ideewirwa.de: Übersicht für Crawler, KI-Werkzeuge, llms.txt, IndexNow und vereinfachte HTML-Ansicht

28.10.2026 · @Sebastian Ebner · Quelle: KI Werkzeug Cloude

> In den Befehlen und Pfaden stehen Platzhalter: `name` = EOTK-Projektname, `benutzer` = Benutzername beim Webhosting, `domain.de` = eigene Domain, `SCHLUESSEL` = eigener IndexNow-Schlüssel. Die Code-Ausschnitte aus der `index.html` zeigen den tatsächlichen Stand.

Ausgangslage

Die Startseite `index.html` ist eine Einzelseiten-Anwendung. Der sichtbare Inhalt wird in einem Rahmen (`<iframe>`) aus Unterseiten wie `ex/startseite.html` geladen, das Menü baut `ideewirwa.js` per JavaScript auf. Die Übersicht aller Themengebiete stand bisher in einem `<noscript>`-Block.

Ein Test am 28.10.2026 hat gezeigt: Alle vier Adressen (http/https, mit und ohne www) laden ohne Fehler oder Blockierung. Das Abrufwerkzeug liefert aber nur Titel und Meta-Angaben. Den `<noscript>`-Block wirft es beim Umwandeln in Text weg, ebenso Skripte und iframes. Viele solcher Werkzeuge arbeiten so, weil `<noscript>` in einem Browser mit JavaScript nie angezeigt wird.

Änderungen in der index.html

Es sind fünf Stellen. Texte und Links der Übersicht sowie alles andere bleiben unverändert.

1. Im <head>, direkt vor <title>

```
<script>document.documentElement.classList.add('js');</script><style>.js #uebersicht{disp
lay:none}html:not(.js) #seiteninhalt{display:none}html:not(.js),html:not(.js) body{overfl
ow:auto!important;height:auto!important}html:not(.js) #uebersicht{display:block;position:
relative;z-index:2147483647;max-width:60em;margin:1em auto;padding:1em 1.2em;color:#111;l
ine-height:1.45}html:not(.js) #uebersicht a{color:#1a4f8a}html:not(.js) #uebersicht p{fon
t-size:100%}html:not(.js) #uebersicht h2{font-size:1.4em}html:not(.js) #uebersicht h3{fon
t-size:1.1em;margin:.8em 0 .2em}</style>
```

Das Skript setzt sofort die Klasse `js` am `<html>`-Tag. Mit JavaScript blendet `.js #uebersicht{display:none}` die Übersicht aus, bevor sie gezeichnet wird. Alle Regeln mit `html:not(.js)` greifen nur ohne JavaScript und sorgen für die einfache Darstellung (siehe „Vereinfachte HTML-Ansicht“).

2. Nach <meta name="author">

```
<meta name="llms" content="https://www.ideewirwa.de/llms.txt">
```

Ein Hinweis auf die llms.txt im Kopfbereich. Kein Standard, schadet aber nicht. Die volle Adresse ist nötig, weil Meta-Angaben nicht wie Links ergänzt werden. Über die Onion-Adresse schreibt EOTK sie automatisch auf die Onion-Adresse um.

3. <noscript> wird zu <section>

```
<section id="uebersicht" aria-label="Uebersicht der Themengebiete">
```

Die Übersicht ist jetzt normales HTML und wird auch von Werkzeugen gelesen, die `<noscript>` verwerfen. Zwischendurch stand hier `<nav>`. Das wurde durch `<section>` ersetzt, weil viele Abrufwerkzeuge Navigationsbereiche als Menü absichtlich wegfiltern. Die Links bleiben relativ, damit Besucher auf der Onion-Adresse und den Spiegelservern nicht auf die Hauptdomain springen.

