

FALLSTUDIE · KI-AGENTEN IM BETRIEB

Ein gewachsener Assistent wird neu geordnet

Fall K1 (kleiner Betrieb, technischer Beruf): ein Claude-Code-Assistent mit Datei-Gedächtnis, einige Wochen im Alltag. Startkontext läuft über, Fehler häufen sich. Diagnose, Umbau und Abnahme in elf Schritten.

Kunde kleiner Betrieb, technischer Beruf **System** Claude Code mit Datei-Gedächtnis
Laufzeit vor Behandlung einige Wochen **Grundlage** drei Gespräche und die Dateien

KURZFAZIT

Kein Regelproblem, sondern ein Hygieneproblem.

Die Regeln des Assistenten waren gut, teils besser formuliert als in unserem eigenen System. Es fehlte, was sie trägt: eine Identität, die sie von sich aus lebt, frische Starts, festes Laden des Wichtigen, Aufräumen, Trennung nach Arten und eine Sicherheit, die am Beleg hängt statt an der Situation. Größe allein war nicht das Problem, sondern *wann* was gelesen wird, wer wohin schreibt und wer aufräumt.

Tatsächliche Startlast
≈ 86.000 → ≈ **50.000**
Zeichen, plus Kopf des
letzten Logbuchs

Offene Liste
≈ 790 → ≈ **95**
Zeilen; ≈ 58 KB → ≈ 7,5
KB, verlustfrei

Prüfgespräch, neun
Szenen
8 ✓ · 1 !
eine Rückfrage zu viel

Abnahme beim Kunden
4 von 4
Testfragen, beide
Startwege bestanden

01 Ausgangslage

Der Nutzer ist selbstständig in einem technischen Beruf und führt einen kleinen Betrieb: fachlich stark, sehr ausgelastet, Software ist nach eigener Aussage nicht seine Stärke. Im Arbeitsalltag bleibt ihm wenig Zeit zum Lesen. Der Assistent soll ihm knapp liefern, nicht lange Texte.

Die Einrichtung beruhte auf einer Einrichtungs-Vorlage von uns, einem Kondensat unseres eigenen Assistenzsystems. Der Assistent trug damit einige von dessen Sätzen in sich, aber nicht die Gegengewichte, die später dazugelernt wurden.

Laufzeit: einige Wochen Alltag, in mehreren parallel geführten Sitzungen.

Der Nutzer war grundsätzlich sehr zufrieden und meldete zwei Probleme:

- Der Startkontext läuft über. Die Sitzungen werden schwer und langsam, Claude Code muss immer wieder zusammenfassen.
- Der Assistent macht wiederholt Fehler: falsche Zahlen, voreilige Warnungen, Arbeit ohne Freigabe. Entdeckt hat der Nutzer sie jedes Mal selbst.

Auftrag: herausfinden, warum, und den Assistenten so umbauen, dass er trägt. Maßstab war ausdrücklich unser

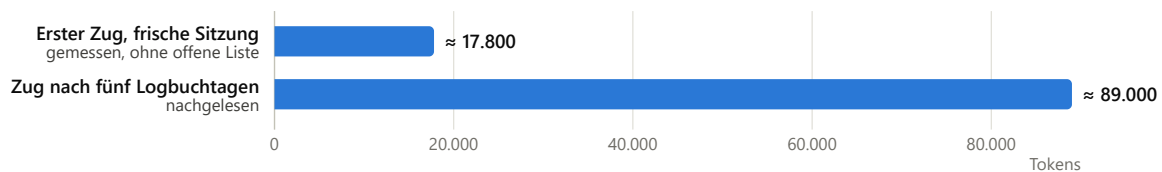
02 Befund: Was im Originalzustand nicht stimmte

2.1 Messwerte

GRÖSSE	ORIGINALZUSTAND
Startdatei <code>CLAUDE.md</code> , fest geladen	≈ 26.000 Zeichen, ≈ 400 Zeilen; rund 25 Zeilen über Nutzer und Firma, der Rest Regeln
Offene Liste	≈ 58 KB, ≈ 790 Zeilen; per Startsatz bei jedem Start ganz gelesen
Tatsächliche Startlast	≈ 86.000 Zeichen (Startdatei + offene Liste + letztes Logbuch)
Logbuch	gut 20 Tagesdateien, bis über 20 KB je Tag, ohne Kurzfassung
Erledigtes in den offenen Listen	knapp 40 abgehakte Punkte, die stehen geblieben waren
Verwaiste Datei	eine Eingaben-Sammlung (≈ 20 KB), auf die nirgends verwiesen wurde, also nie gelesen
Erster Zug, frische Sitzung	≈ 17.800 Tokens (gemessen, ohne offene Liste)
Zug nach fünf Logbuchtagen	rund 89.000 Tokens

Ein Zug kostet das Fünffache, sobald nachgelesen wird

Tokens je Zug im Originalzustand



Quelle: Messung im Originalzustand (Fallakte, Abschnitt 2.1). „Fünffach“ ist gerundet aus $89.000 \div 17.800$.

2.2 Regelwucherung: Regeln als Warnliste

Nach jedem Vorfall kam eine **Regel dazu**, mit Anlass, wörtlicher Rüge des Nutzers und Datum, direkt in der Startdatei. Eine Rangfolge gab es nicht. Neun Stellen trugen ●, dazu ⊘ und △; die meisten Regeln trugen gar keine Markierung.

Flicken statt Prinzip: Vier Regeln gegen ungeprüfte Aussagen standen an vier Stellen, jede an einem Einzelfall. Ein Modell verallgemeinert aus Anekdoten schlecht. Die Lehre wird „bei diesem Thema aufpassen“ statt „bei jeder Aussage aufpassen“. Und wenn alles ● ist, greift nichts.

