AGENTEN-ÖKONOMIE · ÜBERBLICK

Warum KI-Agenten teuer werden — die sechs Kostentreiber

Token-Preise fallen, KI-Rechnungen steigen trotzdem. Sechs Muster erklären, warum — jedes hat Gegenmittel.

1 Langlebiger Kontext

KI-Modelle sind zustandslos — Agenten senden den wachsenden Verlauf bei jedem Schritt komplett neu mit.

bis ~1.000×

2 Verfeinerung ist das Kostenloch

Nicht die erste Antwort ist teuer — das Prüfen, Reparieren und Nachverifizieren danach.

~60 %

3 Autonomie erzeugt Varianz

Derselbe Auftrag kann je nach gewähltem Lösungsweg des Agenten sehr unterschiedlich viel kosten.

bis 30×

4 Teures Denken für simple Aufgaben

Der Reflex, für alles das stärkste Modell zu nutzen — auf einfachen Aufgaben reiner Overhead.

Overhead

5 Orchestrierung vervielfacht Kosten

Wie Arbeit auf Agenten, Werkzeuge und Zwischenschichten verteilt wird, verändert die Rechnung dramatisch.

mehrfach bezahlt

6 Informationsstruktur treibt Verbrauch

Wie Information formuliert und formatiert ist, bestimmt die Token-Zahl — bei gleichem Inhalt.

–30–40 %

→ Zu jedem Treiber folgt eine Detailseite: Problem als Diagramm links, Gegenmittel rechts.

Kostentreiber nach: McKinsey & Company, „Is that AI agent worth it? Agentic economics and the modern operating model“, Juli 2026. Gegenmittel: NoviCogi.

Billigere Modell-Aufrufe allein senken die Rechnung nicht.

Vier der sechs Treiber sind Architektur- und Prozessfragen — deshalb ist KI-Kostensteuerung Managementarbeit.

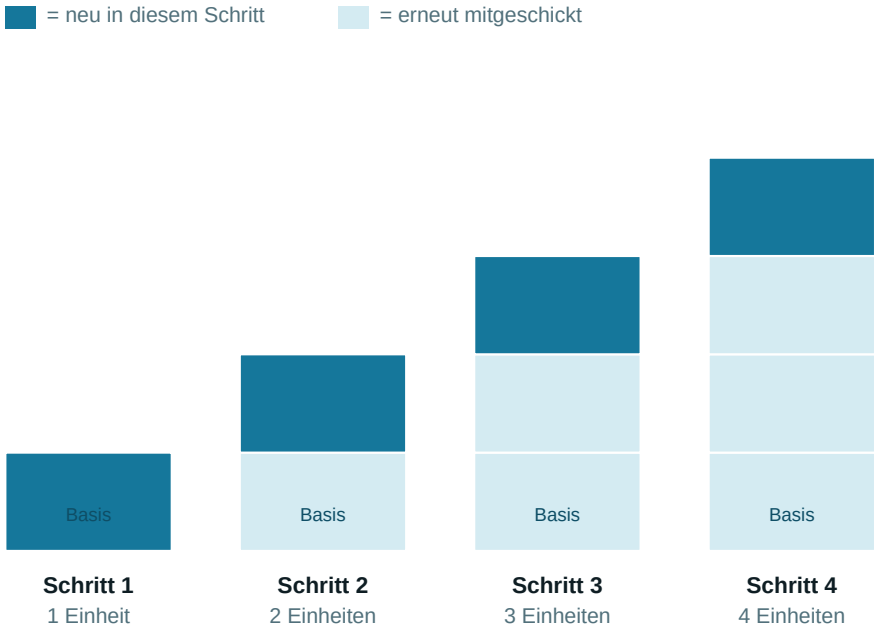
KOSTENTREIBER 1 VON 6

Langlebiger Kontext

bis ~1.000×

KI-Modelle sind zustandslos — Agenten senden den wachsenden Verlauf bei jedem Schritt komplett neu mit.

Jeder Schritt schickt alles Vorherige erneut



Vier Schritte kosten nicht 4, sondern 1+2+3+4 = 10 Einheiten — die Kosten wachsen annähernd quadratisch mit den Schritten.

Gegenmittel

✓ Prompt-Caching
Der statische Teil (Anweisungen, Faktenblatt) wird zwischengespeichert und kostet bei Wiederholung nur noch rund ein Zehntel.

✓ Kontext-Hygiene
Zwischenergebnisse zusammenfassen oder verwerfen; nur die relevante Scheibe laden statt der ganzen Datenbank.

✓ Single-Turn bevorzugen
Wenn die Aufgabe keinen Mehrschritt-Agenten braucht: ein Aufruf mit kuratiertem Kontext schlägt zehn Schritte mit Suche.

Kontext ist ein Betriebsgut — gemanagt, nicht mitgeschleppt.

