
KI ist jetzt da
Und wird nicht wieder verschwinden – und jetzt?
Foto von Zanyar Ibrahim auf Unsplash

# Stefan Jacobs

- RWTH-Aachen, Diplom-Informatiker
  - Schwerpunkte Maschinenbau, künstliche Intelligenz und wissensbasierte Systeme
  - Wissenschaftlicher Mitarbeiter bei RoboCup
- Seit 2005 bei der Deutschen Bahn
  - 2005-2016 als Externer in verschiedenen Rollen
  - 2016-2021 DB Systel, Neubau des Vertriebssystems
  - 2022-heute DB Fernverkehr
- Lead technisches Innovationsmanagement von bahn.de und dem DB Navigator:
  - Wir sind zu fünft
  - Wir versuchen, 700+ Leuten zu helfen…
  - Mit dem Blick auf den Horizont
  - Wir machen keine Produktentwicklung… (???)



Privat:

- Gebürtiger Niederrheiner
- Wohnhaft in der Nähe von Groß-Gerau
- Verheiratet und 2 Kinder (5+7)

- Hobby und Beruf liegen glücklicherweise dicht beieinander…

# Disclaimer

Der folgende Foliensatz ist zu 100% durch einen Menschen erstellt worden.



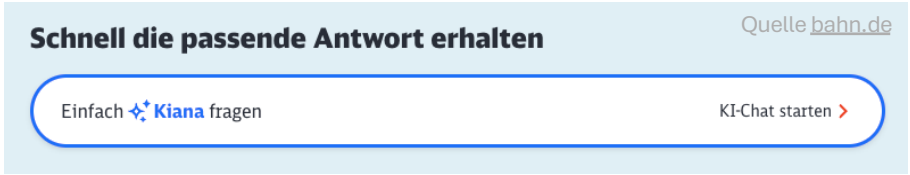
Foto von damien dufour auf Unsplash

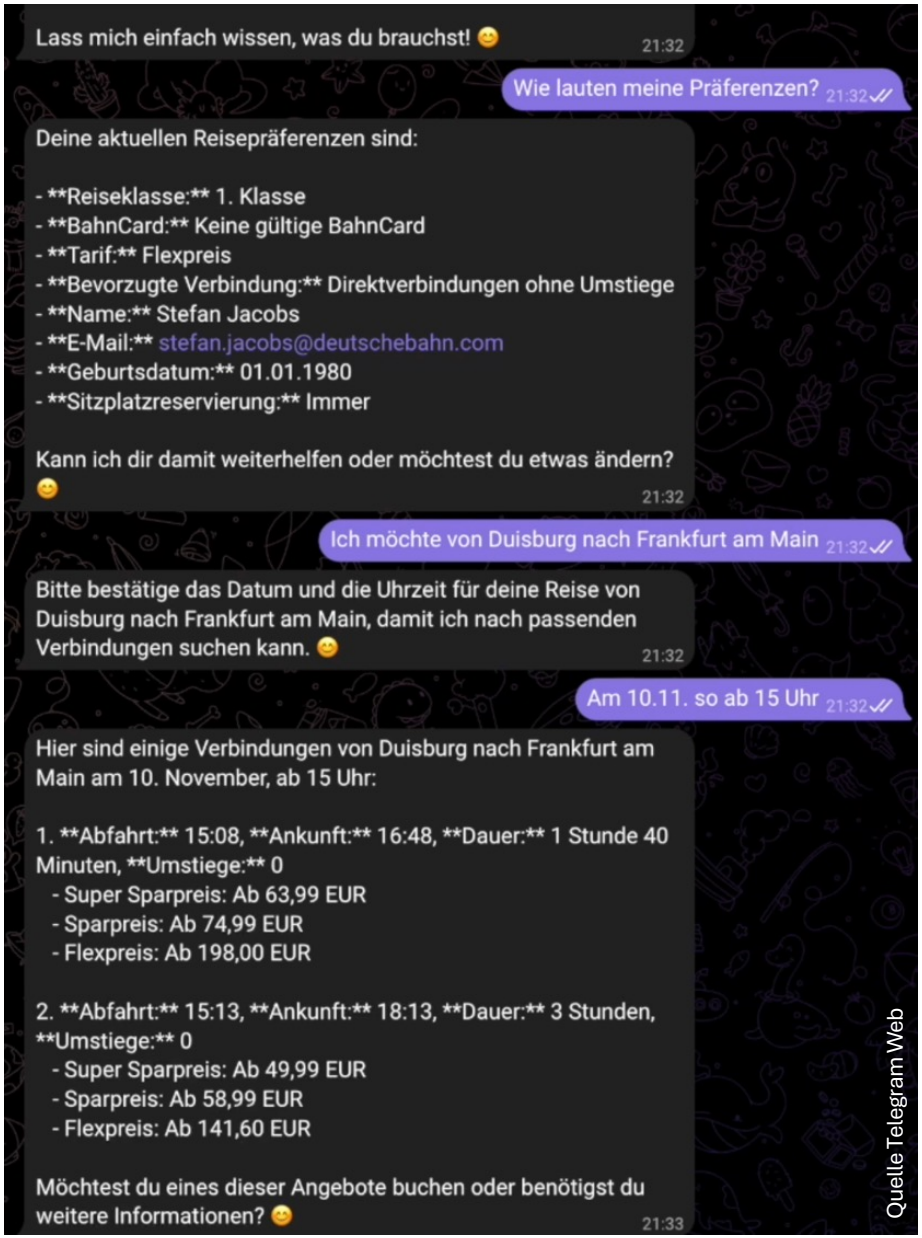
# Agenda

1. ✅ Wer ich bin
2. ➡️ Warum glaube ich, darüber reden zu können
3. Kernthesen und warum ich daran glaube
4. Was daraus folgt

# Kurzer Abholer

- 2023: Hausautomatisierung
- 2024: Externe Dienstleister
- 2024: „Dieter Bratwurst“
- 2025: „Kiana“



# Kiana Details

Ziele von Kiana:

- Vermeidung von Halluzinationen
- Korrekte Antworten

Lernen



# Technologiestack

**Backend**

- Python 3.13, uv, mise
- Google ADK 2.5 und FastAPI
- GenAI Hub:
  - OpenAI GPT-6-Luna via LiteLLM
  - (Sonnet und Haiku als Backup Modelle via LiteLLM)
- Redis (Cache) + Azure GAIDC (Wissen)
- Open Telemetry und Grafana Cloud
- Automatisierte Tests in Pipelines, DeepEval inkl. LLM as a judge

