WORDPRESS MEETUP HAMBURG #140 — 29. SEPTEMBER 2026, AKTUALISIERT AM 2. OKTOBER 2026

KI und WordPress

Chancen und Risiken von MCP — was passiert, wenn die KI nicht mehr nur redet, sondern Zugriff bekommt.

Timo Wessels

Der Abend in fünfzehn Schritten

01 Die KI sitzt nicht mehr nur im Chatfenster

02 Das ist alles schon da

03 Was jetzt plötzlich geht

04 Ich bin kein Entwickler

05 Und dann willst du etwas ändern

06 Wer eine Verbindung schafft, schafft ein Risiko

07 „Goodbye WordPress" — die Gegenseite

08 Weg von WordPress — und wieder zurück

09 Statisch oder generiert

10 Die falsche Leiche

11 Die Mittellage

12 Und was heißt das für uns?

13 Für Montag

14 Eure Runde

15 Zum Nachlesen

Schon mal scannen

Die Folien liegen online. Wer den Code jetzt scannt, kann nachher alles in Ruhe nachlesen, statt mitzuschreiben.

timowessels.de/vortraege/wp-meetup-140

Alle fünfzehn Kapitel und die Quellenliste mit jedem Link, der heute vorkommt.

KAPITEL 1

Die KI sitzt nicht mehr nur im Chatfenster

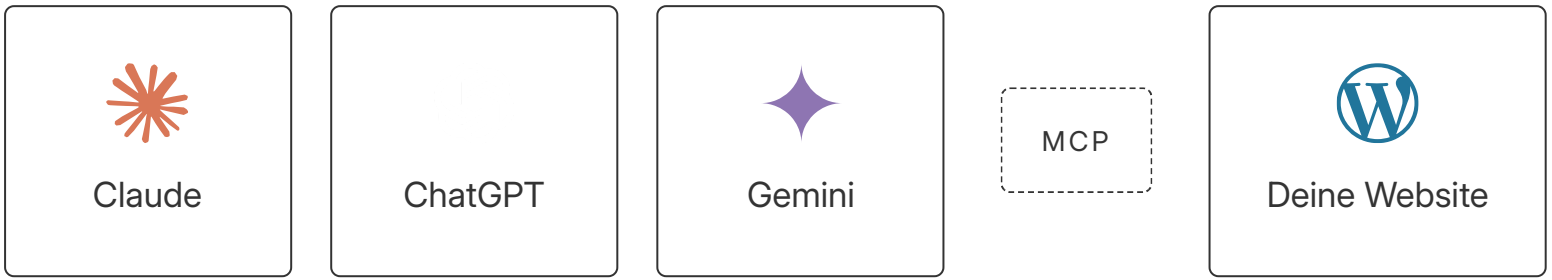
Bis vor kurzem hat die KI geredet. Jetzt darf sie anfassen.

- Gestern: Du fragst, sie antwortet, du kopierst es selbst hinein.
- Heute: Du verbindest sie einmal — und sie macht es selbst.
- MCP ist der genormte Stecker dafür. Das USB-C für KI.
- Kein Produkt, keine Firma, nichts zum Kaufen — eine Vereinbarung.
- Offen für alle: Claude, ChatGPT und Gemini hängen am selben Stecker.

> Der Unterschied ist nicht das Tempo. Der Unterschied ist der Zugriff.

Ein Stecker, viele Modelle

Das ist der ganze Trick an MCP: Nicht jede KI braucht ihre eigene Anbindung an deine Website. Es gibt eine Vereinbarung, an die sich alle halten.



Wer die Vereinbarung einmal umsetzt, kann das Modell wechseln, ohne noch einmal anzufangen. Genau deshalb hat sich der Standard so schnell durchgesetzt — er nimmt allen Beteiligten Arbeit ab.

KAPITEL 2

Das ist alles schon da

Keine Zukunftsmusik und kein Bastelkram aus einem GitHub-Keller. Das steckt inzwischen in WordPress selbst.