Agentische Aufgaben kosten bis zu ~1.000× mehr Tokens als ein einfacher Chat.

KOSTENTREIBER 2 VON 6

Verfeinerung ist das Kostenloch

~60 %

Nicht die erste Antwort ist teuer — das Prüfen, Reparieren und Nachverifizieren danach.

Wo die Kosten einer Agenten-Aufgabe wirklich sitzen

Der Erstentwurf ist der billige Teil. Teuer wird es danach: Qualitätskontrolle, Ausnahmebehandlung und Nacharbeit — sie gehören von Anfang an in die Prozesskosten.

Gegenmittel

✓ Vorne investieren

Strikte Ausgabeformate und kuratierter Kontext senken die Nacharbeit drastisch — Qualität entsteht vor dem ersten Aufruf.

✓ Billige Prüfung

Verifikation kleinen Modellen oder festen Regeln (Schema-Checks) überlassen — nicht dem teuersten Modell.

✓ Rework-Quote messen

Wie oft wird ein Ergebnis verworfen oder wiederholt? Diese Quote ist der echte Kosten-KPI, nicht der Einzellauf.

Nicht die erste Antwort ist teuer — die Nacharbeit.

Rund 60 % der Kosten einer Agenten-Aufgabe stecken in der Verfeinerung.

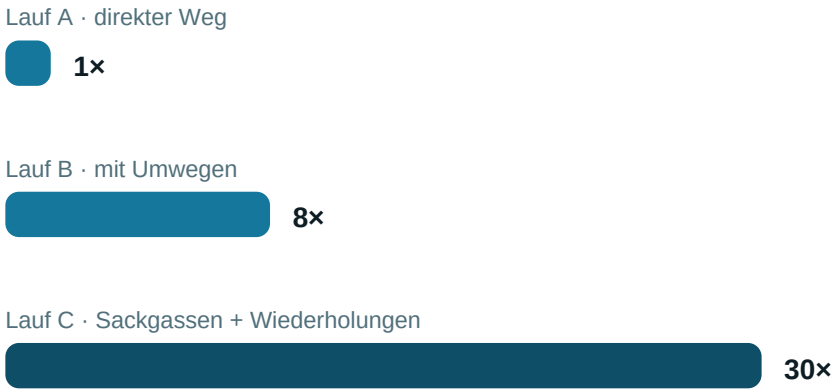
KOSTENTREIBER 3 VON 6

Autonomie erzeugt Varianz

bis 30×

Derselbe Auftrag kann je nach gewähltem Lösungsweg des Agenten sehr unterschiedlich viel kosten.

Dieselbe Aufgabe, drei Läufe — drei Rechnungen



Der Agent wählt seinen Weg selbst: andere Suchpfade, andere Tool-Aufrufe, Wiederholungen. Kosten verhalten sich als Verteilung — ein Durchschnitts-Budget übersieht die Ausreißer.

Gegenmittel

✓ Stoppregeln setzen

Kein autonomer Lauf ohne Limit: maximale Schritte, maximale Kosten, maximale Zeit — als feste Regel im System.

✓ Autonomie dosieren

Feste Abläufe, wo der Weg bekannt ist; freie Agenten nur dort, wo wirklich exploriert werden muss.

✓ Ausreißer überwachen

Nicht den Durchschnitt budgetieren — die teuersten Läufe ansehen, erklären und deckeln.

Kosten sind eine Verteilung — budgetiere die Ausreißer, nicht den Durchschnitt.

Gleiche Aufgabe, bis zu 30-fach unterschiedliche Kosten je Lauf.

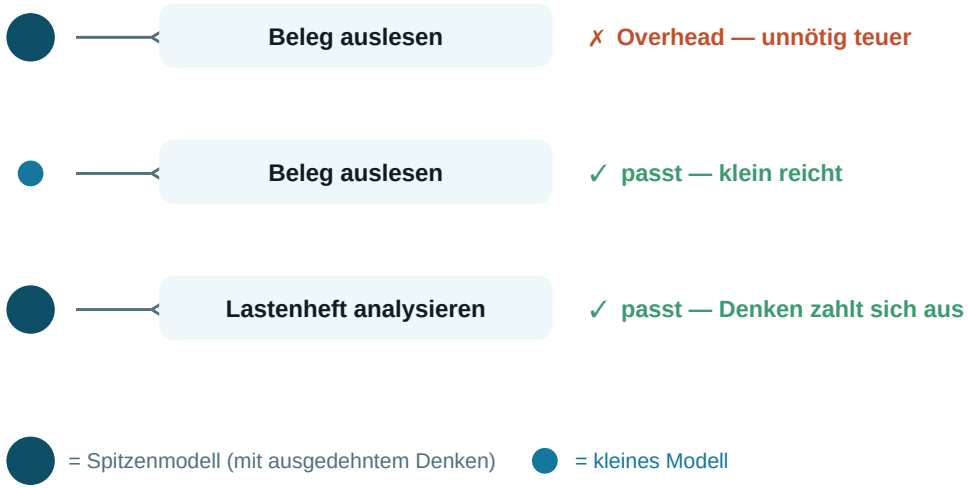
KOSTENTREIBER 4 VON 6

Teures Denken für simple Aufgaben

Overhead

Der Reflex, für alles das stärkste Modell zu nutzen — auf einfachen Aufgaben reiner Overhead.