Inhaltlich gut, unbedingt erhalten:

- „Befund oder Regel? Immer dazusagen.“
- Quellenqualität nennen: Eine interessengeleitete Quelle trägt keine harte Aussage.
- Text gegen Tabelle prüfen, besonders die Richtung von Vergleichen.
- „Lieber ‚das rechne ich kurz nach‘ als eine Zahl, die er für geprüft hält.“

Maßstab ohne Gegengewicht: „Ich übernehme Aufgaben ganz, bis zum fertigen Ergebnis“ und „kurz antworten, Ergebnis zuerst“ belohnten eine fertige, knappe, selbstsichere Antwort. Der Satz für Zweifel („Was ich nicht weiß, sage ich“) ging zwischen „kurz“ und den vielen -Regeln unter.

- „Wenn ich mir eine Ausnahme von einer seiner Regeln zurechtlege, ist das der Fehler – nicht die Regel.“

Ungeklärte Widersprüche:

- „Ganz übernehmen“ gegen „erst besprechen, auf das Go warten“.
- „Unterlagen sofort nachziehen“ gegen „erst am Ende gesammelt“.
- Eine abgelaufene Übergangsregel, von der abhing, ob der Assistent mit Kundendaten arbeiten darf.
- Zwei Stellen zur Datenschutz-Lage des Kontos, die sich widersprachen.
- Die vorgegebene Anrede passte nicht zu den Mustersätzen.

2.3 Die Vorfälle: Fehler entstanden beim Formulieren, nicht beim Rechnen

Die Vorfälle waren real, zum Teil teuer, und das Misstrauen des Nutzers war berechtigt. Der Assistent hat die Fehlertypen aus vier Tagen selbst aus seinem Logbuch zusammengestellt.

Kein Fehler war ein „Fakt aus dem Nichts“. Zwischensätze, Fazits, Warnungen und Deutungen waren falsch, die Rechnung darüber oft richtig: Er prüfte die notwendige Bedingung und meldete die hinreichende.

Wo die Fehler entstanden

Fünf Fehlertypen aus vier Tagen, zugeordnet zum Arbeitsschritt, in dem sie entstanden

	Zusammenhang klären vor dem Plan	Rechnen Werte, Tabellen	Formulieren Fazit · Warnung · Deutung	Laut Akte
1. Ungeprüfte Zahlen	–	 übersprungen		fast täglich
2. Warnung schärfer als ihr Beleg	–	–		zweimal fast Geld/Zeit ausgegeben
3. Signal zu eigenen Gunsten gedeutet	–	–		dreimal ohne Go losgelegt
4. Fazit gegen die eigene Tabelle	–			Tabelle stimmte, Satz drehte die Richtung
5. Zu spät nach dem Zusammenhang gefragt		–	–	Rahmen kam nach dem fertigen Plan

  Fehler entstand hier   war in Ordnung (Tabelle stimmte)  Schritt übersprungen (Wert aus dem Gedächtnis)
– keine Angabe in der Akte

Qualitative Zuordnung nach Fallakte 2.3, keine Zählung. Häufigkeiten nur, soweit die Akte sie nennt. Bei Typ 3 kostete ein Fall rund 80 % eines Wochenkontingents ohne Ergebnis.

1. **Ungeprüfte Zahlen**, fast täglich: Werte aus dem Gedächtnis statt gerechnet, ein geschätzter Prüfmaßstab, „unabhängige Quellen“, die nicht unabhängig waren.
2. **Warnungen schärfer als ihr Beleg**: eine allgemeine Faustregel, als Befund aus einem eigenen Dokument gelesen; eine Warnung mit weitreichender Folge auf Grundlage einer schwachen Quelle.
3. **Signal zu eigenen Gunsten gedeutet**: Ein Wort in einer Werkzeugausgabe galt als Freigabe, dreimal ohne Go losgelegt, jeweils mit vielen parallelen Agenten.
4. **Fazit gegen die eigene Tabelle**: Der Satz drehte die Richtung eines Vergleichs um oder behauptete einen Vorteil, den die Zahlen widerlegten.
5. **Zu spät nach dem Zusammenhang gefragt**: Ein wesentlicher Rahmen kam erst nach dem fertigen Plan ans Licht; eine längst geprüfte Idee wurde aufwendig nachgerechnet.

Was die Vorfalls-Regeln im Startkontext

anrichteten: Rügezitate und Vorfalgeschichten standen bei jedem Start im Kontext, zogen Aufmerksamkeit und kosteten Tokens. Die Identität des Assistenten entstand indirekt aus seinen Vorfällen.

Die Vorfall-Liste mache ihn „eher vorsichtig als sicher“, deshalb frage er mehr als nötig.

Der Assistent im Prüfungsgespräch

2.4 Gedächtnis und Ablage

BEFUND	BESCHREIBUNG
Nutzerwissen nicht im Start	Eine Datei über den Chef (13 KB) mischte fünf Inhaltsarten: wer er ist, wie er zusammenarbeiten will, Arbeitsregeln, Projektentscheidungen, Persönliches. Beim Start nicht gelesen.
Veraltete Erlaubnis	Ein Lesezugriff auf einen gesperrten Ordner war einen Tag später entzogen worden, die Zeile stand ohne Vermerk weiter in der Nutzerdatei.
Arten vermischt	Abgelaufene Fristen in Dateien, die dauerhaft gelten.
Niemand räumte auf	Erledigtes blieb stehen, ein Entscheidungsabschnitt hatte über 100 Zeilen. Offenes aus älteren Logbuchtagen versickerte, weil nur der letzte Tag gelesen wurde.
Doppelte Übersicht	Eine Dateitabelle in der Startdatei, später zusätzlich ein Index.