**Frontend**

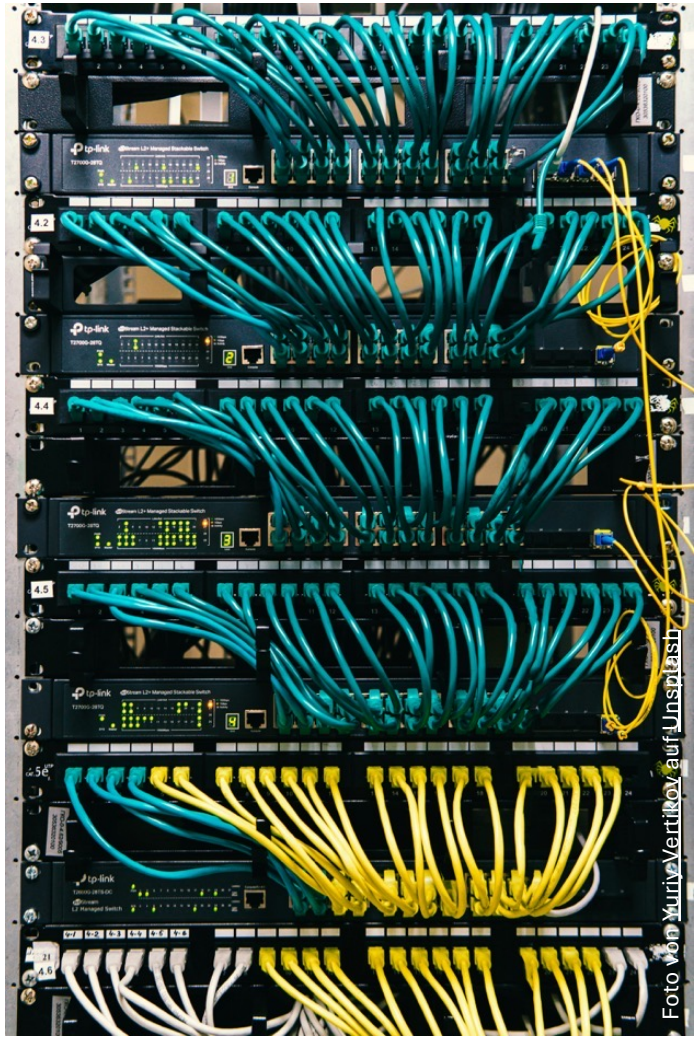
- Vue + Typescript

**Deployment**

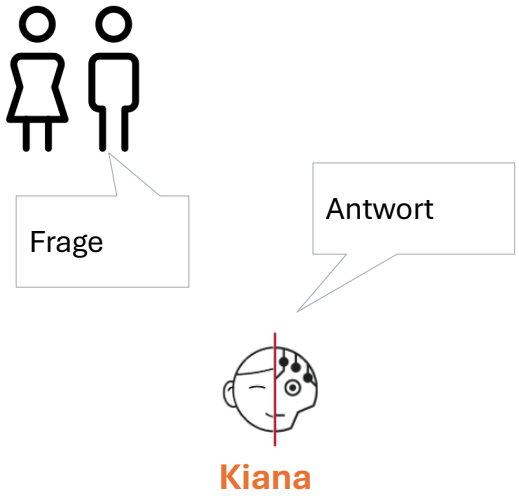
- Gitlab und Gitlab CI, Pipeship und Renovate
- Docker, Kubernetes

**Dokumentation**

- Docusaurus + Mermaid