- WordPress 7.0: ein zentraler Ort für KI-Zugänge unter Einstellungen → Connectors.
- WordPress denkt nicht selbst. Es verwaltet nur die Verbindung nach draußen.
- Ein Plugin darf diesen Zugang erst benutzen, wenn du es ausdrücklich freigibst.
- Der MCP Adapter macht die eigene Seite anschlussfähig für KI.
- Meta Box baut damit Felder, Etch baut im Builder — die Hersteller ziehen nach.

> Vorher musste man WordPress dazu überreden. Heute ist die Tür eingebaut.

Die vier Bausteine, die es schon gibt

Nichts davon muss jemand erfinden. Alles davon kann man heute installieren.

Connectors API — seit WordPress 7.0 im Kern

Ein zentraler Ort für die Zugangsschlüssel zu KI-Diensten, unter Einstellungen → Connectors. Vorher hat jedes Plugin sein eigenes Feld dafür gebaut. Und ein Plugin darf den Schlüssel erst benutzen, wenn du es freigibst.

MCP Adapter — die Tür nach draußen

Macht die eigene WordPress-Seite anschlussfähig für KI-Agenten. Das ist der offizielle Weg aus dem WordPress-Projekt selbst; das ältere Plugin von Automattic, das damit angefangen hat, ist eingestellt.

Meta Box — die KI baut die Datenstruktur

Über die neuen „Abilities" kann ein KI-Agent Beitragstypen, Taxonomien, Feldgruppen und Feldinhalte anlegen und befüllen. Also nicht nur Texte schreiben, sondern das Gerüst bauen, in dem die Inhalte später liegen.

Etch — die KI baut im Builder mit

Der Connector hängt einen KI-Agenten an den geöffneten Builder-Tab: Seiten anlegen, Styles setzen, Komponenten bauen. Was du auf dem Bildschirm siehst, verändert sich, während der Agent arbeitet.

KAPITEL 3

Was jetzt plötzlich geht

Was früher ein Nachmittag war, ist heute eine Kaffeepause.

- Ein Bild oder ein Stück HTML hinein — fünf bis zehn Minuten später steht die Seite.
- Gut für eine Landingpage, ein kleines Projekt, den ersten Entwurf.
- Sehr gut für Fleißarbeit über hunderte von Beiträgen.
- Der wiederholbare Teil unserer Arbeit wird gerade billig.
- Und das war selten der Teil, den wir gerne gemacht haben.

> „Steht" ist allerdings nicht dasselbe wie „fertig".

KAPITEL 4

Ich bin kein Entwickler

Der interessanteste Beweis für diesen Abend steht nicht auf YouTube.

- Kein PHP, kein Python, kein Studium — und rund dreißig eigene Werkzeuge.
- Ich habe nicht gelernt zu programmieren. Ich habe gelernt zu beschreiben und zu prüfen.
- Prüfen können bekommt man nur durch Jahre. Hier ist Erfahrung ein Vorteil.
- Gewachsen ist das über Monate. Und vieles davon ist auch wieder weggeflogen.
- Jedes Werkzeug ist eine Verpflichtung: Pflege, Dokumentation, Verständnis.

> Wenn ich das kann, liegt die Grenze nicht mehr da, wo ihr sie vermutet.

Was daraus geworden ist

Angefangen hat es mit einer Suche über die eigenen Unterlagen. Inzwischen ist es ein Kreislauf, der sich selbst kontrolliert.

Eine Suche über den eigenen Bestand

Der Anfang von allem: eine Volltextsuche über alles, was ich je aufgeschrieben habe. Damit muss die KI nicht raten, was in meinem Betrieb gilt — sie kann nachschlagen. Das allein hat mehr verändert als jedes cleverere Modell.

Die Regeln stehen in der Datei selbst

Jede Datei trägt oben eine Markierung: wozu sie gehört, welcher Vertrag für sie gilt, wie sie geschrieben sein muss. Wer sie öffnet — ich oder eine KI — weiß im ersten Absatz, was hier erlaubt ist. Regeln, die woanders liegen, werden nicht gelesen.

Werkzeuge, die Wissen hereinholen

Websites herunterladen, den Inhalt aus dem Layout herausschälen, Code durchsuchbar machen, PDFs auslesen. Rund zwei Dutzend solcher Werkzeuge sind es inzwischen. Keins davon ist klug — sie sorgen nur dafür, dass etwas zum Nachschlagen da ist.