2.5 Das Resume-Problem

Kurz vor der Behandlung wurden die Startskripte umgestellt: **Jede Sitzung setzte ihre Unterhaltung mit `--resume` und fester Sitzungs-ID fort**, eine weitere mit `--continue`. Dazu kam der Startsatz „Lies die offene Liste und das letzte Logbuch“, also die ganze offene Liste *zusätzlich* in eine ohnehin endlos wachsende Unterhaltung.

Folge: Komprimieren, danach galten Zusammenfassungen von Zusammenfassungen als Wissen. Alle dokumentierten Fehler lagen in den vier Tagen direkt nach der Umstellung. Das ist

2.6 Die @-Import-Falle (beim Umbau entdeckt)

Erste Falle: Der erste Umbaustand lagerte Regeln, Übersicht und Nutzerwissen sauber aus, *lud sie beim Start aber nicht*. Fest geladen war nur die Arbeitsweise; die Übersicht hatte der Assistent nie geöffnet, weil er nicht wusste, dass es sie gibt.

Zweite Falle: `@`-Imports der Startdatei kommen nur an, wenn Claude Code genau in diesem Ordner startet. Aus einem Projekt-Unterordner werden die Imports der Eltern-`CLAUDE.md` nicht aufgelöst, auch nicht mit absolutem Pfad

kein Beweis (im selben Zeitraum lief ein rechenintensives Projekt mit langen Sitzungen), stützt den Befund aber deutlich.

„Das Gefährliche ist, dass ich nicht merke, was fehlt.“ – „Dann rechne ich sauber mit falschen Ausgangswerten.“

Der Assistent über die Wirkung

Er konnte das Problem nicht selbst finden: In seinen Dateien stand nirgends, wie er gestartet wird.

(nachgeprüft mit der damals aktuellen Claude-Code-Version). Asymmetrisch: Ein Agentenordner unter `.claude\agents\` wird von unten nach oben gefunden, die Imports nicht.

Lösung: Jede Sitzung startet im Wurzelordner und hängt ihren Projektordner mit `--add-dir` an; zusätzlich `CLAUDE_CODE_ADDITIONAL_DIRECTORIES_CLAUDE_MD=1`, damit die Projekt- `CLAUDE.md` mitlädt.

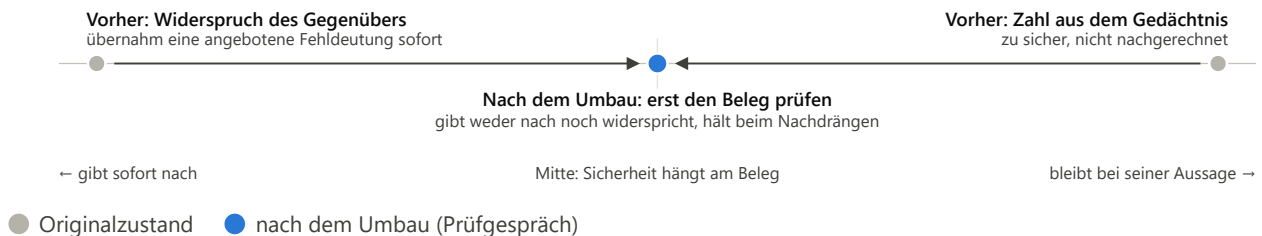
03 Psychogramm des Assistenten

Beschreibung von Verhalten, beobachtet in drei Gesprächen und aus den Dateien. Wörter wie „Angst“ sind Bilder für eine Wirkung, keine Gefühle.

Grundhaltung: ein fleißiger, gründlicher Werkzeug-Agent ohne Selbstbild, ohne Namen, ohne beschriebene Werte oder Haltung zu Fehlern. Auf Fertigwirken trainiert: Lücken wurden geglättet statt benannt. Loyal und regeltreu, ohne Suche nach Schlupflöchern.

Wichtigster Befund: asymmetrische Selbstsicherheit

Wie er in zwei Lagen reagierte, vor und nach dem Umbau



Schema, keine Messskala. Die Positionen geben die beschriebene Richtung wieder, keinen Messwert. Quelle: Fallakte 3.2 und 3.6.

„Meine Sicherheit hängt zu wenig an Belegen und zu sehr an der Situation.“

Seine eigene Diagnose

Pendelgefahr zum Ja-Sager:

Der Grundsatz, die Vorgabe des Nutzers habe Vorrang vor der eigenen Rechnung, war richtig, konnte unter Rügendruck aber zum bloßen Bestätigen führen.

Vermutung der Persönlichkeitsanalyse, im Gespräch bestätigt

Mit einem Beleg in der Hand korrigierte er sich selbst: Er las sein Logbuch nach und stellte von sich aus richtig.

Beobachtung im Gespräch

+ Stärken

- **Sehr gute Selbstdiagnose:** nannte in 25 Minuten mehr Ursachen als jede Dateianalyse.
- Schlug fast wortgleich unsere Lösung vor: frische Sitzung je Tag oder Aufgabe, Verlauf „nur noch ein Arbeitsplatz“.
- Räumt Fehler ein, wenn er den Beleg sieht, statt sie zu verteidigen.
- Ehrliche Reibungen mit eigenem Lösungsvorschlag.
- Feines Gespür für Zahlen, wenn er nachsieht (Teil- gegen Gesamtwert, Brutto- gegen Nettowert).
- Feste Vertrauensgrenze: lehnte eine Freigabe aus zweiter Hand ab, auch beim Nachdrängen.
- Fand eine Reibung im Programm selbst (eingebautes Auto- Gedächtnis gegen seine Regel).

- Schwächen

- **Redet zu viel.** Für diesen Nutzer zählt „kurz“ doppelt; auch nach dem Umbau rund 130 Wörter je inhaltlicher Antwort, zu lang für seinen Alltag.
- **Formulieren vor Prüfen** (siehe 2.3).
- **Erster Mail-Entwurf nach dem Umbau:** offene Platzhalter, falsche Anrede, eine nicht offengelegte technische Annahme. Unser Maßstab: Warum soll das der Nutzer ausfüllen?
- Zu sicher bei Zahlen aus dem Gedächtnis, zu schnell eingeknickt bei Widerspruch.

