

Healthy AI First, Folge 2: Wohin die Zeit verschwindet

Quellen und Belege zur Folge

Wenn KI den Code schreibt, verschwindet die Arbeit nicht. Sie wandert: vom Schreiben zum Erklären, Prüfen und Zusammenfügen. Die Folge verfolgt eine Änderung über die ganze Strecke vom Ticket bis zum Kunden, zeigt mit Zahlen aus Befragungen und Telemetrie, warum mehr Code ein schlechter Maßstab für Erfolg ist, warum sich vor dem Engpass Prüfung ein Stapel bildet und was "fast richtig" kostet. Der Werkzeug-Tipp zeigt, wie man Durchlaufzeiten aus Git und GitHub abliest, statt sie zu schätzen.

Alle Links wurden am 17.10.2026 geprüft.

Was wir in dieser Folge erwähnen

Willkommen

- **Aussage:** In Folge 1 stand Daniel für erfahrene Entwickler, die sich mit KI schneller fühlten und gemessen langsamer waren.
Quelle: Becker, J., Rush, N., Barnes, E. & Rein, D. (METR, 2025): Measuring the Impact of Early-2025 AI on Experienced Open-Source Developer Productivity. Preprint.
Link: <https://arxiv.org/abs/2507.09089>
Einordnung: Randomisierte Studie, aber nur 16 Personen und Werkzeuge von Anfang 2025 (Details in Folge 1).

Miriams Statusbericht

- **Aussage:** Eine Befragung von über dreihundert Wissensarbeitern zu fast tausend Situationen zeigt: Kritisches Denken wandert zum Überprüfen, Einbauen und Steuern; mehr Vertrauen in die KI geht mit weniger kritischem Denken einher, mehr Selbstvertrauen mit mehr.
Quelle: Lee, H.-P., Sarkar, A., Tankelevitch, L., Drosos, I., Rintel, S., Banks, R. & Wilson, N. (Microsoft Research und Carnegie Mellon, 2025): The Impact of Generative AI on Critical Thinking: Self-Reported Reductions in Cognitive Effort and Confidence Effects From a Survey of Knowledge Workers. CHI 2025.
Link: <https://doi.org/10.1145/3706598.3713778> (PDF: https://advait.org/files/lee_2025_ai_critical_thinking_survey.pdf)
Einordnung: 319 Personen, 936 Beispiele, Selbstauskunft. Misst berichteten Denkaufwand, nicht Denkfähigkeit, und zeigt Zusammenhänge, keine Ursachen.

Ein Sechstel des Tages

- **Aussage:** Entwickler verbringen nach eigener Angabe nur rund 16 % ihrer Zeit mit Programmieren; 68 % sparen mit KI mehr als zehn Stunden pro Woche, 50 % verlieren ähnlich viel durch Reibung in der Organisation.
Quelle: Atlassian (2025): State of Developer Experience Report 2025.
Link: <https://www.atlassian.com/blog/developer/developer-experience-report-2025>
Einordnung: Herstellerbefragung von 3.500 Entwicklern und Führungskräften in sechs Ländern; Selbstauskunft.

- **Aussage:** Bei Anthropic kann mehr als die Hälfte der Befragten höchstens ein Fünftel ihrer Arbeit vollständig an die KI abgeben; ein Engineer schätzt, dass über 70 % seiner Arbeit inzwischen Prüfen ist; es gibt die Sorge, dass Fähigkeiten verkümmern, die man zum Beaufsichtigen braucht.
Quelle: Anthropic (02.12.2025): How AI is transforming work at Anthropic.
Link: <https://www.anthropic.com/research/how-ai-is-transforming-work-at-anthropic>
Einordnung: Interne Befragung von 132 Personen plus 53 Interviews bei einem KI-Hersteller; selektive Umgebung, kein Kontrollvergleich. Die 70 % sind eine Einzelstimme.