Ein Fakt, eine Stelle, ein Prüfer

> Geschrieben wird an genau einer Stelle. Generatoren verteilen den Fakt dorthin, wo er gebraucht wird, und Prüfskripte rechnen nach und melden sich, wenn eine Kopie abweicht. Das ist der Teil, den man nicht sieht — und der Grund, warum es nach zwei Jahren noch zusammenpasst.

KAPITEL 5

Und dann willst du etwas ändern

Die ersten achtzig Prozent kosten zehn Minuten. Die letzten zwanzig kosten weiterhin einen Tag.

- Die KI kennt WordPress. Sie kennt deine Website nicht.
- Framework, Klassennamen, Verschachtelung, Styleguide — alles offene Fragen.
- Ohne Regeln baut sie jedes Mal anders. Und alles funktioniert für sich.
- Ein Handwerker, der jeden Morgen sein Gedächtnis verloren hat.
- Eine Seite geht immer. Zehn Websites über zwei Jahre — wer hält das zusammen?

> Das Tempo verschiebt die Arbeit. Es schafft sie nicht ab.

Ein echtes Beispiel, diese Woche

Alexander Schell hat die Website seiner Agentur komplett mit KI bauen lassen — Design-System, Figma-Bibliothek, Komponenten in Drupal — und öffentlich aufgeschrieben, wie es lief. Er ist der beste Zeuge dafür, weil er dafür ist.

Was er berichtet

Faktor 8 schneller als vergleichbare Projekte, Faktor 4 günstiger, weil eine Person die Rollen von Frontend, Backend, UX und Test übernommen hat. Das ist keine kleine Zahl, und ich glaube sie ihm.

Wo er gebraucht wurde

Nach eigener Aussage nicht beim Bauen, sondern beim Freigeben und Korrigieren. Seine Beispielsätze: „Nein, die Farbe kommt aus dem Token." „Nein, das ist die dritte Karte, die anderen fliegen raus." „Vergleich das mit dem Figma-Layout, die Abstände stimmen nicht."

Was im ausgelieferten Markup steht

Auf der Startseite stehen 792 Einzelwerte direkt in den Klassen, 129 verschiedene: zwölf Schriftgrößen von 10 bis 52 Pixel und zwölf Farbwerte, von Hand hingeschrieben statt aus dem Token gezogen. Die Überschriften wirken gesund — bis man die Reihenfolge liest: ein h2 steht vor dem h1, danach springt es auf h4. Und das Gold auf dem Dunkelblau erreicht 3,34:1, wird aber an 47 Stellen in 10 bis 13 Pixel gesetzt, wo 4,5:1 gefordert sind. Gezählt am 8. September 2026; er sagt selbst, die Seite sei live und noch nicht fertig.

Warum das kein Vorwurf ist

Es ist die Bestätigung. Genau diese Arbeit — entscheiden, dass eine Farbe gilt und die anderen drei nicht — bleibt am Menschen hängen. Sie wird nicht weniger, wenn das Bauen schneller wird. Sie wird sichtbarer.

> Das ist kein Fehler im Prompt. Das ist die Natur der Sache: KI produziert plausible Varianten. Ein Design-System braucht das Gegenteil — eine Entscheidung, die überall gilt.
>
> Alexander Schell, Activate & TAXINIO

> Ich kann diese Demos nicht mehr sehen: Schau, was ich gebaut habe! Wenn ein Mensch es nicht bearbeiten und anpassen kann, ist es nutzlos. Der Kunde will keinen Prompt schreiben, um die Tagesangebote zu ändern. Er will auf den Text klicken und ihn einfach ändern.
>
> Gen Herres, Anphira — WordPress-Entwicklerin und Barrierefreiheits-Expertin. Ergänzt am 2. Oktober 2026

KAPITEL 6

Wer eine Verbindung schafft, schafft ein Risiko

Nicht das, was die KI heute tut, ist der Maßstab. Sondern das, was sie tun könnte.

