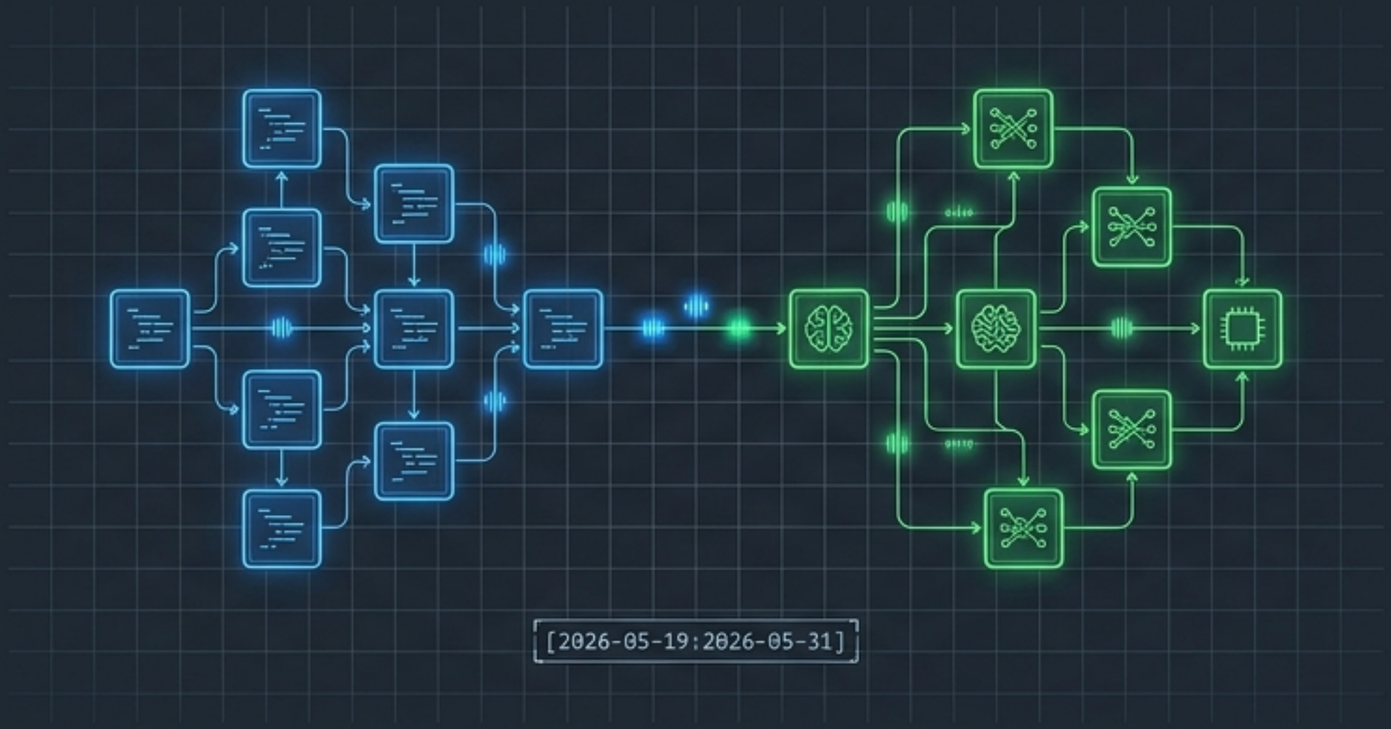
SEM-Agentic: Erkenntnisse einer Feasibility Study

Architektur, Pitfalls und AI-Engineering-Praktiken für den Bau von Agentic AI



- Systematischer Bau deterministischer KI-Orchestrierung
- Messbare Architektur-Entscheidungen statt Prompt-Hype
- Das Playbook vom Prototyp zur Produktionsreife

Die radikale Trennung zwischen KI als Produkt und KI als Bau-Werkzeug

Der SEM-Agent

KI als Produktkomponente

- Selektiver, kostenbewusster Einsatz
- Fokus auf absolute Auditierbarkeit
- LLM nur für Rationale und Randfälle

Claude Chat / Code

KI als Bau-Werkzeug

- Intensiver, explorativer Einsatz
- Fokus auf Architektur-Sparring und schnelle Iteration
- LLM kompensiert fehlendes Detailwissen durch Geschwindigkeit