4. Zwischen und dem Ende der Übersicht

```
<p><a href="llms.txt">Kurzfassung fuer KI-Werkzeuge (llms.txt)</a> · <a href="llms-full.t
xt">Volltext fuer KI-Werkzeuge (llms-full.txt)</a> · <a href="sitemap.xml">Sitemap (sitem
ap.xml)</a></p>
```

Zuerst stand hier nur der Link auf die `llms.txt`. In einem zweiten Schritt kamen die Links auf `llms-full.txt` und `sitemap.xml` dazu. Die IndexNow-Schlüsseldatei wird bewusst nicht verlinkt.

5. </noscript> wird zu </section>

```
</section>
```

Zusätzlich wurde der erklärende Kommentar „TEIL B“ an die neue Lösung angepasst.

Ergebnis

Getestet mit Chromium (Playwright) und der echten `ideewirwa.css`, mit und ohne JavaScript. In beiden Fällen sind alle 28 Links vorhanden. Zusätzlich im Tor Browser auf dem Handy geprüft.

Wer	Was er sieht
Normaler Browser (JavaScript an)	Die Website wie bisher. Die Übersicht ist unsichtbar.
Browser ohne JavaScript, z. B. Tor Browser „Am sichersten“	Die Übersicht mit allen Links direkt auf dem Hintergrundmuster. Der Rahmen mit der Startseite ist ausgeblendet. Ohne JavaScript ist die Übersicht der einzige Weg zu den Unterseiten.
Crawler und KI-Werkzeuge	Überschrift, Beschreibungen und alle Links als normales HTML.

Eine Variante, die die Übersicht auch ohne JavaScript per CSS ausblendet, wurde verworfen. Tor-Nutzer kämen sonst nicht mehr zu den Unterseiten, und Google kann versteckten Text geringer gewichten.

Vereinfachte HTML-Ansicht (ohne JavaScript)

Das Problem

Im Tor Browser auf der Stufe „Am sichersten“ ist JavaScript abgeschaltet, und einige Schriftarten, Symbole und Bilder sind gesperrt. Zu sehen waren nur das Hintergrundmuster und der Text „logo_sitego1“, von der Übersicht nichts.

- **Übersicht verdeckt:** In der `ideewirwa.css` steht `#seiteninhalt{position:fixed;top:0;left:0;width:100%;height:100%;…;z-index:1}`. Der Rahmen liegt fest über dem ganzen Bildschirm. Mit JavaScript ist das gewollt. Ohne JavaScript verdeckt er die Übersicht und fängt auch die Fingertipps ab.
- **„logo_sitego1“:** Das ist der Ersatztext des Logos `present/logo_sitego1.svg`, das auf dieser Stufe nicht geladen wird.

Die Lösung

Regel	Wirkung ohne JavaScript
`html:not(.js) #seiteninhalt{display:none}`	Blendet den bildschirmfüllenden Rahmen aus. Er zeigt ohne JavaScript ohnehin nur das gesperrte Logo.
`overflow:auto!important; height:auto!important` für `html` und `body`	Die Seite lässt sich scrollen.
`#uebersicht{display:block; position:relative; z-index:…; max-width:60em; margin; padding; color; line-height}`	Die Übersicht erscheint im Vordergrund, mittig, mit Abstand zum Rand und gut lesbar.
`#uebersicht a{color:#1a4f8a}`	Links in Blau statt Grau aus der CSS der Website.
`p{font-size:100%}, h2{font-size:1.4em}, h3{font-size:1.1em;…}`	Kleinere Schrift. Die CSS der Website vergrößert Absätze auf 135 %.

In einer ersten Fassung lag hinter der Übersicht ein weißer Kasten (`background:rgba(255,255,255,.94)`). Er wurde entfernt, damit der originale Hintergrund (`present/background.webp`) durchscheint. Die CSS der Website muss nicht geändert werden.

Wenn die alte Darstellung bleibt

Der Tor Browser behält Seiten im Zwischenspeicher. „Quelltext anzeigen“ (`view-source:`) lädt die Datei frisch und zeigt, was der Server wirklich liefert. Die angezeigte Seite kann trotzdem noch alt sein. Abhilfe: im Tor Browser Menü → „Beenden“, neu starten, Seite neu öffnen.