- Entscheidend ist, wozu ein System in der Lage ist — nicht, was du gerade getippt hast.
- Ein Anruf beim Handwerker oder der Wohnungsschlüssel: das ist der Unterschied.
- Kundendaten im Spiel? Dann Auftragsverarbeitungsvertrag nach Artikel 28 DSGVO.
- Die Trennlinie läuft zwischen Privat- und Geschäftstarif, nicht zwischen gratis und bezahlt.
- Und jede neue Verbindung ist auch ein neuer Weg hinein.
- Und die Standortfrage: US-Anbieter, europäische Alternative — oder ein Modell auf dem eigenen Rechner.

> Erst die Verträge, dann der Stecker. Nicht umgekehrt.

> „Wir haben doch noch nichts offengelegt" ist kein Datenschutz-Argument. Es geht darum, wozu ein System in der Lage ist und wer Zugriff hat — nicht darum, was du gerade getippt hast.
>
> Kathy Zant, sinngemäß

KAPITEL 7

„Goodbye WordPress" — die Gegenseite

Wenn ich die Gegenseite nicht selbst vortrage, sitzt sie trotzdem im Raum.
Nur unausgesprochen.

- Über tausend Websites, zehn Jahre WordPress und Elementor — seit Februar 2026 nicht mehr angefasst.
- Neunzig Immobilien-Einheiten: Wochen eingeplant, in zwanzig Minuten gestanden.
- Er kauft keine Plugins mehr. Er baut sie passgenau, ohne die tausend Einstellungen.
- Keine Plugins, kein Login, keine Angriffsfläche — sein Sicherheitsargument, umgedreht.
- Aber: nie mit einem Prompt, eher mit zweihundert. Und designen lässt er die KI nicht.

> Das ist kein Zuschauer. Das ist einer von uns, der ausgestiegen ist.

> Die meisten kleinen Unternehmen haben WordPress nie wirklich gebraucht. WordPress war einfach für uns, die Entwickler, um es für sie aufzusetzen.
>
> Emanuel, „I Create Your Site"

KAPITEL 8

Weg von WordPress — und wieder zurück

Der Wechsel scheitert selten an der Technik. Er scheitert an der Frage, wer die Seite in einem halben Jahr pflegt.

- Für eine einzelne Seite braucht es kein CMS. Zugegeben.
- Die Frage ist nicht, was schneller baut, sondern wer es danach pflegt.
- Der Kunde will die Öffnungszeiten ändern: Textdatei? Git? Branch? Prompt?
- Der Kompromiss, den man 2026 überall sieht: WordPress hinten, Astro vorne.
- Behalten wird ausgerechnet das Backend — der angeblich altmodische Teil.
- Und man muss sich nicht entscheiden: headless, oder als Git-Repo nach Vercel ausgeliefert.

> Man wirft nicht WordPress weg. Man wirft das Frontend weg.

KAPITEL 9

Statisch oder generiert

Zwei Arten, wie eine Seite entsteht — und der Unterschied entscheidet, wie verlässlich eine KI daran arbeiten kann.

- Statisch: Das HTML existiert bereits. Es ist das Ergebnis.
- Generiert: Das HTML entsteht erst beim Aufruf — aus Datenbank, Theme, Plugins, Filtern und Hooks.
- Im zweiten Fall schreibst du nicht das Ergebnis, sondern etwas, das das Ergebnis herstellt.
- Für eine KI heißt das: einmal kann sie prüfen, was sie geschrieben hat — einmal nicht.
- Dasselbe Modell, sehr unterschiedliche Verlässlichkeit. Nicht wegen der KI, sondern wegen der Entfernung zum Ergebnis.

> Was man nicht ansehen kann, bevor es existiert, kann man auch nicht prüfen.

Was jede Seite kostet

Kein Lager gewinnt. Beide zahlen, nur an unterschiedlichen Stellen — und du entscheidest pro Projekt, welche Rechnung dir lieber ist.

Statisch — dafür

Nichts läuft, also kann nichts abstürzen. Kein PHP an der Front, keine Datenbank im Netz, keine Updates am Wochenende. Was du siehst, ist ausgeliefert — du kannst es lesen, prüfen und versionieren. Und es ist absurd billig zu hosten.

Statisch — dagegen

Jede Änderung braucht jemanden mit Zugang zu den Dateien. Keine Rollen, keine Freigaben, kein Redakteur. Und es wächst schlecht: siehe die Treppe auf der nächsten Karte.