Das LLM ist austauschbar, die deterministische Library ist das echte Asset

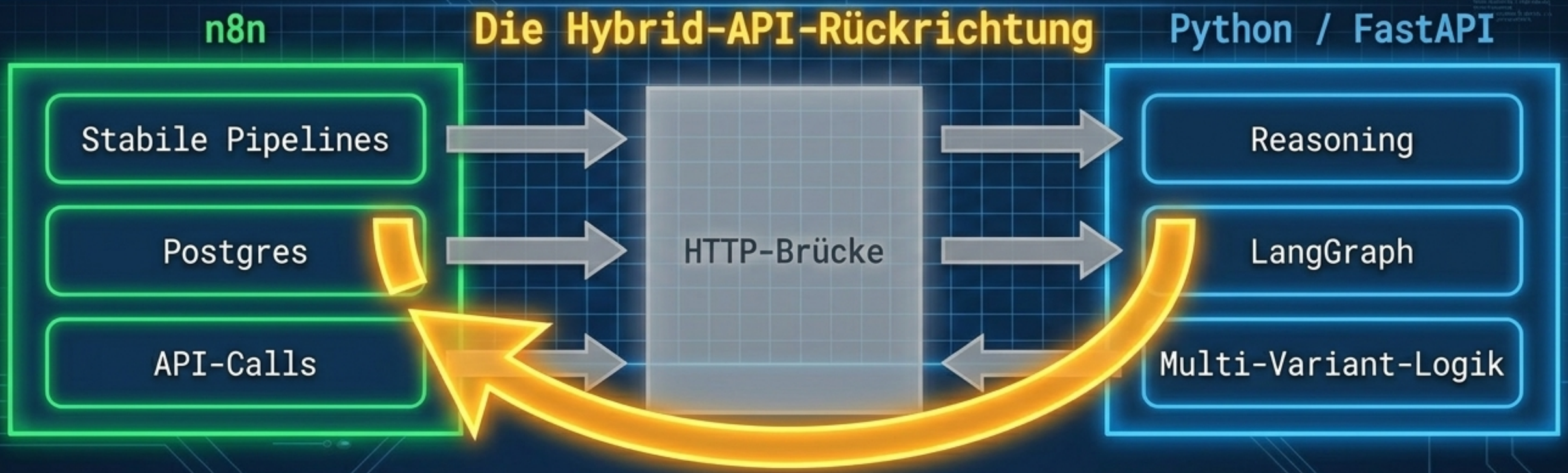
[Best-Practice-Library (Tacit Knowledge) + Lokales Reasoning (0€)] > **[Pure-Vendor-LLM-Lösung]**

Der empirische Beweis (Tag 4): Architektur (Voll-v2) isoliert steigerte den F1-Score um **+244%**.

Vendor-Rolle: Vendor-LLMs wandern nicht aus dem Produkt, sondern in die **Onboarding-Schicht** (Tacit-Knowledge-Extraktion) und Edge-Cases.