# Multiagentensystem Zuständigkeiten



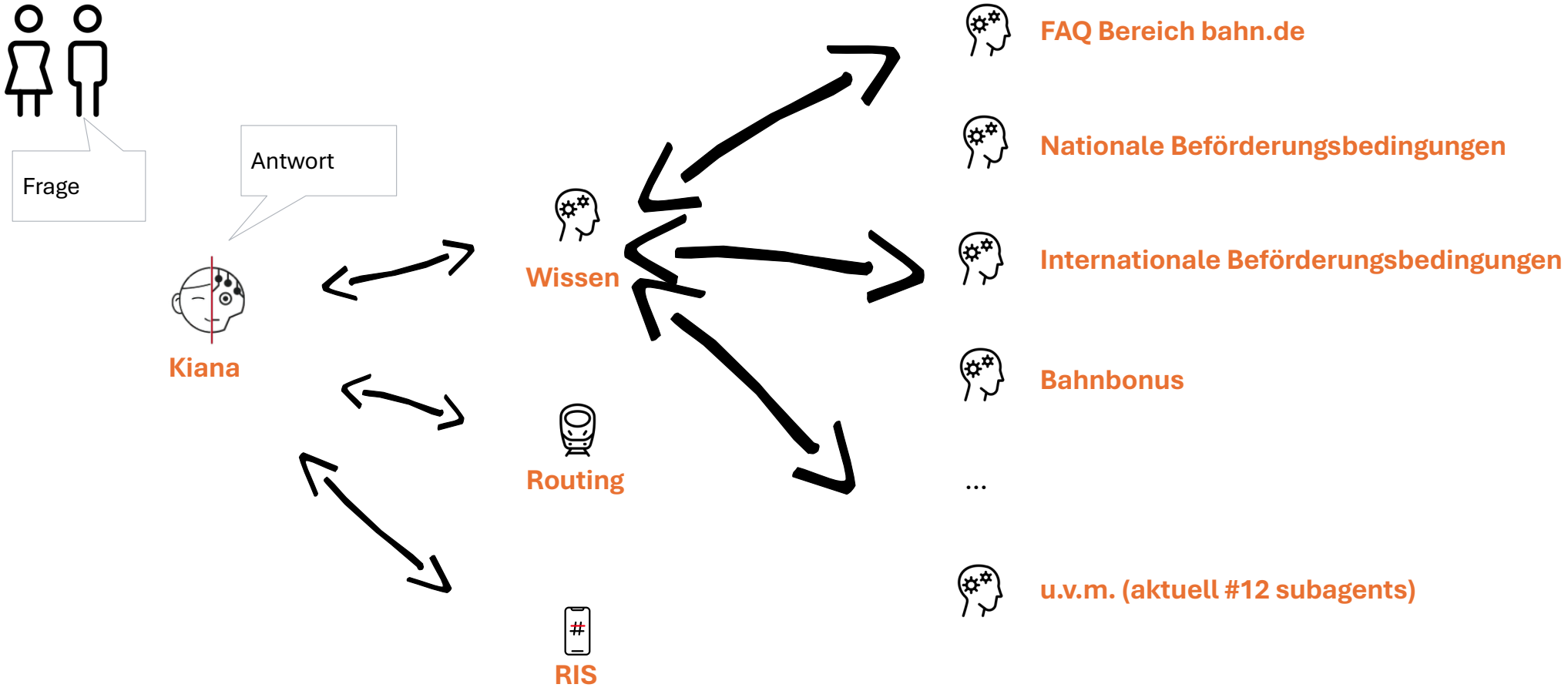

**Kiana**

- Auftrag: “Hilf dem Kunden bei seiner Anfrage“
- Hat die Kommunikationsregeln
- Hat 3 Werkzeuge, die verwendet werden können („Subagent As Tool“)
- Kleinigkeiten (Spracherkennung, Guardrails)