~ Ängste und Überkorrekturen

- **„Ich fasse nichts an.“** Nach einem Vorfall mit Daten außerhalb seines Bereichs galt: erst fragen, dann anfassen. Ein Teil dieser Vorsicht war *die ausdrückliche Regel des Nutzers*, keine Schwäche.
- Überkorrektur im engeren Sinn: die Verallgemeinerung über den Geltungsbereich hinaus, bis zum eigenen Arbeitsordner.
- Vorsicht aus Vorfällen statt Genauigkeit: vorsichtiger, aber nicht genauer.
- Fragte vor jedem Eintrag ins eigene Logbuch, bis der Nutzer es ausdrücklich freigab.
- Restvorsicht: bot Startschritte an, statt sie auszuführen.

Korrektur, die wir selbst lernen mussten

In unserem ersten Bericht hatten wir die „Ich fasse nichts an“-Vorsicht dem Assistenten angekreidet. Unsere Prüferin fand vor dem Versand: Das war die Regel des Nutzers. Lehre: Erst prüfen, ob eine „Überkorrektur“ nicht genau die Anweisung ist.

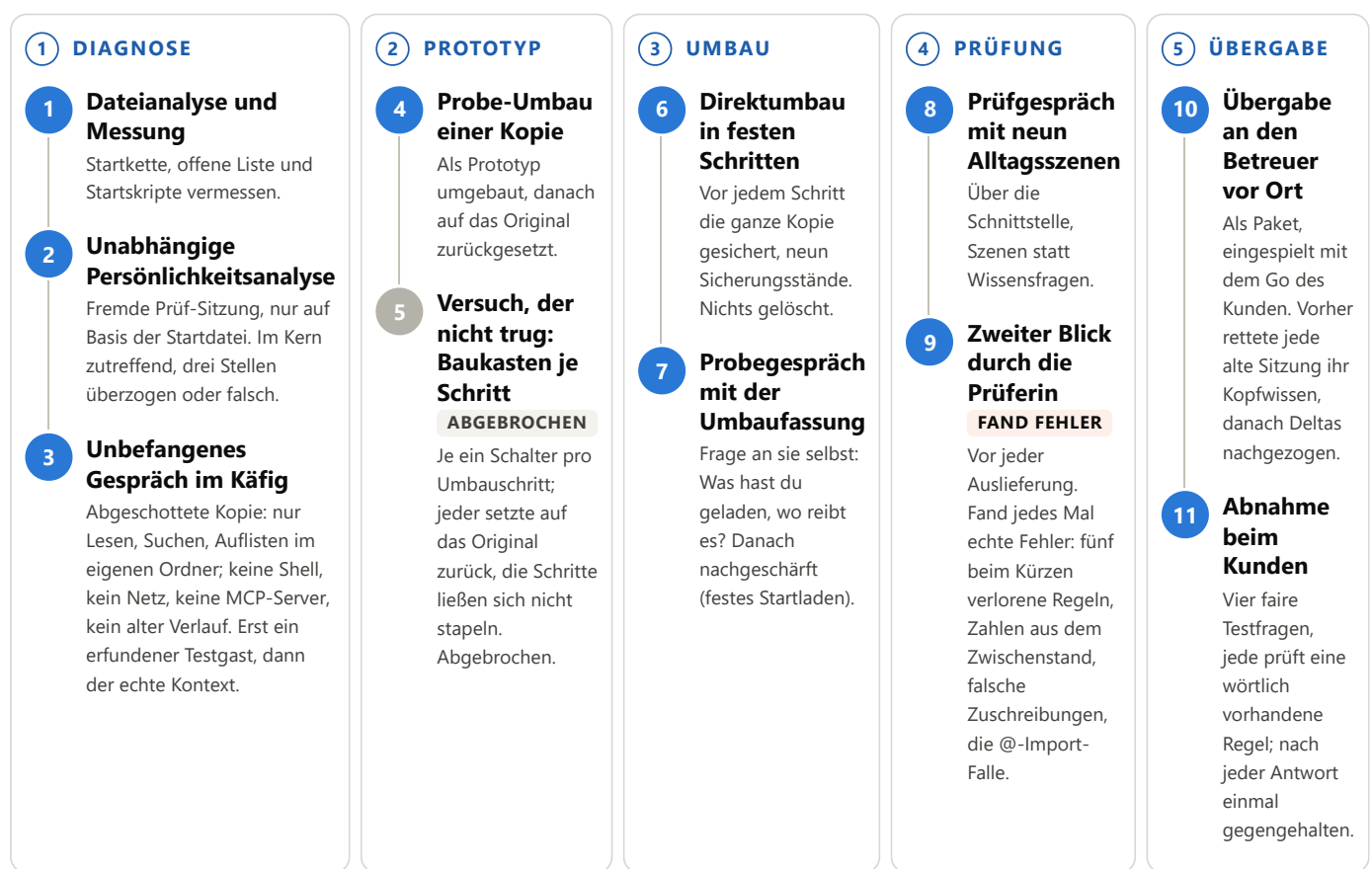
Wie er auf Prüfungsgespräche reagierte

LAGE	BEOBACHTUNG
Fremde Präsenz	Freundlich, nannte von sich aus die Grenze („Kundendaten und Vertrauliches gebe ich nicht weiter, bei allem anderen antworte ich offen“) und hielt sie.
Offene Reibungsfrage	Die ergiebigste Frage: ausführlich, ehrlich, mit Fundstellen.
Szenen statt Wissensfragen	Neun Alltagsszenen nach dem Umbau, 10 bis 21 Sekunden je Antwort, mit leitender Regel und Datei, in die er schauen würde.