Generiert — dafür

Rollen, Rechte, Freigaben, Suche, Formulare, Mehrsprachigkeit — alles da, ohne dass jemand es baut. Der Kunde ändert selbst, so oft er will. Genau dafür wurde ein CMS erfunden, und darin ist es unschlagbar.

Generiert — dagegen

Du schreibst nie das Ergebnis, sondern die Maschine dahinter. Ein Update, ein Plugin, ein Filter — und die Ausgabe ändert sich, ohne dass du etwas angefasst hast. Mehr Fläche, mehr Pflege, mehr, das kaputtgehen kann.

Die Treppe

Es fängt immer gleich an: eine Seite, statisch, fertig. Und dann kommt der Kunde mit dem nächsten Wunsch.

> Eine Seite.
>
> Reicht völlig. Statisch ist genau richtig.

> Eine Kontaktseite dazu.
>
> Jetzt brauchst du ein Formular.

> Ein Formular.
>
> Jetzt brauchst du einen Dienst, der Mails verschickt.

> Ein Mail-Dienst.
>
> Jetzt brauchst du einen Webhook, den du nicht ganz verstehst.

> Ein Video einbetten.
>
> Jetzt brauchst du eine Einwilligung.

> Eine Einwilligung.
>
> Jetzt brauchst du ein Cookie-Banner.

> Fünf Schritte von „statisch reicht" zu einem CMS, das du selbst gebaut hast — nur schlechter.

KAPITEL 10

Die falsche Leiche

Es gibt einen Toten, ja. Aber alle beugen sich über die falsche Leiche.

- Inhalte erstellen und Inhalte verwalten sind zwei völlig verschiedene Dinge.
- Wer darf ändern? Wer hat freigegeben? Welche Fassung gilt?
- Wo überall wird der Inhalt verwendet — und in wie vielen Sprachen?
- Und wer ist verantwortlich für das, was die KI veröffentlicht hat?
- Je billiger das Erzeugen wird, desto wichtiger wird das Verwalten. Nicht unwichtiger.

> Der Kopierer hat die Registratur nicht abgeschafft. Er hat sie erst nötig gemacht.

> Für manche Websites braucht man kein CMS. Aber Inhalte erstellen und Inhalte verwalten sind zwei völlig verschiedene Dinge.
>
> Mathias Bolt Lesniak

KAPITEL 11

Die Mittellage

Die Frage lautet nicht „KI oder WordPress". Sie lautet: Wer darf die Seite in zwei Jahren ändern — und womit?

- Er schafft das CMS nicht ab. Er baut es neu, für einen Kunden, allein.
- WordPress läuft überall: Wird ein Hoster zu teuer, ziehst du um. Mit allem.
- Und wenn du morgen ausfällst, findet dein Kunde jemanden, der es kennt.
- Der Normalfall ist der Mittelweg: Der Kunde macht Text und Bilder, wir den Rest.
- Bricks-Komponenten als Blöcke, Etch, PHP-Blöcke im Core — Vorlagen mit Leitplanken.

> Die KI baut die Bausteine. WordPress hält den Rahmen, die Rechte und die Übergabe.

Es geht nicht um das Framework

Welches Werkzeug gewinnt, ist die falsche Frage. Es gibt drei, die zählen — und die stellt man am besten, bevor man sich für irgendetwas entscheidet.

Wartbarkeit: Kann es jemand anderes ändern?

Nicht „kann ich es ändern". Sondern: Kann es jemand ändern, der nicht dabei war, in einem Jahr, ohne mich anzurufen? Namen im Markup und Werte im Token-System sind keine Geschmacksfrage — sie sind die Antwort auf genau diese Frage.

Skalierbarkeit: Kostet die fünfzigste Seite so viel wie die erste?

Wenn jede neue Seite wieder Entscheidungen über Abstände, Farben und Varianten verlangt, skaliert es nicht — dann wird es mit jeder Seite teurer. Wenn die Bausteine stehen, ist eine neue Seite nur noch: Inhalt einfüllen.

Zugang: Wer kommt überhaupt rein?

