare Listen, zerstört er regelmäßig die Sichtbarkeit von Content, der eigentlich gefunden werden will. Wer das ignoriert, spielt digitales Russisch Roulette: Deine “versteckten” Inhalte sind für Google so sichtbar wie ein Stealth-Bomber bei Nacht. Das Problem ist technischer Natur – und die Lösung? Ebenfalls. In diesem Artikel zerlegen wir bis ins letzte Detail, wie du deinen Load More Button so umbaust, dass er nicht nur für User, sondern auch für Crawler funktioniert. Ohne Bullshit, ohne Ausflüchte – aber mit maximaler technischer Tiefe.

Load More Button und SEO: Warum fast alle Lösungen Crawling-Probleme verursachen

Der Load More Button ist aus modernen Websites und Web Apps nicht mehr wegzudenken. Ob in Produktlisten, Blogs oder Social Feeds – er sorgt für dynamisches Nachladen und damit für eine bessere User Experience. Doch während User begeistert klicken, versteht der Googlebot oft nur Bahnhof. Warum? Weil klassische Load More Buttons fast immer per JavaScript und AJAX Inhalte nachladen, die erst nach dem initialen Seitenaufruf erscheinen. Für Crawler, die keine Interaktionen ausführen, existieren diese Inhalte schlicht nicht.

Der Load More Button wird oft so implementiert, dass er bei Klick per AJAX einen Request an den Server schickt und neue Inhalte per JavaScript in die DOM-Struktur einfügt. Das Problem: Suchmaschinen crawlen Seiten standardmäßig ohne Interaktion. Sie sehen das initiale HTML, aber alles, was erst nach dem Klick kommt, bleibt ihnen verborgen. Das bedeutet im Klartext: Die mühsam produzierten Inhalte hinter deinem Load More Button sind für Google nicht indexierbar – und damit SEO-technisch tot.

Das große Missverständnis: Viele Entwickler verlassen sich darauf, dass Google JavaScript inzwischen “versteht”. In Wahrheit ist das Rendering von dynamisch nachgeladenen Inhalten zwar besser geworden, aber noch lange nicht zuverlässig. Besonders kritisch wird es, wenn Inhalte erst durch User-Interaktion (Klick auf den Load More Button) sichtbar werden. Ohne zusätzliche technische Maßnahmen bleibt der Großteil deines Contents im digitalen Niemandsland.

Wer zudem auf Infinite Scroll setzt – also das automatische Nachladen beim Scrollen – verschärft das Problem sogar noch. Zwar ist die User Experience dann besonders smooth, doch für Crawler ist der zusätzliche Content komplett unsichtbar, solange er nicht über klassische Links erreichbar ist. Fazit: Was für Nutzer bequem ist, kann für die Sichtbarkeit deiner Seite das Todesurteil sein.

Crawlability und Indexierbarkeit: Die SEO-Kernprobleme bei nachladbaren Inhalten

Im Kern geht es bei SEO-freundlichen Load More Buttons um zwei Dinge: “Crawlability” (also die Fähigkeit, dass Suchmaschinen überhaupt alle Inhalte erreichen können) und “Indexierbarkeit” (ob diese Inhalte dann tatsächlich auch in den Google-Index aufgenommen werden). Klingt simpel, ist aber technisch ein Minenfeld, weil herkömmliche Load More Buttons beide Anforderungen missachten.

Crawler wie der Googlebot folgen klassischen HTML-Links, analysieren strukturierte Pagination und erkennen nur Inhalte, die beim initialen Seitenaufruf im HTML vorhanden oder per klassischen Link erreichbar sind. Wenn der Load More Button jedoch nur auf clientseitiges JavaScript setzt und keine URLs nach außen sichtbar macht, steht der Googlebot vor einer Wand. Kein Link, kein Crawl, kein Index.

Ein weiteres Problem: Selbst wenn Google JavaScript-Rendering in einer zweiten Crawling-Welle durchführt, ist das keine Garantie dafür, dass nachgeladene Inhalte indexiert werden. Besonders große oder stark frequentierte Seiten haben oft ein limitiertes Crawl-Budget – Google entscheidet dann, wie tief und wie oft gecrawlt wird. Inhalte, die “versteckt” hinter Interaktionen liegen, werden oft einfach ignoriert.

Deshalb gilt: Wirklich SEO-freundlich ist nur, was ohne User-Interaktion und ohne JavaScript-Rendering zugänglich ist. Alles andere ist bestenfalls “nice to have”, aber niemals eine solide Grundlage für nachhaltige Sichtbarkeit. Wer das nicht beherzigt, verschenkt Reichweite – und zahlt mit Traffic, Umsatz und Relevanz.

