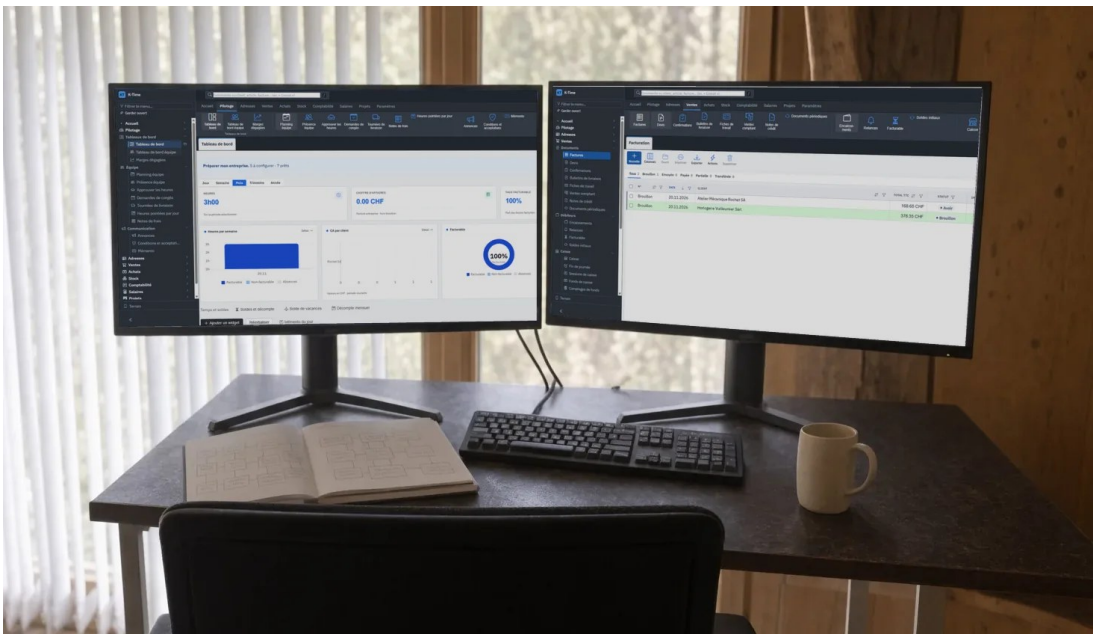
CHANTIER EN COURS

Un ERP et sa fabrique

Nous développons K-Time, un logiciel de gestion pour les PME suisses, avec une équipe d'agents d'IA. Le plus dur n'est pas de leur faire écrire du code, mais de garder la main et de prouver que ça marche.

Publié le 26 novembre 2026 — Imprimé le 27 novembre 2026



Wir entwickeln K-Time, eine Verwaltungssoftware für Schweizer KMU, mit einem Team aus Agenten der künstlichen Intelligenz. Das Schwierigste ist nicht, sie Code schreiben zu lassen. Es ist, die Kontrolle über das zu behalten, was sie erzeugen, und zu beweisen, dass es funktioniert.

Warum ein ERP

Buchhaltung, Löhne, Einkauf, Verkauf, Lager, Projekte, Einsätze, Fakturierung: Ein ERP lässt Fachbereiche nebeneinander bestehen, die jeweils ihre eigenen Regeln haben. Ein Eingriff beim Lager kann den Wert der Waren ändern, damit die Rechnung, damit die Buchhaltung. Schwierig ist nicht jedes Modul für sich, sondern das, was zwischen ihnen geschieht.

Deshalb haben wir es als Versuchsfeld gewählt. Wenn eine Arbeitsmethode bei einem ERP trägt, trägt sie bei vielen anderen Programmen. Unsere Praxis mit Verwaltungssoftware reicht bis 1999 zurück; K-Time erlaubt es uns, zugleich das Werkzeug zu haben, das wir für unsere Kunden wollen, und diese Methoden in grossem Masssstab auf die Probe zu stellen.

Die Software selbst bleibt klassisch: PHP, SQL und das Web. Die künstliche Intelligenz kommt vor allem bei ihrer Herstellung zum Einsatz.

Die Fabrik

Einem Agenten eine Aufgabe zu übertragen, ist einfach. Mehrere gleichzeitig, an derselben Software, über Wochen hinweg arbeiten zu lassen, ohne den Faden zu verlieren, ist ein anderes Handwerk. Deshalb haben wir gebaut, was wir die Fabrik nennen.