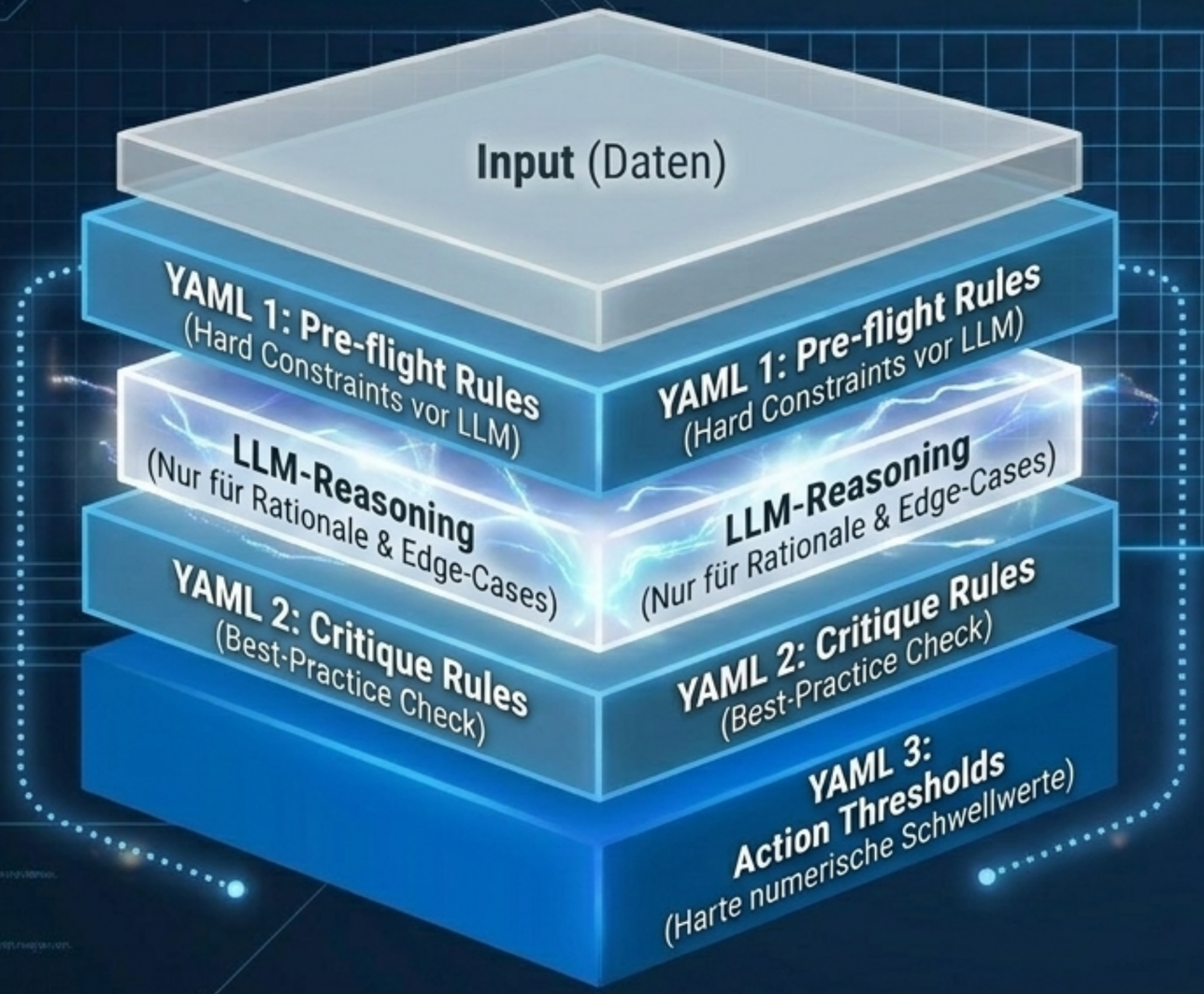
Kernaussage: Marketers sind Tool-Profis, können aber Best-Practices oft nicht formalisieren. Die Library füllt diese Lücke **deterministisch** auf.

Die Hybrid-API-Architektur trennt Orchestrierung und Sub-Agent-Komposition



Der Python-Coordinator orchestriert die Logik und triggert n8n-Workflows (als Sub-Agentten) via Webbhook. n8n für das stabile Backend, Python für hochiterative Reasoning-Loops.