Der zweite Halbsatz

- **Aussage:** Googles Chef sagte im Oktober 2024, mehr als ein Viertel des neuen Codes bei Google werde von KI erzeugt und dann von Entwicklern geprüft und angenommen.
Quelle: Alphabet (29.10.2024): Q3 earnings call, Remarks von CEO Sundar Pichai.
Link: <https://blog.google/company-news/inside-google/message-ceo/alphabet-earnings-q3-2024/>
Einordnung: Unternehmensaussage, keine unabhängige Messung.
- **Aussage:** Fehlen vorhandene Projektbausteine im Kontext, kann ein Agent sie übersehen und eine Funktion unnötig neu schreiben; jede Sitzung beginnt mit einem frischen Kontext, der aus Nutzerangaben, automatisch geladenen Projektnotizen, Verlauf und Werkzeugergebnissen entsteht. Eine gezielte Suche vorher senkt das Risiko.
Quelle: Anthropic: Claude Code Dokumentation, How Claude remembers your project.
Link: <https://code.claude.com/docs/en/memory>
Einordnung: Herstellerdokumentation, technische Erklärung; dass doppelte Funktionen daher entstehen, ist ein plausibler Mechanismus, kein gemessener Befund.
- **Aussage:** Auswertungen großer Codebestände zeigen mehr kopierten und weniger aufgeräumten Code; dass KI die Ursache ist, ist nicht belegt.
Quelle: GitClear (2025): AI Copilot Code Quality: 2025 Data Suggests 4x Growth in Code Clones.
Link: https://www.gitclear.com/ai_assistant_code_quality_2025_research
Einordnung: 211 Millionen geänderte Zeilen (2020 bis 2024); Herstellerdaten, Korrelation.
- **Aussage:** Wird eine Kennzahl zum Ziel, taugt sie nicht mehr als Messung.
Quelle: Goodhart, C. (1975), in der bekannten Kurzform von Strathern, M. (1997): "Improving ratings": audit in the British University system. European Review 5(3), 305-321.
Link: <https://doi.org/10.1017/S1062798700002660> (Überblick: https://en.wikipedia.org/wiki/Goodhart%27s_law)
Einordnung: Erfahrungssatz aus Ökonomie und Anthropologie, kein Experiment. Ausführlich in Folge 8.
- **Aussage:** Eine bessere Zahl als Codemenge ist die Zeit vom Ticket bis zum Kunden.
Quelle: DORA: Software delivery performance metrics (u. a. change lead time).
Link: <https://dora.dev/guides/dora-metrics/>
Einordnung: Die DORA-Kennzahl misst vom Commit bis zur Produktion; "Ticket bis Kunde" ist bewusst weiter gefasst und eine Empfehlung der Folge.

Der Stapel vor dem Engpass

- **Aussage:** Teams mit hoher KI-Nutzung übernehmen fast doppelt so viele Änderungen (+98 %), die Änderungen werden viel größer (+154 %), die erfasste Reviewzeit ist fast doppelt so hoch (+91 %, ein Gruppenwert, keine Messung jeder einzelnen aktiven Prüfung), und auf Unternehmensebene

zeigt sich kein nennenswerter Zusammenhang.

Quelle: Faros AI (23.07.2025): The AI Productivity Paradox Report 2025.

Link: <https://www.faros.ai/blog/ai-software-engineering>

Einordnung: Telemetrie von über 10.000 Entwicklern in 1.255 Teams; Herstellerdaten, Korrelation.

- **Aussage:** Ein System ist so schnell wie seine engste Stelle; davor stapelt sich Halbfertiges.

Quelle: Goldratt, E. M. & Cox, J. (1984): The Goal. Theorie der Engpässe.

Link: https://en.wikipedia.org/wiki/Theory_of_constraints (Überblick)

Einordnung: Managementkonzept aus der Fertigung; Übertragung auf Softwareprüfung ist ein Denkmodell.

Fast richtig

- **Aussage:** Langes konzentriertes Prüfen ermüdet; man übersieht dann eher etwas.

Quelle: Warm, J. S., Parasuraman, R. & Matthews, G. (2008): Vigilance Requires Hard Mental Work and Is Stressful. Human Factors 50(3), 433-441.

Link: <https://doi.org/10.1518/001872008X312152>

Einordnung: Übersicht zur Vigilanzforschung; auf Code Reviews übertragen, nicht dort gemessen.

- **Aussage:** In einer der größten Entwicklerumfragen nennen etwa zwei Drittel "fast richtig, aber nicht ganz" als größten Ärger mit KI.

Quelle: Stack Overflow (2025): Developer Survey 2025, Abschnitt KI.

Link: <https://survey.stackoverflow.co/2025/ai>

Einordnung: 66 % der Antwortenden auf diese Frage; große, aber selbst ausgewählte Stichprobe.

- **Aussage:** Häufigster Grund für Verwirrung in Code Reviews ist eine fehlende Begründung der Änderung.