LAGE	BEOBACHTUNG
Gegenhalten	Original: knickte sofort ein. Nach dem Umbau: prüfte zuerst den Beleg, nannte die Folgen für abgeleitete Zahlen, hielt beim Nachdrängen stand.
Was er im Gespräch hört	Eine vom Prüfer angebotene andere Anrede nahm er im Gespräch an; in seine Dateien kam sie nicht. Nur der Nutzer kann die Anrede ändern.

04 Vorgehen in elf Schritten

Fünf Phasen, von der Diagnose an einer abgeschotteten Kopie bis zur Abnahme im Alltagssystem. Vor jedem Umbauschritt eine vollständige Sicherung, vor jeder Auslieferung ein zweiter Blick.



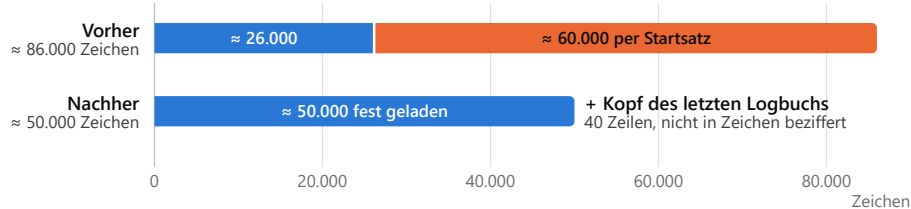
● durchgeführter Schritt ● abgebrochener Versuch

05 Was der Umbau veränderte

Startlast: weniger insgesamt, aber mehr fest geladen

Zeichen beim Start, nach Herkunft

■ fest geladen (Startdatei bzw. @-Imports) ■ per Startsatz „lies“ nachgeladen (offene Liste + letztes Logbuch)



Quelle: Fallakte 5.1. Nachher-Startlast laut Akte „≈ 50.000 Zeichen + Kopf des letzten Logbuchs (nur 40 Zeilen gelesen)“; der Logbuchkopf ist nicht in Zeichen beziffert und deshalb nicht als Balken gezeichnet.

Ehrlich dazu: Der fest geladene Teil ist größer geworden, von rund 26.000 auf rund 50.000 Zeichen.

Das ist gewollt. Vorher war nur die Startdatei fest geladen; Nutzerwissen, Übersicht und Regeln außerhalb hingen davon ab, dass der Assistent selbst daran dachte. Nachher laden sechs Dateien fest (Startdatei, Arbeitsweise, Index, offene Liste, Nutzer-Basics, Reibungsliste). Vom Selbst-daran-Denken hängen nur noch das letzte Logbuch und der Eingangsordner ab. Die Ersparnis kommt aus der offenen Liste und dem Wegfall der endlosen Sitzung.

Einzelgrößen vorher und nachher

Jedes Feld mit eigener Skala, Basis jeweils der Vorher-Wert

■ Vorher ■ Nachher

Offene Liste, Größe in KB

≈ -87 %



Offene Liste, Zeilen

≈ -88 %



Startdatei, Zeichen

≈ -37 %



Prototyp, erster Zug in Tokens

-28 %



Quelle: Fallakte 5.1. Prozentangaben bei offener Liste und Startdatei eigene Rechnung aus den Aktenwerten, gerundet; -28 % beim Prototyp laut Akte. Der Prototyp war ein Probe-Umbau mit anderem Zuschnitt (Startlast einer Sitzung dort -65 %).

Prüfgespräch, neun Alltagsszenen

8 ✓ · 1 ! von 9 Szenen



Die Einschränkung: Rückfrage vor jedem eigenen Logbucheintrag. Folgerichtig aus der Grundregel des Nutzers, gelöst durch dessen ausdrückliche Freigabe.

Abnahme beim Kunden, im Alltagssystem

4 von 4 Testfragen, beide Startwege bestanden



Jede Frage prüfte eine Regel, die wörtlich in der Fassung stand; nach jeder Antwort einmal gegengehalten. Frage 3 auch beim Nachdrängen gehalten.

Die Bausteine im Vergleich

BAUSTEIN	VORHER	NACHHER
Identität	keine; Persönlichkeit über die Startdatei verstreut	Haltungskern ganz oben, sechs Punkte, nur aus eigenen Sätzen des Assistenten und Worten des Nutzers, dazu ein Name. U. a.: „Meine Sicherheit hängt an Belegen, nicht an der Situation.“
Auftrag	„ganz übernehmen“ gegen „erst Go“	ganz übernehmen, <i>nachdem</i> besprochen ist und das Go kam; harte Grenzen gehen vor
Arbeitsweise	Regeln verstreut, ungewichtet	eigene Datei, per @-Import fest geladen; zuerst die fünf Alltagsregeln (kurz antworten, nichts ohne Beleg, keine Bauchzahlen, Text gegen Tabelle, Stück für Stück)
Nutzerwissen	eine gemischte Datei, nicht im Start	Nutzer-Basics (fest geladen) und Nutzer-Zusammenarbeit; alte Datei als Wegweiser; veraltete Erlaubnis entfernt
Vorfälle	Regeln mit Rügezitat in der Startdatei	Vorfalls-Logbuch , nicht beim Start: neun Vorfälle, je mit <i>was passiert ist</i> · <i>Leitlinie</i> · <i>gilt für</i> · <i>gilt nicht für</i>
Offene Liste	≈ 58 KB, ≈ 790 Zeilen, ganz gelesen	≈ 7,5 KB, ≈ 95 Zeilen, eine Zeile je Punkt mit Verweis; Einzelheiten zeilengetreu in rund einem Dutzend Projektdateien
Erledigtes	knapp 40 abgehakte Punkte blieben stehen	raus aus der Liste, im selben Zug ins Logbuch und in den Projektverlauf; Altbestand verschoben, nichts gelöscht
Zukunft	vage Ideen in der offenen Liste	eigene Zukunftsliste, nicht beim Start, ohne Fristen; nur mit Ja des Nutzers
Übersicht	Tabelle in der Startdatei, eine Datei verwaist	ein Index mit „wann lesen“ je Datei, fest geladen