Rückweg

Falls Menü, Navigation, Vorlesen oder Übersetzen nach der Änderung Probleme machen: `<section …>` und `</section>` wieder gegen `<noscript>` und `</noscript>` tauschen. Dann verhält sich die Seite wie vorher. Die Zeilen im `<head>` stören in diesem Fall nicht.

Was das Abrufwerkzeug von Claude sieht

- **Startseite:** nur Titel, bekannte Meta-Angaben und die kanonische Adresse. Den Inhalt des `<body>`, alle `<link>`-Elemente außer `canonical` und unbekannte Meta-Namen wie `llms` zeigt es nicht an. Das liegt am Werkzeug, nicht an der Seite. Per `curl` war geprüft, dass Übersicht und Meta-Zeile online sind.
- **llms.txt:** wird vollständig gelesen, Umlaute korrekt. Über die vollen Adressen darin ließ sich z. B. `ex/steinmetzarbeiten.html` direkt abrufen.
- **llms-full.txt:** lädt mit korrekten Umlauten, das Werkzeug liest aber nur den Anfang, etwa bis „Persönliches → Geschichtswelle 2“.
- **Vorübergehende Umleitung:** Für einige Minuten bekam das Werkzeug eine `302`-Umleitung von https auf http. Im Zugriffsprotokoll der Domain tauchten diese Anfragen nicht auf. Wahrscheinlich hat ein Schutz bei Linevast nach vielen schnellen Abrufen kurz gebremst. Danach lud die Seite wieder normal. Die Domain hat nur IPv4.

Fazit: Für KI-Werkzeuge ist die `llms.txt` unter ihrer festen Adresse der verlässliche Weg. Übersicht und Meta-Zeile sind Zusatzhinweise. Suchmaschinen lesen den rohen HTML-Code. Laut Zugriffsprotokoll hat Google die Startseite samt `schema.js` vollständig abgerufen.

EOTK (Onion-Adresse)

- EOTK schreibt alle Adressen der Seite auf die Onion-Adresse um, auch `canonical`, `og:url` und die neue Meta-Zeile. Tor-Nutzer bleiben so im Tor-Netz.
- Der Zwischenspeicher von EOTK wurde mit `./eotk recreate name` geleert und in der Projekt-Datei `name.tconf` vor der `hardmap`-Zeile abgeschaltet:

```
set nginx_cache_seconds 0

./eotk config name.tconf
./eotk restart name
```

Änderungen erscheinen über Tor damit sofort. Nachteil: Jeder Aufruf wird direkt beim Webspace abgeholt. `60` wäre ein Mittelweg. Der genaue Name der Einstellung wurde nicht in der EOTK-Dokumentation nachgeprüft. Der weiße Kasten lag nicht an diesem Cache, sondern am Zwischenspeicher des Tor Browsers.

Versionsstand v490

Alle Seiten wurden auf die Version 490 gebracht, damit Browser und Service Worker die neuen Dateien laden statt der alten aus dem Zwischenspeicher.

- `quellcodes/js/sw.js`: `const CACHE_NAME = 'ideewirwa-v490';` (vorher v485).
- Alle 34 HTML-Dateien: Das Skript-Tag lädt `ideewirwa.js?v=490` und `sprachsteuerung.js?v=490`. Die meisten standen auf 485. Angeglichen wurden auch `fragestellungen.html` (vorher 482) und die vier Chat-Archive (vorher 481).
- `quellen_sources.html`: zusätzlich 7 Stellen im Text und in Linktexten, die die aktuelle Version nennen.
- `ex/impressum.html`: zusätzlich eine Stelle im Datenschutztext.

- Nicht geändert: Versionsnummern im *Gesprächsinhalt* der Chat-Archive (67 Stellen mit v480, v1 und v24). Sonst wären die Archive verfälscht.