Technische Lösungen: Wie du deinen Load More Button crawlfähig machst

Es gibt mehrere technische Ansätze, um das SEO-Desaster beim Load More Button zu verhindern. Keine davon ist trivial – aber alle sind einer AJAX-only-Lösung haushoch überlegen. Die Königsdisziplin: Eine Kombination aus klassischer Pagination, progressivem Enhancement und sauberer URL-Struktur. Hier ein Überblick über die wichtigsten Methoden:

- Klassische Paginierung mit echten URLs: Baue zusätzlich zum Load More Button eine echte, serverseitige Pagination ein. Jede Seite (z.B. /blog/page/2/) ist über einen normalen Link erreichbar. Der Load More Button lädt dann per AJAX genau diese Seiten nach – aber im HTML sind die Links vorhanden.
- Progressives Enhancement: Die Seite ist ohne JavaScript als klassische Pagination navigierbar, mit JavaScript wird daraus ein Load More Button. So bleibt der Content crawlbar, und User profitieren trotzdem von der modernen UX.
- History API: Nutze die HTML5-History-API, um beim Nachladen von Inhalten die URL zu aktualisieren. So bekommt jede nachgeladene Seite eine eigene, crawlbare URL, die auch direkt angesurft werden kann.
- Server-Side Rendering (SSR): Statische HTML-Renderings aller Seitenvarianten, sodass Google auf jeder URL den vollständigen Content sieht – unabhängig von JavaScript.
- Pre-Rendering/Dynamic Rendering: Für Crawler wird immer die vollständige Seite ausgeliefert, während Nutzer die dynamische Variante erleben. Ein Ansatz, der zunehmend kritisch gesehen wird, aber technisch funktioniert.

Die schlechteste aller Lösungen: Einfach hoffen, dass Google schon irgendwie den Content findet. Spoiler: Tut er nicht. Wer clever ist, baut seine Seite so, dass alle Inhalte sowohl für User als auch für Crawler erreichbar sind – und zwar ohne Workarounds oder Black-Hat-Tricks. Alles andere ist digitaler Selbstmord.

Step-by-Step: So machst du deinen Load More Button SEO-freundlich

Genug Theorie, jetzt wird’s praktisch. So gehst du vor, wenn du deinen Load More Button wirklich crawlfähig machen willst – und zwar so, dass Google garantiert alle Inhalte findet:

- 1. Serverseitige Paginierung implementieren
  Stelle sicher, dass jede Seite deiner Liste eine eigene, statische URL hat – z.B. /produkte/page/2/. Diese URLs müssen aufrufbar sein und den vollständigen Content für diese Seite liefern, ohne dass JavaScript ausgeführt werden muss.
- 2. HTML-Pagination sichtbar machen
  Baue klassische “Weiter”- oder “Seite 2”-Links ins HTML ein, die auch ohne JavaScript funktionieren. Diese Links sind essenziell für die Crawlability durch Suchmaschinen.
- 3. Load More Button per JavaScript ergänzen
  Jetzt kommt das progressive Enhancement: Blende den klassischen Paginierungsbereich aus und ersetze ihn durch einen Load More Button, der per AJAX Inhalte nachlädt. Im DOM bleiben die Links trotzdem erhalten, für den Fall, dass JavaScript nicht ausgeführt wird (z.B. beim Googlebot).
- 4. History API nutzen
  Nach jedem Nachladen von Inhalten aktualisiere die URL im Browser mit pushState(), damit jeder Zustand eine eigene URL hat. So können Nutzer und Crawler jede Seite direkt aufrufen.
- 5. Canonical Tags sauber setzen
  Jede Seite der Pagination bekommt ihren eigenen Canonical Tag, der auf ihre eigene URL verweist. Vermeide es, alle Seiten auf die erste Seite zu kanonisieren – das killt die Indexierung aller Folgeseiten.
- 6. Strukturierte Daten anpassen
  Setze strukturierte Daten (Schema.org/Pagination), damit Google die Zusammengehörigkeit der Seiten erkennt. So stärkst du die Indexierbarkeit und das Verständnis der Content-Struktur.
- 7. Testen, testen, testen
  Nutze die Google Search Console (“URL-Prüfung”), Fetch-as-Google, Screaming Frog und den Mobile-Friendly Test, um zu prüfen, ob alle Seiten korrekt gecrawlt und indexiert werden. Ohne Testing ist alles nichts.