Passt das Werkzeug zur Aufgabe?



Ausgedehntes Denken zahlt sich auf harten Aufgaben vielfach aus — auf einfachen ist es reiner Overhead, der bei jedem einzelnen Lauf anfällt.

Gegenmittel

✓ Modell je Use Case festlegen

Extraktion und Sortierung ans kleine Modell, Analyse und Abwägung ans Spitzenmodell — als einfache Tabelle, einmal im Quartal geprüft.

✓ Denkbudget begrenzen

Moderne Schnittstellen erlauben, den Denkaufwand pro Anfrage zu steuern — niedrig für Routine, hoch nur für harte Fälle.

✓ Frontier nur, wo es zählt

Das teuerste Modell dort, wo es das Ergebnis wirklich verändert — nirgendwo sonst.

Man fräst keine Unterlegscheibe auf dem 5-Achs-Zentrum.

Teures Denken gehört dorthin, wo entschieden wird — nicht in die Routine.

KOSTENTREIBER 5 VON 6

Orchestrierung vervielfacht Kosten

mehrfach bezahlt

Wie Arbeit auf Agenten, Werkzeuge und Zwischenschichten verteilt wird, verändert die Rechnung dramatisch.

Dieselben Tokens, mehrere Kassen

Immer mehr Werkzeuge in der Kette rechnen selbst pro Token ab — dieselben Tokens werden auf dem Weg zum Modell mehrfach verarbeitet und mehrfach bepreist.

Dazu kommt: Wie eine Aufgabe auf Agenten und Werkzeuge zerlegt wird, verändert die Kosten dramatisch — bei identischem Geschäftsergebnis.

Gegenmittel

✓ Gesamtkosten je Lauf rechnen

Einmal durchkalkulieren, was ein Lauf über alle Stationen kostet — nicht nur der Modell-Aufruf.

✓ Schlanke Übergaben

Zwischen Teilschritten nur Ergebnisse weitergeben („hier die 5 Werte“) — nie den ganzen Verlauf.

✓ Wenige Zwischenschichten

Jedes Werkzeug, das pro Token abrechnet, multipliziert die Rechnung — einfache Architektur ist auch die billige.

Die Rechnung entsteht in der Architektur, nicht im Modell.

Dieselben Tokens werden auf dem Weg zum Modell oft mehrfach bepreist.

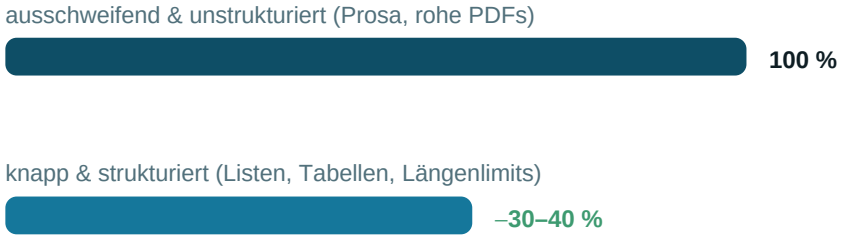
KOSTENTREIBER 6 VON 6

Informationsstruktur treibt Verbrauch

–30–40 %

Wie Information formuliert und formatiert ist, bestimmt die Token-Zahl — bei gleichem Inhalt.

Derselbe Inhalt — zwei Rechnungen



Wie Information formuliert und formatiert ist, bestimmt die Token-Zahl — ohne dass sich am Inhalt etwas ändert. Sogar die Sprache zählt: nicht-englischer Text wird in mehr Tokens zerlegt als englischer mit derselben Bedeutung.

Gegenmittel

✓ Knapp formulieren

Prägnante Anweisungen und Längenlimits für die Ausgabe sparen 30–40 % — ohne messbaren Qualitätsverlust.

✓ Strukturierte Formate

Listen und Tabellen statt Fließtext, aufbereitete Daten statt roher PDFs.

✓ Kosten sichtbar machen

Wer sieht, was sein Prompt kostet, formuliert von selbst sparsamer — Transparenz je Lauf genügt.

Präzise formuliert ist halb gespart.

Knappe, strukturierte Ein- und Ausgaben sparen 30–40 % — ohne Qualitätsverlust.