10 Cent vs. 4 Euro: Wer billig kauft, kauft doppelt.

Der Kostenunterschied zwischen einem 10-Cent-Prototyp und einer 4-Euro-Applikation ist irrelevant — wenn die 4-Euro-Version die einzige ist, die wirklich deploybar ist.

2026-04-07

(english below)

Deutsch

Ein typischer AI Code Generator kostet ein paar Cent.

Ein APA-Build kostet ein paar Euro.

Hier ist warum — und warum das keine Rolle spielt.

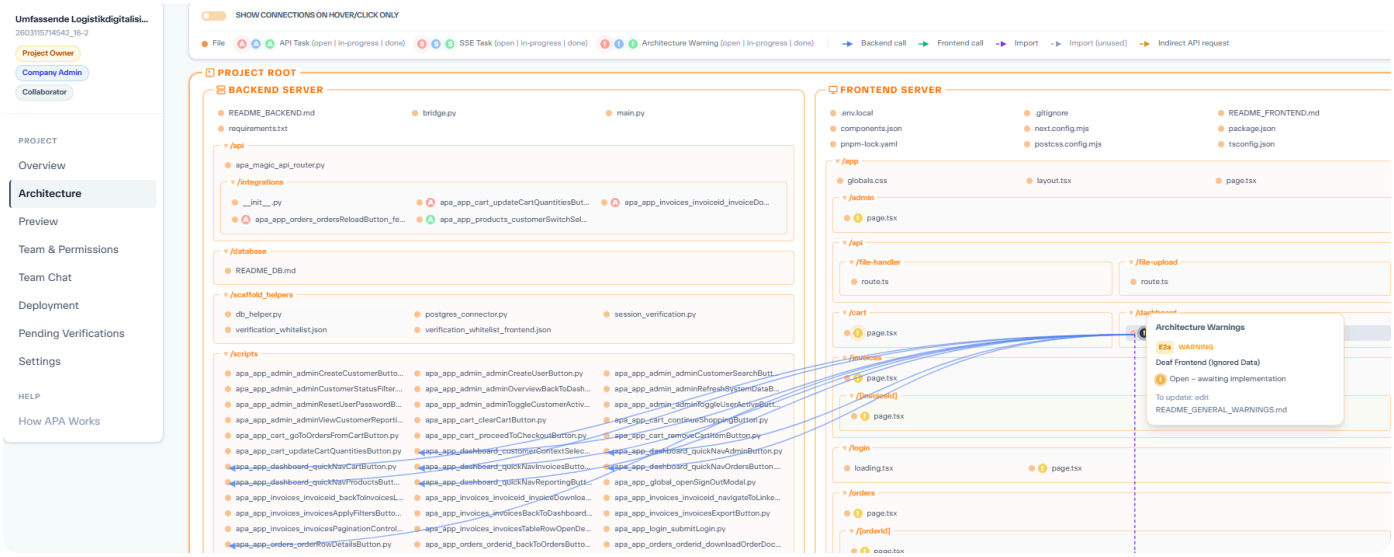
APA verarbeitet pro Build mehr als **1 Million Token**. Das klingt nach Ineffizienz. Es ist das Gegenteil. Es ist eine bewusste Entscheidung.

Was in diesen 1M+ Token passiert:

Die Architektur wird **konzipiert bevor sie gebaut wird.** Jede Seite, jede Komponente, jedes UI-Element wird typisiert und strukturiert. Datenflüsse zwischen Seiten werden als explizite Control Flows designt — wer schickt was wohin, und wann — bevor eine einzige Datei geschrieben wird.

Nach der Code-Generierung baut die Pipeline einen **Architecture Graph** — eine vollständige Karte jeder Frontend-Backend-Verbindung in der Applikation. Dann inspiziert sie ihn auf strukturelle Fehler.

Wenn Fehler gefunden werden — ein fehlendes Backend-Script, ein gebrochener Daten-Vertrag — werden **dedizierte Repair-Agents** entsandt. Sie sammeln Kontext, treffen Entscheidungen, patchen die spezifische Datei und wiederholen die Inspektion. Die Schleife läuft bis die Architektur sauber ist.



Dann wird jede Frontend-Datei **parallel reviewed** auf Runtime-Bugs und UX-Probleme. Markierte Dateien werden automatisch gefixt.

Das ist was Senior-Entwickler tun: erst designen, dann bauen, dann reviewen, dann fixen. Wir haben diesen Prozess in eine Pipeline repliziert.

Der Kostenunterschied zwischen einem 10-Cent-Prototyp und einer 4-Euro-Applikation ist irrelevant wenn die 4-Euro-Version die ist die man tatsächlich deployen kann.

> *Hashtags: #KI #AI #LLM #GenerativeAI #SoftwareArchitektur #BuildInPublic #APA #TechStartup*

English

A typical AI code generator costs a few cents.

An APA build costs a few euros.

Here's why — and why it doesn't matter.

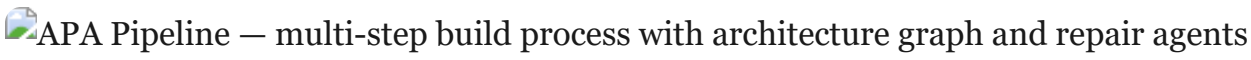
APA processes more than **1 million tokens per build**. That sounds like inefficiency. It's the opposite. It's a deliberate choice.

What happens in those 1M+ tokens:

The architecture is **conceptualized before it is coded.** Every page, component, and UI element is typed and structured. Data flows between pages are designed as explicit control flows before a single file is written.

After code generation, the pipeline builds an **architecture graph** — a full map of every frontend-to-backend connection in the application. Then it inspects it for structural errors.

When errors are found — a missing backend script, a broken data contract — **dedicated repair agents are dispatched.** They gather context, make decisions, patch the specific file, and rerun the inspection. The loop continues until the architecture is clean.



Then every frontend file is **reviewed in parallel** for runtime bugs and UX issues. Flagged files are automatically fixed.

This is what senior developers do: design first, build second, review third, fix fourth. We replicated that process in a pipeline.

The cost difference between a 10-cent prototype and a 4-euro application is irrelevant when the 4-euro version is the one you can actually ship.

> *Hashtags: #KI #AI #LLM #GenerativeAI #SoftwareArchitektur #BuildInPublic #APA #TechStartup*