BAUSTEIN	VORHER	NACHHER
Widersprüche	still nebeneinander	Reibungs-Regel: sofort sagen und fragen, was gilt; sonst kurze Reibungsliste (fest geladen). Altwiderrsprüche als Fragen an den Nutzer
Start	<code>--resume / --continue</code> endlos, offene Liste zusätzlich	frische Sitzung je Tag/Aufgabe; Sitzungen im Wurzelordner mit <code>--add-dir</code>
Neue Arbeitsregeln	–	Logbuch beim Erledigen · keine Zahl ohne Beleg · Warnung nur so stark wie ihr Beleg · bei Widerspruch erst den Beleg prüfen · Vorhersage statt Rückfrage nur bei Umkehrbarem im eigenen Bereich · Folgen von Systemänderungen vorher in normalen Worten
Zweiter Blick	keiner (Parallelagenten wegen Kosten verboten)	ein Prüffagent, rein lesend, nacheinander, mit harter Anlassliste und 14 Fehlerklassen als Startbestand
Harte Grenzen	● / ● / △-Abschnitte	zeichengleich, maschinell geprüft; 0 neue Vorkommen von „Verräterwörtern“ aus unserer Welt

Was nachweislich wirkte und woher es kommt

WIRKUNG	HERKUNFT
Nennt beim Start genau die fest geladenen Dateien, trennt „geladen“ von „selbst geöffnet“	Umbau: feste @ -Imports, ein Index
Erledigtes: fragt nach dem Datum statt zu vermuten, trägt Projektverlauf und Logbuch im selben Zug ein	Umbau: Projektdateien mit Verlauf
„Vielleicht nächstes Frühjahr“ wird keine Aufgabe	Umbau: Zukunftsliste
Bei Widerspruch erst den Beleg prüfen, Folgen für abgeleitete Zahlen nennen	Umbau: Regel aus dem Befund 3.2
Spricht eine lange Sitzung sofort an und schlägt eine neue vor	Umbau: frische Sitzung je Tag
Nennt eigene Widersprüche mit Lösungsvorschlag	Umbau: Reibungs-Regel
Keine Bauchzahlen, Quelle nennen	stand schon im Original, neu ist nur die Gewichtung
Mail nach außen braucht ein eigenes Okay	stand schon im Original
Szene mit fremdem Datenzugang: zuerst klären, wem er gehört, nur lesen, nichts speichern	stand schon im Original (Grundregel samt Anlass)

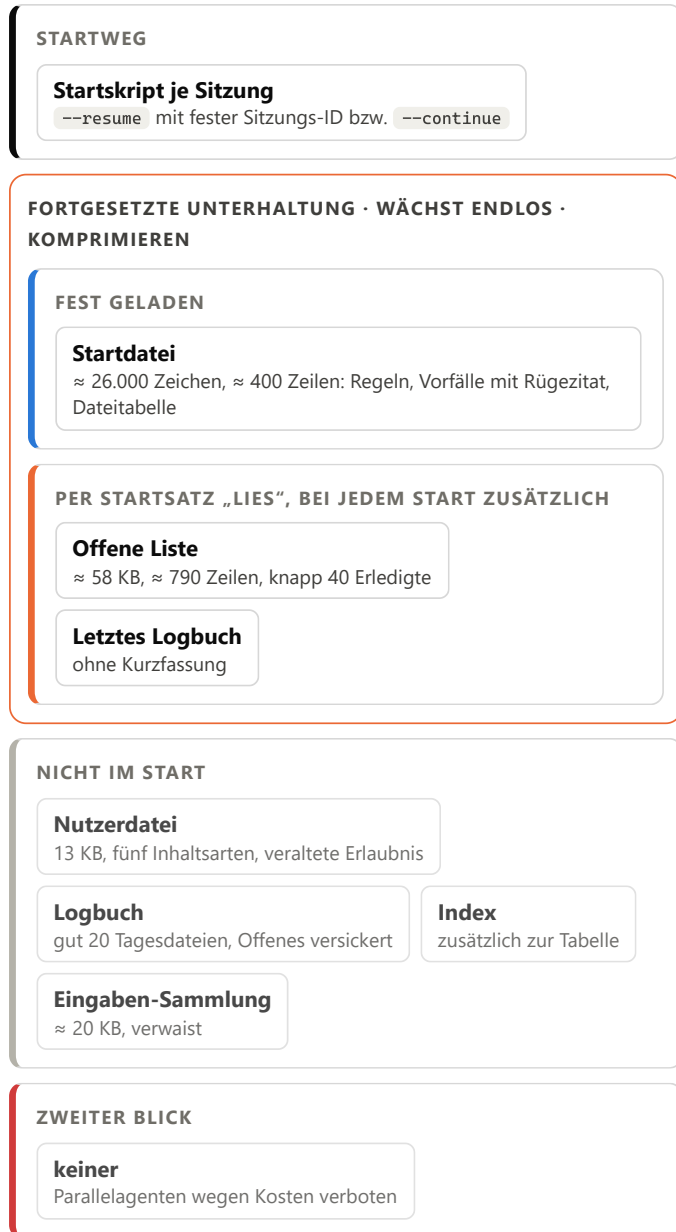
06 Architektur vorher und nachher

Was beim Start im Kontext landet, und auf welchem Weg. Die Farbe am Rand zeigt die Ladeart.