Quelle: Ebert, F., Castor, F., Novielli, N. & Serebrenik, A. (2019): Confusion in Code Reviews: Reasons, Impacts, and Coping Strategies. SANER 2019.

Link: <https://doi.org/10.1109/SANER.2019.8668024>

Einordnung: Befragung plus Analyse von Reviewkommentaren; qualitative Evidenz. Die "drei Sätze über jedem Änderungsvorschlag" sind eine daraus abgeleitete Empfehlung.

Die Uhr im Werkzeug

- **Aussage:** Git speichert zu jedem Commit einen Zeitstempel, die GitHub-CLI gibt pro Pull Request aus, wann er geöffnet, geprüft und gemergt wurde, und ein Coding Agent kann beides selbst aufrufen.

Quelle: Git-, GitHub-CLI- und Claude-Code-Dokumentation (siehe Werkzeug-Tipp).

Einordnung: Offizielle Dokumentation. Die Zeitpunkte ergeben Kalenderabstände (Durchlaufzeiten), keine Trennung von aktiver Arbeit, Liegezeit, Nächten und Überarbeitung; ein eingereichtes Review ist nicht der Beginn des Lesens.

Eure Zeit, aufgeteilt

- **Aussage:** Der Versionsverlauf eines Dokuments zeigt oft, wer wann daran gearbeitet hat.

Quelle: Google Docs Editors Help: Find what's changed in a file.

Link: <https://support.google.com/docs/answer/190843>

Einordnung: Herstellerdokumentation; nicht jedes Programm bietet das.

Werkzeug-Tipp: Durchlaufzeiten ablesen statt schätzen

Beispiel-Prompt: "Zeig mir für meine letzten zehn Pull Requests die Zeitabstände zwischen erstem Commit, Eröffnung, erstem eingereichten Review und Merge." Das sind Durchlaufzeiten, keine reinen Wartezeiten. Aktive Arbeit und tatsächliche Liegezeit müsst ihr zusätzlich erfassen.

- `gh pr list --author @me --state merged --limit 10 --json number,title,createdAt,mergeDate,reviews,commits`: pro Pull Request Erstellungs- und Merge-Zeitpunkt, Reviews und Commits. https://cli.github.com/manual/gh_pr_list
- `git log --format='%h %aI %s'`: Commits mit Autorendatum; ein erster Commit ist nicht unbedingt der Arbeitsbeginn. <https://git-scm.com/docs/git-log>
- Claude Code mit `gh` und in Skripten (`git log | claude -p "..."`): <https://code.claude.com/docs/en/common-workflows>
- Notizdatei gegen doppelte Funktionen: CLAUDE.md (wird in jeder Sitzung geladen) <https://code.claude.com/docs/en/memory> · offenes Format AGENTS.md <https://agents.md>
- Begriffe Pull Request, Diff, Merge: <https://docs.github.com/en/pull-requests/reference/pull-requests>

Zum Weiterlesen

- Baolin, Harvey (DORA, 10.03.2026): Balancing AI tensions. Thematische Analyse von 1.110 offenen Antworten von Google Engineers; gesparte Zeit wird in Prüf- und Prompt-Aufwand umverteilt. Qualitativ, ein Unternehmen. <https://dora.dev/insights/balancing-ai-tensions/>
- DORA / Google Cloud (2024): Announcing the 2024 DORA report. Mehr KI-Nutzung geht mit etwas weniger Liefergeschwindigkeit und Stabilität einher (Befragung, Korrelation). <https://cloud.google.com/blog/products/devops-sre/announcing-the-2024-dora-report>
- Dijkstra, E. W. (1988): EWD 1036, On the cruelty of really teaching computing science ("lines spent" statt "lines produced"). <https://www.cs.utexas.edu/~EWD/transcriptions/EWD10xx/EWD1036.html>
- Grundlage der Serie: Healthy AI First Software Development, Fassung 2 (01.10.2026), Kapitel 3 "Was KI an der Arbeit verändert".

Hinweis

Healthy AI First ist KI-recherchiert und KI-generiert, gesprochen von einer KI-Version der Stimme von Björn und von ihm geprüft.

Jonas, Daniel, Miriam und Hanna sind fiktive Figuren zur Veranschaulichung; ihre Szenen (Dashboard, vierzehn Pull Requests, Statusbericht, Hochzeitsfotos) sind erfunden.