Das deterministische Sandwich-Modell umschließt den stochastischen LLM-Kern



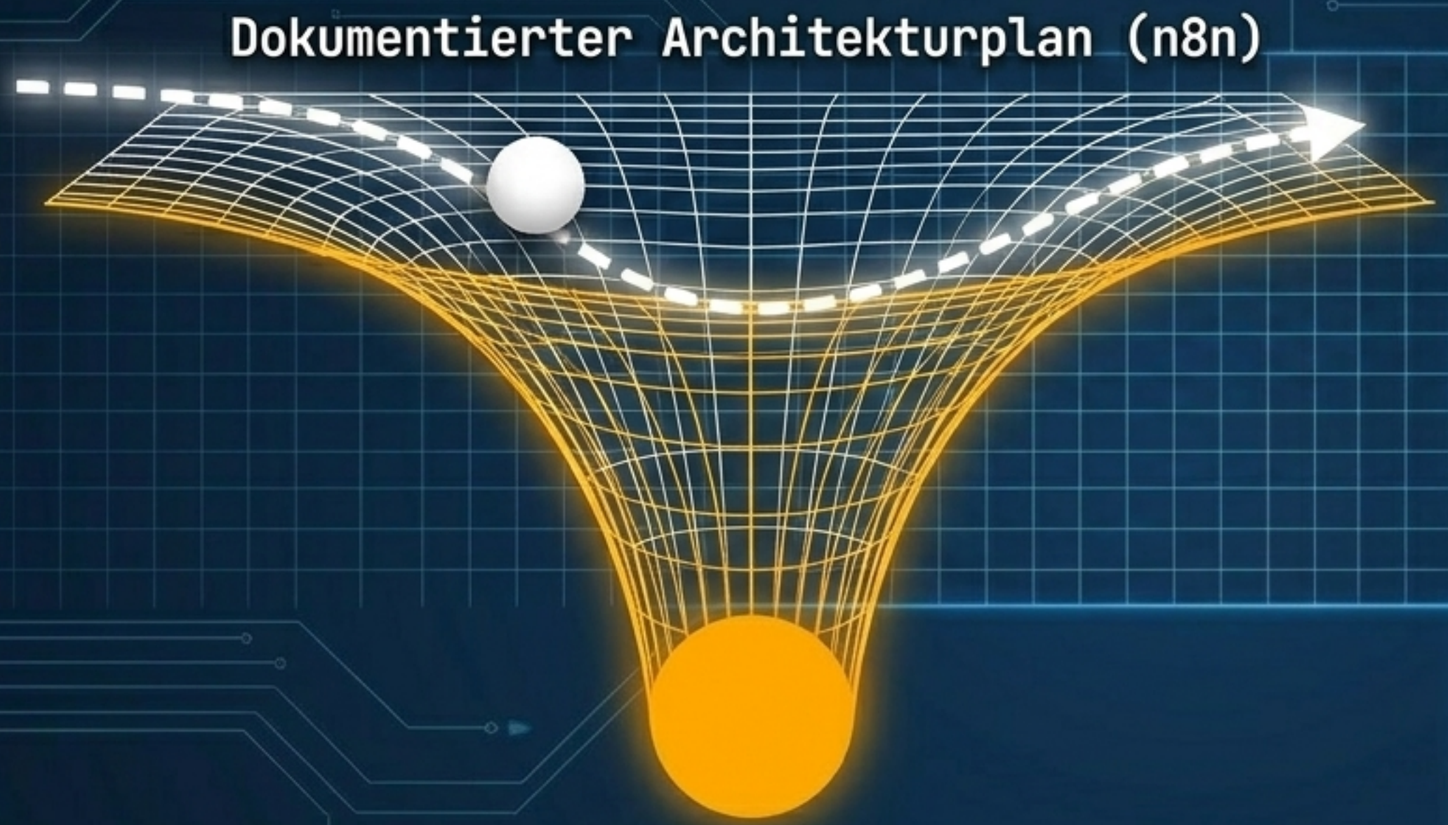
Insight: Wenn das LLM eine Hard-Constraint überschreiten kann, gehört die Regel nicht in den System-Prompt, sondern deterministisch in eine YAML-Datei. Dies macht LLM-Modelle austauschbar (+244% F1-Score isoliert durch Architektur).

Lokale Modelle vs. Vendor LLMs: Eine Frage von lokalen vs. globalen Constraints

	Qwen2.5:7b (Lokal, 0€)	Claude Sonnet 4.5 (Vendor, ~3-5¢)
Lokale Constraints (Keyword-Level)	**PASS** (F1 0.430)	**PASS** (F1 0.430)
Globale Constraints (Aggregat / Budget)	**FAIL** (Summen variieren 50-100%)	**PASS** (100.0% Konsistenz in 4/4 Runs)

Takeaway: Reasoning-Backends müssen pro Prozess-Schritt konfigurierbar sein. Lokale Modelle für hochvolumige Single-Tasks, Vendor-LLMs zwingend für Aggregat-Logik.

Pitfall: Architektur-Erosion durch Werkzeug-Schwerkraft



Das Problem
Ein AI-Assistent schreibt Python-Dateien als direkten Edit. Der Bau von n8n-Workflows erfordert UI-Interaktion und Tool-Reibung. Ohne Gegenkraft gewinnt das Tool-Bias.