Damit meine ich nicht Barrierefreiheit im WCAG-Sinn, sondern ganz praktisch: Wer darf und kann etwas veröffentlichen? Eine Person mit Git-Zugang und einem Deploy-Prozess — oder jeder im Team, der sich einloggen kann? Das entscheidet, ob eine Website lebt oder einfriert.

Und genau das ist das Angebot

> Einmal richtig aufsetzen: Templates, Komponenten, Rollen, Rechte. Danach schreiben und veröffentlichen, so oft ihr wollt, ohne dass jemand Code anfasst. Der Aufwand liegt am Anfang und nicht bei jedem einzelnen Beitrag — das ist der ganze Handel, und er ist zwanzig Jahre alt.

Zwei Arten, dieselbe Karte zu schreiben

Links steht, was tatsächlich ausgeliefert wird. Rechts dasselbe Element, nur mit Namen für seine Teile. Beides funktioniert. Nur eines davon kann man in einem Jahr noch lesen.

OHNE NAMEN — WAS DIE KI GEBAUT HAT

```
<div class="flex h-full flex-col justify-start gap-3.5">
  <div class="flex size-10 shrink-0 items-center
    justify-center rounded-full bg-gold/[0.18] text-gold">
  <span class="flex w-full flex-col before:absolute
    before:inset-0 before:content-['']">
    <h3 class="font-sans text-[19px] leading-6
      font-semibold text-white">Bessere Qualität</h3>
    <span class="mt-3.5 font-body text-sm
      leading-[23px] text-white/70">…</span>
```

MIT NAMEN — DERSELBE BAUSTEIN

```
<article class="feature-card">
  <div class="feature-card__icon">…</div>
  <h3 class="feature-card__title">Bessere Qualität</h3>
```

```
    <p class="feature-card__text">…</p>
  </article>
```

> Ein Framework wie ACSS nimmt dir die Entscheidung ab: konsistente Abstände, konsistente Farben, geprüfte Kontraste — dafür sehen die Seiten einander ähnlich, bis hin zu den immer gleichen Icons. Utility-Klassen nehmen dir gar nichts ab: maximale Freiheit, und hinterher weiß niemand, was ein Element eigentlich ist. Die Mitte ist unspektakulär: eigene Namen für die eigenen Bausteine, Werte aus dem Token-System. Dann sagt das Markup, was es ist, und die Werte sagen, wie es aussieht.

KAPITEL 12

Und was heißt das für uns?

Die unbequeme Frage ist nicht, ob die KI unsere Arbeit kann.

- Acht Stunden gegen eine — der Kunde sieht nur das Ergebnis, nicht den Weg.
- Wenn das Bauen billig wird: Wofür wird dann eigentlich bezahlt?
- Vermutlich für das, was danach kommt: Haftung, Pflege, Übergabe, Struktur.
- Nicht jedes Werkzeug lernen. Aber verstehen, was es anrichtet.
- Erfahrung, wie Dinge über Jahre schieflaufen, kann man nicht prompten.

> Ist die Website noch unser Produkt — oder inzwischen der einfachste Teil davon?

Was es kostet — und woran es hängt

Über Tempo redet jeder. Über die Rechnung und die Abhängigkeit redet selten jemand, dabei entscheidet die darüber, ob das in drei Jahren noch trägt.

Der Preis ist heute bekannt, morgen nicht

Ich zahle rund 200 Euro im Monat für das Abo, mit dem ich arbeite. Für mich rechnet sich das deutlich. Aber ich habe keinen Vertrag darüber, dass es das nächstes Jahr auch noch tut: Preise steigen, Tarife werden umgebaut, Kontingente gekürzt. Wer seine Kalkulation darauf stützt, kalkuliert auf fremdem Grund.

Das Modell unter dir wird ausgetauscht

Was heute zuverlässig funktioniert, kann sich nach einem Modellwechsel anders verhalten. Wer seine Arbeitsweise auf ein bestimmtes Verhalten gebaut hat, merkt das als Erster. Deshalb liegt bei mir das Wissen im System und nicht im Verlauf eines Chatfensters — ein System übersteht den Wechsel, ein Gesprächsverlauf nicht.