Mit dieser Vorgehensweise stellst du sicher, dass dein Load More Button nicht nur für User, sondern auch für Suchmaschinen optimal funktioniert. Wer stattdessen auf reine JavaScript-Lösungen ohne Fallback setzt, verliert im digitalen Wettbewerb – garantiert.

Typische Fehler und wie du sie vermeidest: Praxisbeispiele

aus dem SEO-Alltag

Niemand ist perfekt – aber viele Load More Buttons sind absolute SEO-Totalausfälle. Hier die häufigsten Fehler, die dir garantiert die Indexierung versauen, und wie du sie vermeidest:

- Fehler: Keine echten Links im HTML
  Viele Entwickler verstecken die Pagination komplett hinter JavaScript. Die Folge: Kein Link, kein Crawl, keine Indexierung. Lösung: Immer HTML-Links für jede Seite bereitstellen.
- Fehler: Alle Seiten auf die erste Seite kanonisieren
  Ein häufiger Irrtum: Alle Paginierungsseiten bekommen den gleichen Canonical Tag. Das Ergebnis: Google indexiert nur die erste Seite, der Rest verschwindet im Nichts. Lösung: Jede Seite bekommt ihren eigenen Canonical.
- Fehler: Unendliches Scrollen ohne Pagination-Links
  Infinite Scroll ohne sichtbare Pagination ist ein SEO-GAU. Google sieht nur die erste Seite, alles andere bleibt unsichtbar. Lösung: Immer Pagination-Links zusätzlich anbieten – auch wenn sie für Nutzer versteckt sind.
- Fehler: JavaScript-Fehler blockieren das Rendering
  Wenn beim Nachladen Fehler passieren (z.B. wegen kaputter AJAX-Requests), sieht Google leere Seiten oder fehlerhafte Inhalte. Lösung: Fallback-Mechanismen und sauberes Error-Handling einbauen.
- Fehler: URLs werden nicht mit der History API aktualisiert
  Nachgeladene Inhalte haben keine eigene URL, Nutzer (und Crawler) können keine einzelnen Seiten ansteuern. Lösung: Immer die URL nach jedem Nachladen aktualisieren.

Ein letzter Tipp aus der Praxis: Teste regelmäßig, ob deine Seiten wirklich gecrawlt und indexiert werden. Die besten Lösungen bringen nichts, wenn sie nicht funktionieren – und das erfährst du nur durch konsequentes Monitoring.

Tools und Tests: Wie du sicherstellst, dass dein Load More Button wirklich crawlfähig ist

Du willst wissen, ob deine Lösung wirklich funktioniert? Dann verlasse dich nicht auf dein Bauchgefühl, sondern auf harte Fakten. Mit diesen Tools prüfst du, ob Google deinen nachladbaren Content wirklich sieht:

- Google Search Console: Nutze die “URL-Prüfung” und das “Live-Test”-Tool, um zu sehen, was Googlebot tatsächlich rendert und indexiert.
- Screaming Frog SEO Spider: Simuliere einen Crawler und prüfe, ob alle Seiten und Pagination-Links auffindbar sind.
- Fetch as Google (Rendern und Abrufen): Zeigt dir, wie Google die Seite wirklich sieht – inklusive aller nachgeladenen Inhalte.
- Mobile-Friendly Test: Teste, ob auch in der Mobile-Variante alle Inhalte geladen und angezeigt werden.
- Logfile-Analyse: Prüfe, welche URLs vom Googlebot tatsächlich besucht werden und wo er aussteigt.

Wichtig: Führe diese Tests regelmäßig durch – vor allem nach Updates am Frontend oder bei Änderungen am JavaScript. Was heute funktioniert, kann morgen schon wieder unsichtbar sein.

Fazit: Ohne crawlfähigen Load More Button bleibt dein Content unsichtbar

Der Load More Button ist eine UX-Errungenschaft – aber technisch eine tickende SEO-Zeitbombe, wenn man ihn falsch implementiert. Wer seine Inhalte nur nach Klick oder durch endloses Scrollen sichtbar macht, verschenkt das größte Kapital im Online-Marketing: Sichtbarkeit. Die Lösung ist kein Hexenwerk, aber sie erfordert technisches Know-how, Disziplin und konsequentes Testing.

Wer jetzt noch glaubt, dass Google schon “irgendwie” alle Inhalte findet, hat im digitalen Wettbewerb längst verloren. Der crawlfähige Load More Button ist kein Nice-to-have, sondern Pflicht für alle, die im Web relevant bleiben wollen. Also: Mach deine Inhalte sichtbar – für User UND Crawler. Alles andere ist digitale Selbstsabotage.