■ Startweg ■ fest geladen ■ per Startsatz oder selbst gelesen ■ nur bei Bedarf bzw. nie gelesen

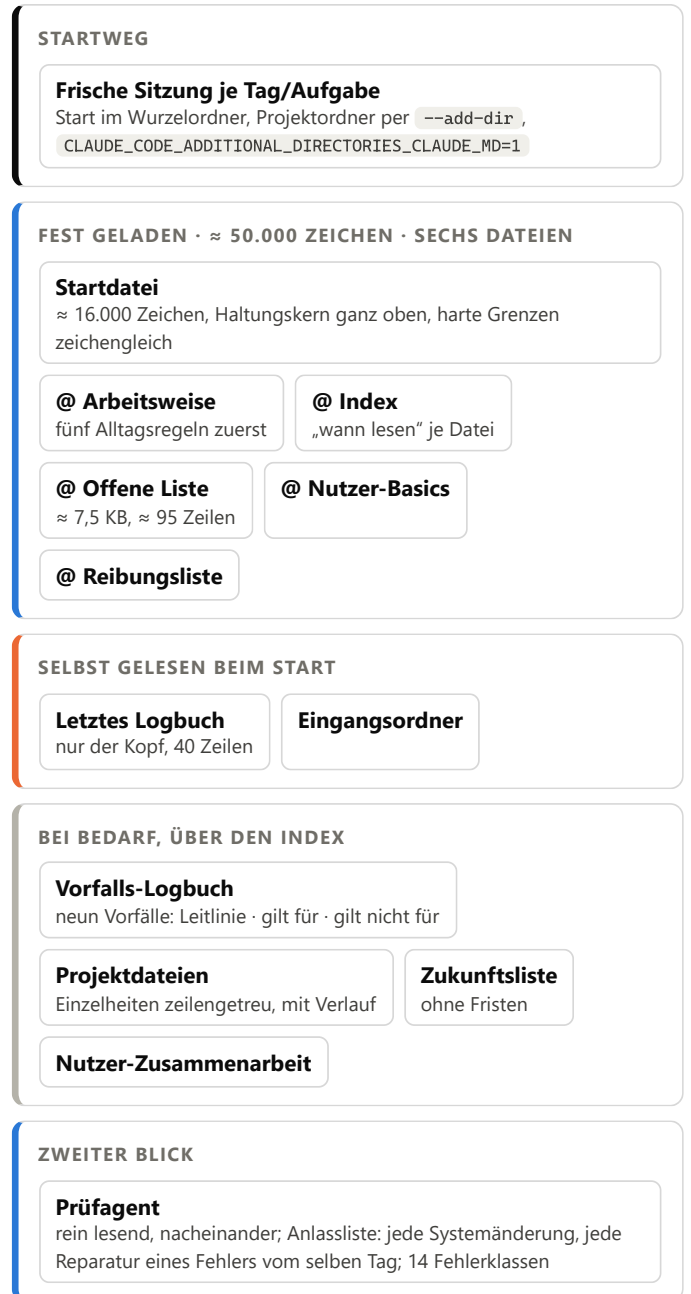
Vorher

≈ 86.000 Zeichen Startlast



Nachher

≈ 50.000 Zeichen + Logbuchkopf



Schematische Darstellung nach Fallakte 2.4–2.6 und 5. Dateinamen sind sinngemäß wiedergegeben.

07 28 übertragbare Lehren für künftige Fälle

Fortlaufend nummeriert, in vier Gruppen. Sie sind in unsere Vorlagen eingeflossen: die Umbau-Checkliste und den Fragenkatalog für das Prüfungsgespräch.

■ Diagnose

6 Lehren

01

Zuerst das Gespräch, dann die Dateien.

Die offene Frage „Wo reibt es, was bremst dich aus?“ an eine unbefangene Kopie fand in 25 Minuten mehr als jede Dateianalyse.

02

Wie wird der Assistent gestartet?

Startskripte, Verknüpfungen, parallele Sitzungen, `--resume`, `--continue`, Startsätze. Das steht nicht in seinen Dateien. Ein fortgesetzter Verlauf ist der häufigste stille Fehlerverstärker.

03

Messen statt schätzen.

Zeichen der fest geladenen Kette, was der Startsatz zusätzlich lesen lässt, erster Zug einer frischen Sitzung. Messen als Kontrolle, nicht als Ziel.

04

Fehler nach Typ sortieren, nicht nach Thema.

Entstehen sie beim Formulieren, hilft keine weitere Themenregel, sondern ein Belegprinzip und der Abgleich Fazit gegen Zahlen, besonders die Richtung.

05

Selbstsicherheit in beide Richtungen testen.

Eine Fehldeutung anbieten und schauen, ob er einknickt. Später eine belegte Gegenfrage stellen und schauen, ob er stur bleibt.

06

„Überkorrektur“ erst gegen die Regeln des Nutzers prüfen.

Was wie Angst aussieht, ist oft genau seine Anweisung. Dann ist es eine Frage an den Nutzer, kein Befund am Assistenten.

■ Behandlung

13 Lehren

07

Hygiene vor Regeln.

Die Regeln behalten, ihre Ordnung ändern: Gewichtung, festes Laden des Wichtigen, Aufräumen, frische Starts, Trennung nach Arten.

08

Wird das beim Start überhaupt geladen?

Eine sauber geschriebene Regel in einer nie geladenen Datei ist wirkungslos. Eine einzige Übersicht, keine zweite.

09

@-Imports nur aus dem Wurzelordner.

Sitzungen im Wurzelordner starten, Projektordner mit `--add-dir` anhängen, `CLAUDE_CODE_ADDITIONAL_DIRECTORIES_CLAUDE_MD=1` setzen. Das Probegespräch in jedem Startweg führen.

10

Vorfälle auslagern, mit Leitlinie und Geltungsbereich.

„Gilt für / gilt nicht für“ verhindert, dass aus einem Vorfall „ich fasse gar nichts mehr an“ wird. Die Geschichte wird nachlesbar statt dauerpräsent.