**Subagents**

- Fokus auf die Aufgabenstellung
- Vorgefilterte und reformulierte Anfragen durch Kiana
- Generieren nur „IDs“, welche auf verifizierte Daten zeigen
- Daten werden jeweils lesbar in den Kontext injiziert

# Multiagentensystem Architektur



# Multiagentensystem Architektur

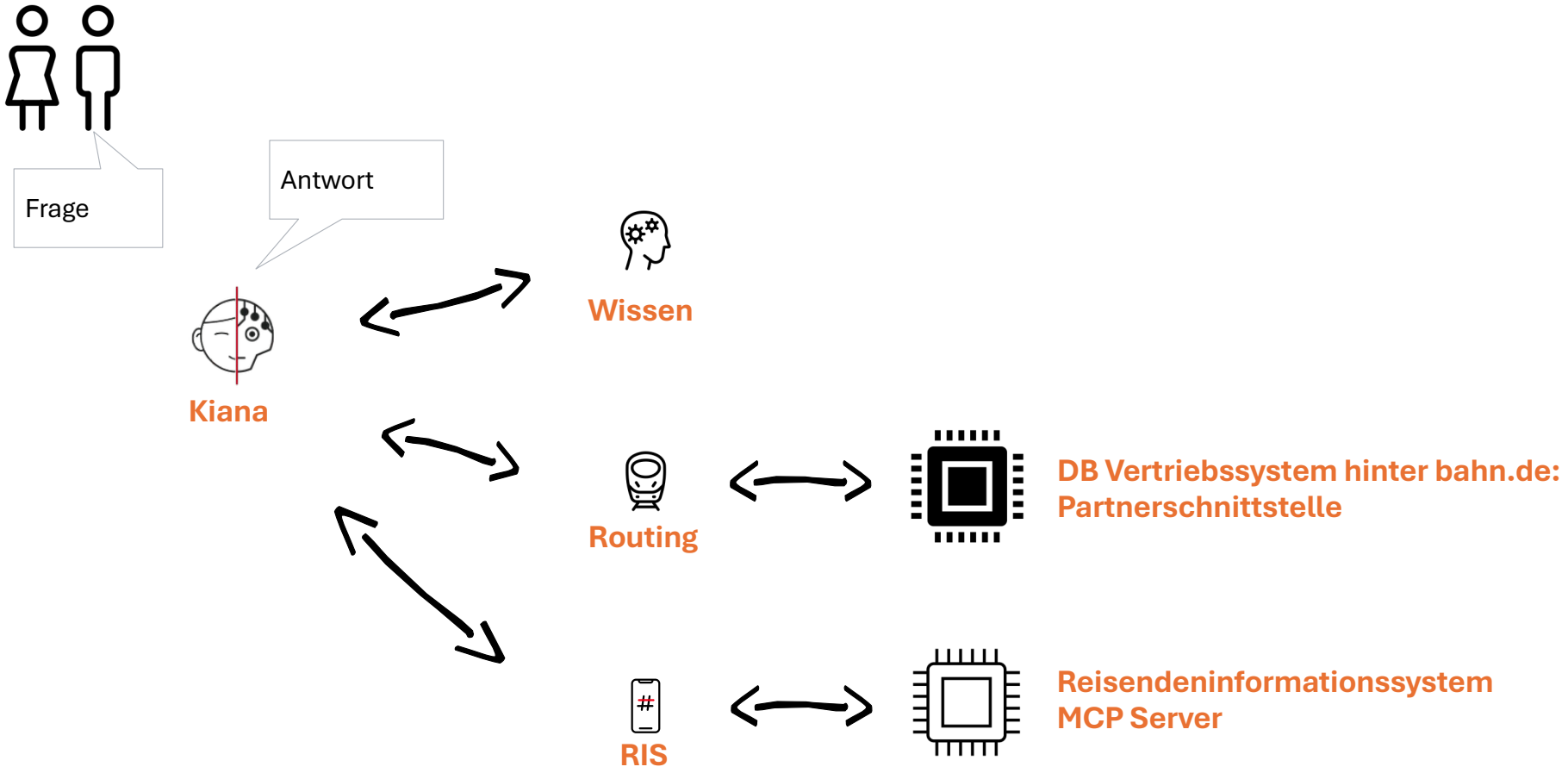




Foto von damien dufour auf Unsplash

# Agenda

1. ✅ Wer ich bin
2. ✅ Warum glaube ich, darüber reden zu können
3. ➡️ Kernthesen und warum ich daran glaube
4. Was daraus folgt