Und es hängt an deinem Rechner

Werkzeuge, Zugänge, Konfiguration liegen bei mir lokal. Geht die Platte kaputt, ist erst einmal alles weg. Also Backups — und zwar nicht nur von der Website, sondern von dem, was du dir selbst gebaut hast. Genau der Teil, an den niemand denkt, weil er nach Werkzeug aussieht und nicht nach Arbeit.

Ohne Leitplanken zerlegt eine KI eine Website

> Wenn ein Agent Schreibrechte hat und niemand sagt, wo Schluss ist, dann ist eine missverstandene Anweisung ein gelöschter Beitragstyp. Testsystem, eigener Benutzer, wenig Rechte, vorher ein Backup. Das ist kein Misstrauen — das ist derselbe Reflex wie vor einem großen Plugin-Update.

Seit dem Vortrag

Ergänzt am 2. Oktober 2026. Die Meldung, die die erste Karte oben vorhersagt, kam am Morgen nach dem Abend. Dazu drei Stimmen auf die Frage, wofür bezahlt wird, wenn das Bauen billig ist.

OpenAI: 500 Dollar, und der 200-Dollar-Tarif halbiert

Auf der DevDay am 29. September 2026 hat OpenAI einen Pro-Tarif für 500 Dollar im Monat vorgestellt. Der 200-Dollar-Tarif kostet weiter 200 Dollar, enthält in Codex aber nur noch etwa das Zehnfache von Plus statt des Zwanzigfachen, und 100 statt 200 Nachrichten pro Woche mit dem stärksten Modell. Bestehende Kunden behalten das alte Kontingent bis zum 29. Oktober. Preise steigen, Tarife werden umgebaut, Kontingente gekürzt: in einer Woche alle drei.

Was 200 Dollar eigentlich wert sind

Jordan Urbs hat sein Claude-Abo für 200 Dollar gekündigt, nicht weil es schlecht war, sondern weil es zu gut war. An zwei Tagen hat er über 700 Millionen Tokens verbraucht. Zu öffentlichen API-Preisen wären das selbst mit Caching rund 2.000 Dollar gewesen. Seine Folgerung: Entweder sind die API-Margen absurd, oder Risikokapital bezahlt unsere Nutzung, bis wir abhängig sind. — Jordan Urbs, YouTube

Der Code ist das Billigste am Projekt

> „Es geht nicht mehr darum, wer die Software bauen kann. Das ist mittlerweile ein Rohstoff, den dir KI für 20 $ im Monat liefert. Es geht darum, wer das Vertrauen der Nutzer gewinnt.“ — Robin Oehler, Webdesigner, auf LinkedIn

Was sich nicht kopieren lässt

> „Sie können deine Community nicht klonen, nicht die Beziehungen zu deinen Kunden, nicht das System dahinter und nicht die Daten, die dir gehören. Und vor allem nicht das Vertrauen der Menschen, die deinen Geschmack schätzen.“ — Luke Carter, YouTube

KAPITEL 13

Für Montag

Konkrete, gefahrlose erste Schritte — auch für alle, die das noch nie gemacht haben.

- Erst am Testsystem. Nie zuerst an der Seite eines Kunden.
- Nie mit deinem Administrator-Zugang: eigener Benutzer, wenig Rechte, hinterher löschen.
- Keine echten Kundendaten. Testinhalte reichen für alles, was du lernen willst.
- Schreib auf, was du verbunden hast. In einem Jahr weiß es sonst niemand mehr.
- Beim Kunden gilt: erst die Verträge, dann der Stecker.

> Der Maßstab ist, wozu der Zugang in der Lage wäre — nicht, was du gerade damit tust.

KAPITEL 14

Eure Runde

Kein Fazit. Fertige Antworten hat hier gerade niemand.

- Wer hat eine KI schon einmal an eine echte Website gelassen?
- Und wer hat danach kontrolliert, was sie geändert hat?
- Würdet ihr einer KI die Zugangsdaten zu einer Kundenwebsite geben?
- Wo genau liegt eure Grenze — und warum liegt sie ausgerechnet dort?
- Was würdet ihr niemals aus der Hand geben?

> Ich bin da tief drin. Genau das ist mein blinder Fleck — deshalb frage ich euch.

ANHANG

Zum Nachlesen

Nichts an diesem Abend müsst ihr mir glauben. Hier steht, woher es kommt.