Die neue Datei llms.txt

Was sie ist

Eine Textdatei im Markdown-Format, die einem KI-Werkzeug in Kurzform sagt, was die Website ist, welche Unterseiten es gibt und unter welcher vollständigen Adresse sie liegen. Die Konvention wurde 2024 von Jeremy Howard (Answer.AI) vorgeschlagen. Sie ist kein offizieller Standard, sondern freiwillig.

Wo sie liegt

```
Server:   /home/benutzer/domains/domain.de/public_html/llms.txt
Adresse:  https://www.domain.de/llms.txt
```

Sie liegt im selben Ordner wie die `index.html`. Über EOTK ist sie automatisch auch auf der Onion-Adresse erreichbar.

Aufbau

```
# Titel                    ← Name der Website
> Zusammenfassung          ← ein, zwei Sätze, worum es geht
Hinweistext                ← z. B. „Inhalte liegen in den Unterseiten“
## Abschnitt               ← Gruppen wie „Werke“, „Service“
- [Name](Adresse): Info    ← je eine Unterseite
## Optional                ← Unwichtigeres, darf übersprungen werden
```

Inhalt

- Titel „Ideewirwa's Welt“ und eine Zusammenfassung in einem Satz.
- Der Hinweis, dass die Startseite die Inhalte per iframe lädt und man die Unterseiten direkt abrufen soll.
- Alle 25 Unterseiten mit vollständigen Adressen (`https://www.ideewirwa.de/…`), in Gruppen: Über Sebastian Ebner, Werke, Projekte und Schriftstücke, Service, Optional.
- Unter „Optional“ zusätzlich die drei Feeds (RSS2, Atom, JSON), die Tor-Adresse und ein Link auf die `llms-full.txt`.
- Kurze, sachliche Beschreibungen aus den Texten der Website. Wohnort, Anschrift und Telefonnummer stehen bewusst nicht drin.

Wie viel sie bringt

Laut einer Auswertung von Serverlogs vom Mai 2026 erhielten 97 % aller llms.txt-Dateien keine einzige Anfrage. OpenAI und Anthropic verweisen für ihre Crawler auf `robots.txt`. Google sagt, die Datei habe weder positiven noch negativen Einfluss auf das Ranking. Nützlich ist sie, wenn man einem KI-Werkzeug die Adresse direkt gibt, und bei KI-Agenten, die Websites selbstständig durchsuchen. Sie kostet nichts und schadet nicht.

Die neue Datei llms-full.txt

Die Ergänzung zur llms.txt: Sie enthält den **Text aller Unterseiten** in einer Datei, damit ein KI-Werkzeug nicht jede Seite einzeln abrufen muss.

- **Wo:** `/home/benutzer/domains/domain.de/public_html/llms-full.txt`, erreichbar unter `https://www.domain.de/llms-full.txt`.
- **Aufbau:** Kurzbeschreibung und Inhaltsverzeichnis mit allen Adressen, dann je Unterseite eine Überschrift, ihre Adresse und der Text.
- **Erzeugt** aus den aktuellen v490-Dateien mit Python (BeautifulSoup). Entfernt sind Skripte, Knöpfe, Formulare und versteckte Teile. Bilder erscheinen nur über ihren Alternativtext als `[Bild: …]`. Skript-Vorlagen von den Seiten stehen als Codeblöcke drin, mit ihren Platzhaltern.
- **Größe:** rund 3,9 MB, davon etwa 3,5 MB „Bereich Bauen“. Da die meisten KI-Werkzeuge nur einen Teil so großer Dateien lesen, steht „Bauen“ am Ende. So kommen die kleineren Seiten auf jeden Fall an.
- **Weggelassen:** die Startseite, die nur das Eingangsbild enthält.
- **Pflege:** Die Datei aktualisiert sich nicht selbst. Nach Änderungen an Unterseiten muss sie neu erzeugt werden.