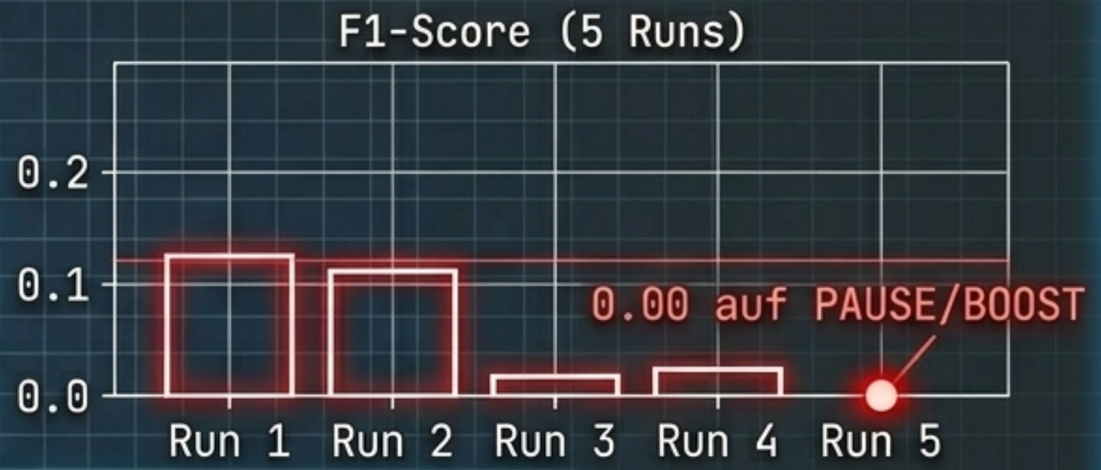
Die Gegenmaßnahme
Aktive Durchsetzung der Architektur pro Arbeitspaket. Formulierung: Baue den Workflow in n8n – Nicht: Schreibe ein Skript.

Die Illusion des Bauchgefühls: Warum Prompt-Tuning ohne Eval-Set Selbsttäuschung ist

Das Bauchgefühl

Sieht plausibel und flüssig aus. Fühlt sich nach guten Empfehlungen an.

Die gemessene Realität



F1 macro median: 0.125

Insight: Ohne historisches Backtesting (Eval-Set) ist jede LLM-Verbesserung ein Bauchgefühl-Verkauf. Risikoaversion von Modellen ist eine messbare strukturelle Eigenschaft, kein Zufall.

Der 30x Speedup: Wie Stage-1-Skizzen teures Re-Designing verhindern

Vergleich: Schätzung vs. Realität

Geschätzte Bauzeit (Bauchgefühl)
4-5 Stunden

Reale Bauzeit (mit Vorab-Skizze)
75 Minuten

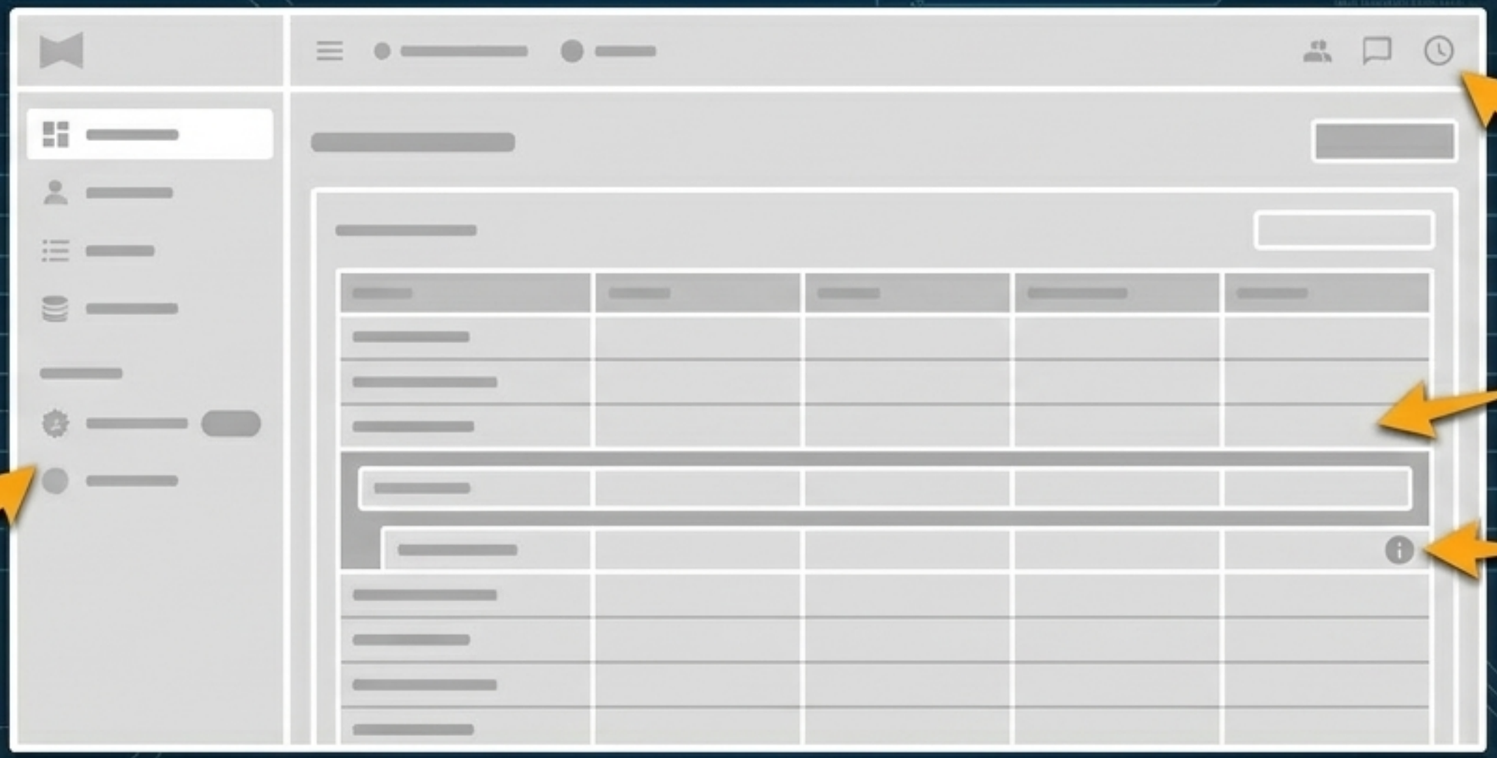
Diagnose-Bug (ohne Skizze)
90 Minuten (geschätzt 20)