- Die Arbeit wird in **Lose** aufgeteilt: ein Bereich, ein Vertrag, Abnahmekriterien.
- Jeder Agent arbeitet auf seinem Branch, in einer isolierten Umgebung, auf einer von mehreren Maschinen und mit dem am besten geeigneten Modell.

- Eine Änderung durchläuft automatische Kontrollen, dann Integrationskontrollen und, wenn nötig, eine unabhängige Bewertung.
- Eine einzige Merge-Warteschlange bringt die Lose ins Produkt.
- Jeder gefundene Mangel wird zu einem neuen Los. Die Schleife läuft ununterbrochen.

Was die Zahlen sagen

Per 26. November 2026 verzeichnet das Register der Fabrik **3'254 Lose**, davon **2'862 integriert** und **224 abgelehnt**. Zwischen dem Start eines Loses und seiner Ankunft am Merge-Tor vergehen im Median **55 Minuten**; bei den 710 Losen, die seit dem 21. November integriert wurden, ist diese Dauer auf **40 Minuten** gesunken.

Ein Los ist oft wenig: im Median 26 Codezeilen und 232 Testzeilen. Das Verhältnis zwischen beiden sagt das Wesentliche unserer Methode: Man schreibt viel mehr, um zu prüfen, als um zu machen.

Diese Zahlen messen weder die Qualität der Software noch ein Äquivalent in Entwicklerstunden. Sie zeigen einen Rhythmus und eine Granularität.

Wer programmiert, ist nicht, wer urteilt

Ein Agent meldet gern «fertig», obwohl die Arbeit erst zur Hälfte getan ist. In einem ERP beweisen eine Seite, die angezeigt wird, oder ein Test, der besteht, nicht, dass man damit arbeiten kann.

Wir haben deshalb **Judges** eingeführt, die eine betriebliche Rolle spielen: Buchhalter, Verkäufer, Einkäufer, Administrator. Sie starten wenn möglich auf einer frischen Installation, mit bekannten Daten und ohne gelesen zu haben, was der Entwickler von seiner Arbeit hält. Der Verkäufer muss eine Offerte erstellen, ausdrucken, wiederfinden, duplizieren und in eine Rechnung umwandeln. Der Buchhalter muss eine Buchung erfassen, eine Rechnung verbuchen und eine Mehrwertsteuerabrechnung ausgeben.

Was der Judge findet, wird zu einem Korrekturlos; nach der Integration beginnt er von vorn. Die Arbeit hat ihren Charakter geändert: Den Code zu erzeugen geht schnell, nachzuweisen, dass er den Bedarf erfüllt, verlangt die ganze Aufmerksamkeit.

Eine Fabrik, die sich selbst repariert

Die Fabrik ist zu einer vollwertigen Software mit eigenen Pannen geworden. Viele sind klassisch: Dateien, die von zwei Prozessen geöffnet sind, Sperren, Netzwerkunterbrüche, Git-Konflikte, Warteschlangen, Wiederanlauf nach einem Vorfall. Andere kommen von den Agenten: zu lange Verträge, zu wenig überprüfte Schlussfolgerungen, zweimal gemachte Korrekturen, eine Aufgabe, die dem falschen Modell übertragen wurde.

Jeder Vorfall dient dazu, den Prozess zu verbessern. Die nächsten Schritte: transaktional verwaltete Zustände, weniger Wartezeit, die realen Kosten jedes Loses und eine Verteilung der Aufgaben danach, was jedes Modell wirklich kann. Lokale Modelle haben darin ihren Platz: Nicht alles braucht ein Spitzenmodell.

Erfahrung macht den Unterschied

Nichts davon ersetzt Kompetenzen, die älter sind als die KI: Architektur, Datenbanken, kontinuierliche Integration, Systemadministration, Branchenkenntnis, Qualitätssicherung. Dank ihnen setzt man die richtigen Rahmenbedingungen und erkennt eine schlechte Lösung, bevor sie ins Produkt gelangt. Die Agenten vervielfachen die Ausführungskapazität; sie entbinden nicht davon, zu verstehen, was man baut.

K-Time ist in Version 0.9.1 RC73, in der Qualifizierungsphase. Übernahme der bestehenden Daten, buchhalterische Stimmigkeit, vollständig validierte Abläufe: Diese Schritte bleiben vor einem Produktivbetrieb unverzichtbar. Die Fabrik dient genau dazu, diesen Weg messbar und reproduzierbar zu machen, und sie könnte morgen auch anderen Produkten als K-Time dienen.

K-Time Bildschirm für Bildschirm · Die Fabrik: ein Jahr Entwicklung mit Agenten