Ergänzend: sitemap.xml und robots.txt

Für Suchmaschinen wirkt eine `sitemap.xml` stärker als die llms.txt. Sie enthält die Startseite und alle 25 Unterseiten (26 Adressen, gültiges XML) und liegt ebenfalls im Hauptordner. In der `robots.txt` wird am Ende ergänzt:

```
Sitemap: https://www.domain.de/sitemap.xml
```

Die Sitemap ist zusätzlich in der Übersicht der `index.html` verlinkt. Danach die Sitemap in der Google Search Console und in den Bing Webmaster Tools einreichen. Am einfachsten zuerst die Domain in der Google Search Console bestätigen und die Sitemap einreichen, dann in den Bing Webmaster Tools „Aus Google Search Console importieren“ wählen.

IndexNow

Was es ist

Ein Protokoll, mit dem eine Website Suchmaschinen **sofort** meldet, dass Seiten neu oder geändert sind, statt zu warten, bis ihr Crawler zufällig vorbeikommt. Gestartet im Oktober 2021 von Microsoft (Bing) und Yandex. Eine Meldung an `api.indexnow.org` wird an alle Teilnehmer weitergegeben.

Wer mitmacht

Anbieter	Bemerkung
Bing (Microsoft)	Mitgründer, seit 2021. Wichtig, weil Copilot, DuckDuckGo, Ecosia und die ChatGPT-Suche auf den Bing-Index zugreifen.
Yandex	Mitgründer, seit 2021
Seznam.cz	Mitautor der Spezifikation
Naver	seit Juli 2023
Yep (Ahrefs)	seit 2023 (nicht aus der offiziellen Liste geprüft)

Google nimmt nicht teil. Für Google bleibt die `sitemap.xml` in der Search Console der Weg.

Die Schlüsseldatei

```
Datei:    SCHLUESSEL.txt   (z. B. 32 Zeichen 0-9 und a-f)
Inhalt:   SCHLUESSEL       (genau der Dateiname ohne .txt)
Server:   /home/benutzer/domains/domain.de/public_html/
Adresse:  https://www.domain.de/SCHLUESSEL.txt
```

Sie ist der Eigentumsnachweis. Die Suchmaschine ruft die Datei auf und vergleicht den Schlüssel mit dem der Meldung. Nur wenn beide übereinstimmen, wird die Meldung angenommen. Der Schlüssel ist kein Geheimnis, er liegt öffentlich auf dem Server. Wer ihn kennt, kann höchstens die eigenen Adressen zur Prüfung melden. Die Datei muss unter genau diesem Namen liegen bleiben.

Anmeldung

Für IndexNow gibt es keine Registrierung und kein Konto, die Schlüsseldatei ersetzt die Anmeldung. Wer sehen möchte, welche Meldungen angekommen sind, findet das in den Bing Webmaster Tools im Bereich „IndexNow“ (auch Yandex Webmaster hat einen Bericht).

Das Sendeskript indexnow-senden.sh

Meldet alle 26 Adressen aus der Sitemap plus `llms.txt` und `llms-full.txt` mit einem Aufruf (JSON per POST an `https://api.indexnow.org/indexnow`). Es gehört nicht auf den Webspace, sondern nach Termux oder auf den V-Server.

```
sh indexnow-senden.sh
IndexNow Antwort: 200   ← oder 202 = angenommen, 403 = Schlüsseldatei nicht gefunden
```

Nach Änderungen an Seiten erneut ausführen.

Hochladen

Diese Dateien kommen in den Hauptordner `/home/benutzer/domains/domain.de/public_html/`:

Datei	Status
`index.html`	geändert (Übersicht als <section>, drei Links, Meta-Zeile `llms`, vereinfachte HTML-Ansicht)
`llms.txt`	neu
`llms-full.txt`	neu
`sitemap.xml`	neu
`SCHLUESSEL.txt`	neu (IndexNow-Schlüssel)

Dazu die übrigen v490-Dateien an ihren gewohnten Orten (`ex/`, `media/aufzeichnung/`, `media/fragestellungen/`, `media/hauptordner/`, `quellcodes/js/sw.js`). Wurde v490 schon hochgeladen, bevor die `index.html` die drei Links bekam, die `sw.js` auf v491 hochzählen.