Der Mechanismus

1. Vorab-Skizze klärt Architektur-Lücken (Spec vs. Code).
2. Hard-Fail-Skelett zwingt zu Fehlerpfaden zuerst.
3. Cache-Replay macht Iterationen billig.

Resultat: Reine Replikation skaliert Faktor 6x bis 30x schneller durch ein 10% Vorab-Investment in Skizzen.

Operator-Empathie: UI-Details als Vorinvestition in operative Sicherheit



Der Grundsatz: Jeder Drill-Down-Klick ist ein **verhinderter Stakeholder-Anruf** in zwei Monaten.

Die Reihenfolge: **Operationalisierbarkeit** kommt vor Visualisierung. **Daten-Persistenz** (state.json) muss zwingend gebaut sein, bevor das Dashboard gerendert wird.

Minimalistischer Human-in-the-Loop: Pragmatismus statt Over-Engineering

BullMQ / Temporal Workflow Engine

Die 3 simplen HitL-Primitives

Two-Phase DB-State
(Pending -> Approved/Rejected)

Reject-Re-Run
(Asynchrones Feedback via UI-Textarea)

Feedback-Adoption
(Prompt-Integration des Feedbacks in den Re-Run)

Takeaway: Für kurze Pipelines (Minuten) sind asynchrone Datenbank-States völlig ausreichend. Langlaufende Prozess-Blocker sind nicht notwendig.

Testing-Realität & Observability: Wenn Token-Kosten monetär signifikant werden

Die Bauchgefühl-Lüge

Geschätzte Coverage

Die gemessene Realität

Reale Coverage mit pytest-cov

`@pytest.fixture(Real-DB) + Auto-Cleanup`

- **Coverage-Lüge:** Schätzung ohne Messung ist toxisch.
- **Real-DB Fixtures:** Funktionieren extrem schnell und CI-robust (9 Cases in 16 Minuten). Mocking ist oft unnötig komplex.
- **Observability-Pflicht:** Sobald Agenten Vendor-LLMs im Loop nutzen, werden Kosten real (0.50-2.00 pro Run). **Cost-Awareness** ist eine **Hard-Fail-Metrik**.

Lernkultur vs. Fehlerkultur: Wie man LLM-Memory operationalisiert

Fehlerkultur (Ineffektiv)

- Fokus auf Schuld: Selbstgeißelung.
- Sprache: Ich habe vergessen, die Regel anzuwenden.
- Ergebnis: Der Bug wiederholt sich im nächsten Bauzyklus.