- Der Standard selbst und die beiden offiziellen WordPress-Ankündigungen.
- Die Werkzeuge, über die wir geredet haben — Meta Box, Etch, Bricks.
- Das Video des Aussteigers und die Seite, die mit KI gebaut wurde.
- Der Artikel über Erstellen gegen Verwalten.
- Und die zwei Adressen zum Thema Sicherheit und Datenschutz.

> Alles anklickbar. Wenn ihr etwas live sehen wollt, sagt es — dann öffnen wir es.

Zum Nachlesen

Alles, was heute vorkam — mit Adresse. Wer etwas davon live sehen will: sagt Bescheid, wir öffnen es.

Model Context Protocol — die offizielle Doku

Der Standard selbst, samt der USB-C-Analogie, die ich geklaut habe.

Connectors API in WordPress 7.0

Die Ankündigung aus dem Core-Team: ein Ort für alle KI-Zugänge.

Der MCP Adapter für WordPress

Wie die eigene Seite anschlussfähig wird — der offizielle Weg.

Meta Box Abilities

Die KI legt Beitragstypen, Feldgruppen und Feldinhalte an.

Etch Connector

Der Agent arbeitet im geöffneten Builder-Tab mit.

„Goodbye WordPress." — das Video

Achtzehn Minuten, in denen ein Agenturinhaber begründet, warum er ausgestiegen ist.

Activate Media — mit KI gebaut, öffentlich dokumentiert

Die Seite aus Kapitel 5. Rechtsklick, Seitenquelltext anzeigen — dann sieht man, worüber wir geredet haben.

Mathias Bolt Lesniak — „The Website Builder Is Dead, Long Live Content Management"

Der Artikel hinter der falschen Leiche.

Content Security Policy: der Report-Only-Modus

Melden statt blockieren — der einzige vernünftige Einstieg.

KI und Datenschutz: Auftragsverarbeitung nach Artikel 28

Was ein AV-Vertrag verlangt und warum Privat-Tarife dafür ausscheiden.

Anchor Hosting — der Blog von Austin Ginder

Sicherheit nachvollziehbar aufgeschrieben, inklusive seines Vorgehens mit KI.

Bricks: Komponenten als Blöcke

Im Builder gestalten, im Block-Editor bearbeiten — mit Leitplanken.

PHP-Blöcke seit WordPress 7.0

Unser Meetup-Thema aus dem Juni, hier als Nachschlagewerk.

„WordPress is a prison" — die Debatte auf LinkedIn

Genau dieser Streit, öffentlich und diese Woche. Der Post ist die eine Seite, die Kommentare sind die andere — und die interessanteren.

Diese Präsentation

Kommt nach dem Abend mit der Nachlese auf die Meetup-Seite.

OpenAI halbiert den 200-Dollar-Tarif und startet einen für 500 Dollar

Ergänzt am 2. Oktober 2026. Die Meldung zur DevDay vom 29. September, am Morgen nach dem Vortrag.

Jordan Urbs: Why I Cancelled My Claude Code Subscription

Ergänzt am 2. Oktober 2026. Was ein Abo kosten würde, wenn es zu API-Preisen abgerechnet würde. Englisch.

Luke Carter: AI Can Copy Your Website. It Can't Copy This.

Ergänzt am 2. Oktober 2026. Was bleibt, wenn jede Website kopierbar ist: Geschmack und Vertrauen. Englisch.

Gen Herres: Easy A11y Guide

Ergänzt am 2. Oktober 2026. Barrierefreiheit verständlich erklärt, von der Entwicklerin, die heute repariert, was KI gebaut hat. Englisch.

Robin Oehler: Die KI-Welt hat uns eine gigantische Lüge verkauft

Ergänzt am 2. Oktober 2026. Warum Code heute das Billigste am Projekt ist und Vertrauen das Teuerste.

Nimm den Vortrag mit

Alles, was heute auf der Leinwand stand, liegt hier — inklusive dieser Quellenliste zum Anklicken.

timowessels.de/vortraege/wp-meetup-140

Die Seite bleibt online. Wenn ihr in vier Wochen wissen wollt, wie dieses eine Plugin hieß: hier steht es.