Nach dem Hochladen

1. `sw.js`-Version hochzählen (v491 bzw. höher), damit der Service Worker die neue `index.html` ausliefert.
2. Prüfen, was online ist: `curl -s https://www.domain.de/ | grep -o '<meta name="llms" [^>]*>\|<section id="uebersicht"'`. Beide Zeilen müssen erscheinen.
3. Im Tor Browser auf „Am sichersten“ testen. Bei alter Darstellung den Tor Browser beenden und neu starten.
4. Website durchklicken: Menü, Navigation, Vorlesen, Übersetzen.
5. `https://www.domain.de/llms.txt` im Browser öffnen. Werden Umlaute falsch angezeigt, in der `.htaccess` `AddCharset UTF-8 .txt` ergänzen.
6. `https://www.domain.de/SCHLUESSEL.txt` öffnen. Dort muss der Schlüssel stehen.
7. `sh indexnow-senden.sh` in Termux oder auf dem V-Server ausführen.
8. In der `robots.txt` die Zeile `Sitemap: https://www.domain.de/sitemap.xml` ergänzen.
9. Der Datei-Wächter meldet die neuen Dateien. Das ist erwartet.
10. Neue Unterseiten künftig eintragen in: `<ul id="seitenliste">` und Übersicht der `index.html`, `sitemap.xml`, `llms.txt`. Danach die `llms-full.txt` neu erzeugen und das Sendeskript erneut ausführen.

Quellen

1. Jeremy Howard (Answer.AI): The /llms.txt file. Vorschlag vom 3.9.2024. https://llmstxt.org/
2. Luis Rijo, PPC Land: llms.txt adoption rises 8.8x but 97% of files get zero AI requests. 2.7.2026. https://ppc.land/llms-txt-adoption-rises-8-8x-but-97-of-files-get-zero-ai-requests/
3. Google Search Central: Build and submit a sitemap. https://developers.google.com/search/docs/guides/create-URLs
4. sitemaps.org: Sitemaps XML format (Protokoll). https://www.sitemaps.org/protocol.html
5. MDN Web Docs: <noscript>. https://developer.mozilla.org/de/docs/Web/HTML/Element/noscript
6. Wikipedia: IndexNow. https://en.wikipedia.org/wiki/IndexNow
7. IndexNow.org: Documentation for search engines. https://www.indexnow.org/searchengines

8. IndexNow.org: Protokoll und Dokumentation. https://www.indexnow.org/documentation
9. Tor Project: Security levels (Sicherheitsstufen). https://support.torproject.org/tor-browser/features/security-levels/
10. MDN Web Docs: <section>. https://developer.mozilla.org/de/docs/Web/HTML/Element/section · :not(). https://developer.mozilla.org/de/docs/Web/CSS/:not
11. EOTK (Enterprise Onion Toolkit). https://github.com/alecmuffett/eotk
12. Bing Webmaster Tools: https://www.bing.com/webmasters · Google Search Console: https://search.google.com/search-console
13. Eigene Tests am 28.10.2026: Abruf von http(s)://(www.)ideewirwa.de mit dem Abrufwerkzeug von Claude; Darstellungstest der geänderten index.html mit Chromium (Playwright) mit und ohne JavaScript (28 Links); XML-Prüfung der sitemap.xml mit Python; Erzeugung der llms-full.txt mit Python (BeautifulSoup) aus den v490-Dateien; JSON- und Syntaxprüfung des Sendeskripts; Vergleich aller v490-Dateien mit den Originalen (nur Versionsnummern geändert, Dateigrößen gleich); Darstellung der hochgeladenen index.html mit der echten ideewirwa.css ohne JavaScript; Abruf von llms.txt, llms-full.txt und ex/steinmetzarbeiten.html; Auswertung des Zugriffsprotokolls 11:30–11:34 Uhr; curl-Prüfungen und Bildschirmaufnahmen aus dem Tor Browser von Sebastian Ebner.