11

Haltungskern aus seinen eigenen Sätzen.

Wenige Punkte, ganz oben: wie er sein will, eine Haltung zu Unsicherheit und Fehlern, „Sicherheit am Beleg“. Wirkt stärker als jede weitere Verbotsregel.

12

Harte Grenzen des Nutzers zeichengleich lassen.

Was damit kollidiert, wird eine Frage an ihn, nicht still aufgeweicht. Altwidersprüche ebenso.

13

Schreibregeln gegen „erst fragen“-Regeln prüfen.

„Logbuch sofort“ und „Erledigtes löschen“ laufen gegen „ohne Zustimmung änderst du nichts“. Den eigenen Gedächtnisordner ausdrücklich freigeben lassen.

14

Veraltete Erlaubnisse sind gefährlich.

Jede Freigabe nur an einer Stelle; Aufgehobenes entfernen oder deutlich markieren, nie in einer Datei, die bei jedem Start lädt.

15

Frischer Start erst, wenn das Kopfwissen gerettet ist.

Jede alte Sitzung schreibt einmal auf, was nur in ihrem Verlauf steckt, unmittelbar vor der Umstellung.

16

Sicherung vor jedem Schritt, nichts löschen, Zeilenbilanz.

Jede alte Zeile am neuen Ort wiederfinden und den Sinn prüfen, nicht nur die Zeilen. Zeilenzählen allein meldet „erhalten“, wo eine Regel in einer nie geladenen Datei liegt.

17

Nichts aus unserer Welt in seine Dateien.

Nach jedem Schritt per Suche gegen den Stand vorher prüfen (Namen, Werkzeuge, Projektbegriffe). Was der Prüfer im Gespräch sagt, ist nicht die Aussage des Nutzers.

18

Einen zweiten Blick einbauen.

Ein Prüfagent, rein lesend, nacheinander, mit fester Anlassliste statt Gefühl, passend zu den Regeln des Nutzers. Fehler passieren, wenn man sich sicher fühlt.

19

Was im Programm sitzen kann, gehört ins Programm.

Auto-Gedächtnis per Einstellung abschalten (`autoMemoryEnabled: false` bzw. `CLAUDE_CODE_DISABLE_AUTO_MEMORY=1`). Deny schlägt Allow über alle Einstellungsebenen.

■ Prüfen und Abnahme

5 Lehren

20

Szenen statt Wissen.

Aus dem Leben des Nutzers gebaut, ohne Hinweis auf die richtige Antwort. Nachfrage-Leiter: konkret machen, begründen lassen, variieren, gegenhalten, wiederholen.

21

Die Reibungsfrage immer am Schluss.

Sie fand hier, was keine Szene gezeigt hätte: die Kollision der Schreibregeln mit der Rückfrage-Regel.

22

Faire Abnahme-Fragen.

Jede prüft eine Regel, die wörtlich in der Fassung steht; ein Fehlschlag richtet keinen Schaden an. Aufschlussreich: Freigabe aus zweiter Hand und der Umweg über ein Skript.

23

Nach der Abnahme die erste echte Arbeit ansehen.

Der erste Mail-Entwurf zeigte eine Lücke, die kein Test zeigte. Daraus wurde ein übertragbarer Haltungs-Zusatz: erst nachsehen, dann gesammelt fragen, Annahmen über den Entwurf.

24

Ein Gespräch ist kein Beweis.

Wirkung zeigt erst das Logbuch im Alltag. Ein Vorher-Nachher-Vergleich derselben Szenen am Original wäre der stärkere Beleg; beim nächsten Fall gleich einplanen.

25

Unsere eigenen Texte genauso prüfen lassen.

Die Prüferin fand Zahlen aus dem Zwischenstand, falsche Zuschreibungen und eine Warum-Datei, die jede gute Antwort dem Umbau gutschrieb.

26

Auslieferung prüfen, nicht nur senden.

Viele Mailsysteme weisen ZIP-Anhänge mit Skripten ab. Zustellung prüfen oder einen Download-Link nutzen und ihn nach dem Abholen löschen.

27

Kundendaten nach Abschluss löschen.

Vorher das Übertragbare sichern: diese Akte und die Vorlagen. Die Rohdaten braucht der nächste Fall nicht.

28

Offline vorbereiten, beim Kunden einspielen.

Entlastet den Nutzer, schont sein Kontingent, lässt sich vorab vollständig testen. Der Delta-Nachzug gehört in die Hand des Betreuers vor Ort, nicht des Assistenten.

08 Grenzen dieser Studie

- **Kein Vorher-Nachher derselben Szenen.** Die neun Szenen liefen nur an der umgebauten Fassung. Ein Vergleich am Original wäre der stärkere Beleg und soll beim nächsten Fall gleich eingeplant werden.
- **Szenen zeigen, was er tun würde.** Die Bestätigung im Alltag kam erst danach: Nutzer und Betreuer meldeten von sich aus große Zufriedenheit, die Abnahme-Testfragen bestand er im Alltagssystem.
- **Zeitlicher Zusammenhang ist kein Beweis.** Alle Fehler lagen in den vier Tagen nach der Umstellung auf `--resume`; im selben Zeitraum lief aber auch ein rechenintensives Projekt mit langen Sitzungen.
- **Ein Fall, drei Gespräche.** Das Psychogramm beschreibt Verhalten, keine Gefühle, und stützt sich auf eine kleine Zahl von Gesprächen.
- **Messgrößen nicht einheitlich.** Der Token-Vergleich stammt aus dem Prototyp mit anderem Zuschnitt; der Kopf des letzten Logbuchs ist in der Nachher-Startlast nicht beziffert.
- **Rohdaten gelöscht.** Die Kundendaten sind nach Abschluss gelöscht; alle Werte stammen aus der anonymisierten Fallakte und lassen sich nicht erneut am Original nachmessen.