Lernkultur (Operationalisiert)

- Fokus auf Mechanismus: Warum hat das System-Memory versagt?
- Sprache: Ort der Regel-Verankerung anpassen.
- Ergebnis: Bug-Erkenntnisse als imperative Regeln in die dedizierte Rule-Zone schreiben.

Insight: Erkenntnisse dürfen nicht als historische Projekt-Notizen gespeichert werden. Sie müssen zwingend als verhaltenssteuernde Prompts verankert werden.

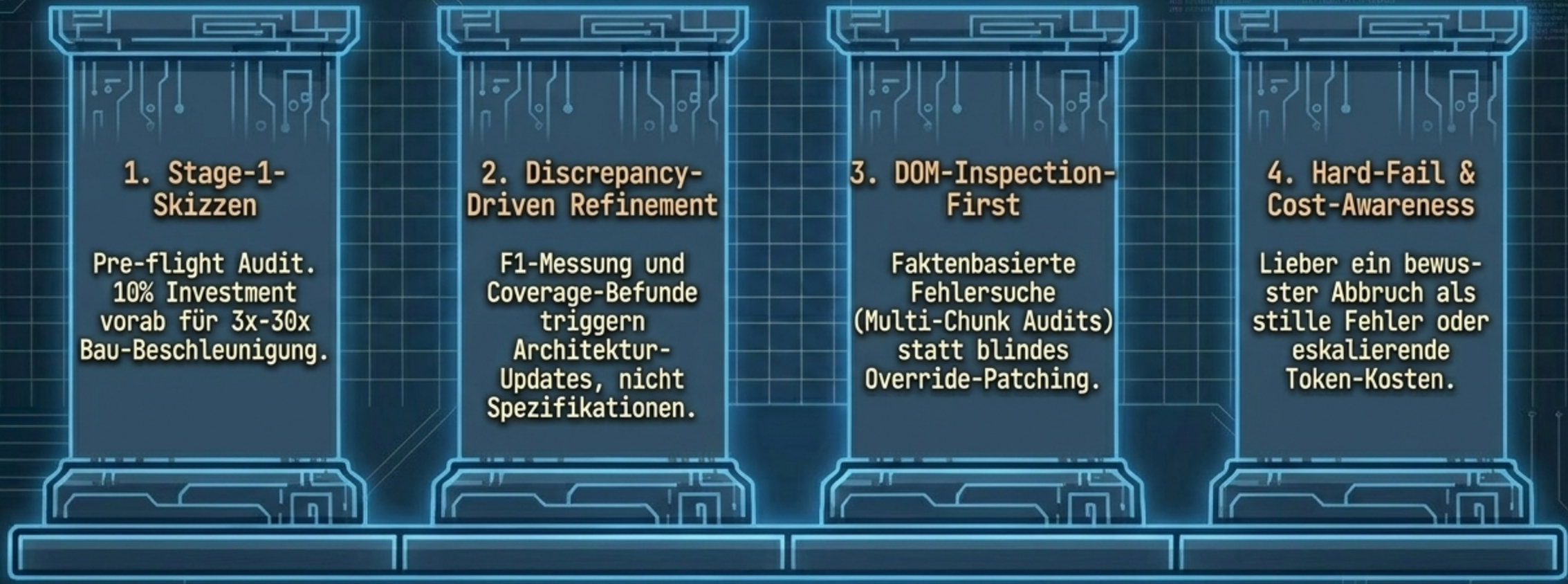
Eval-Disziplin: Der Tag-13-Moment und der Wert des Hard-Fails

Die Situation:
Ein eleganter, überzeugender Claim des LLMs im Abschluss-Proof sah perfekt aus (ADR-015 bestanden).

Die Diagnose:
Der referenzierte Datenpunkt existierte in der synthetischen Ground-Truth gar nicht. Das Modell hatte die Zahlen hochplausibel halluziniert.

Die Konsequenz:
Den PASS zurückzuziehen ist das stärkste Signal für echte Eval-Disziplin. Zahlen echt heißt nicht Kausalität Kausalität echt.

Das AI-Engineering Playbook: 4 Kernpraktiken für Produktion



Takeaway: Der Geschäftswert von Agentic AI liegt nicht in Prompt-Magie, sondern in präziser, auditierbarer Systemarchitektur und empirisch validiertem Engineering.