# Aufschrei 1

*„Den ganzen KI Slop versteht doch keiner!!!“*



# Kernthese 1: Wir verstehen bereits heute unsere Systeme nur punktuell



Foto von Google DeepMind auf Unsplash

- Es gibt niemanden in der Organisation, der alle Code-Zeilen Stand heute versteht
- Das Verständnis muss sich Case by Case erarbeitet werden (das ist übrigens das, was Zeit kostet)
- Eine tragfähige Architektur zahlt u.a. auch darauf ein, dass dieses Verständnis schneller zu erlangen ist (überall dieselben Muster)

Quelle „Must we understand all code all the time?“ <Tudor Girba, LinkedIn>

Besser eigentlich „Can we understand any code at any time?“

# Aufschrei 2

*„Aber ich will selbst programmieren“*

*„Die KI kann doch nicht debuggen“*



# Kernthese 2: Manuelles Coding und Debugging rechnen sich nicht mehr

- **Code schreiben:** KI-generierter Code ist bereits heute häufig schneller und günstiger als manuelle Entwicklung.
- **Code debuggen:** KI-gestützte Fehleranalyse ist zunehmend wirtschaftlicher als die manuelle Suche.
- **Code verstehen:** Die entscheidende Frage ist nicht mehr, ob ich den gesamten Code selbst verstehe – sondern ob ich seine Korrektheit und Wirkung zuverlässig absichern kann.



Foto von Long (Tiga) Nguyen auf Unsplash

# Aufschrei 3

*„Der Mist, der da erzeugt wird, den werde ich nicht verantworten“*



# Kernthese 3: Der Mensch bleibt verantwortlich – aber die Abstraktionsebene verändert sich

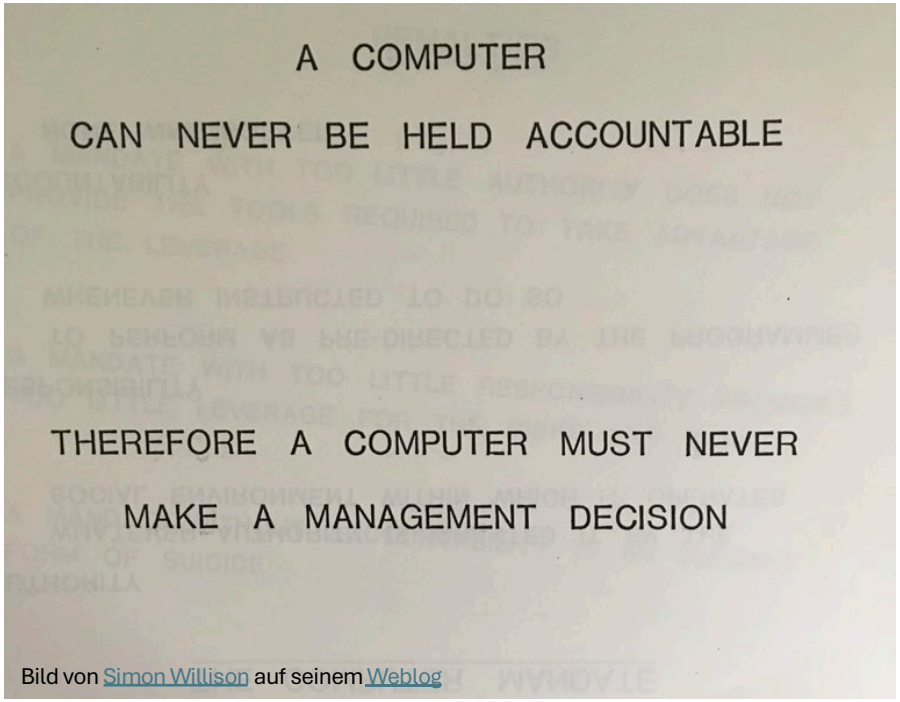



Bild von Simon Willison auf seinem Weblog

- Bereits 1979 bei IBM geschult
- Verändert sich das durch GenAI?
- Wann wird sich das verändern?

# Kernthese 3: Die Abstraktionsschicht ändert sich nicht erst heute

- Wer hier im Raum hat das letzte Mal Assembler verifiziert?
- Wer hier im Raum hat einen Editor, eine IDE oder einen Word Processor verwendet und versteht, wie dieser funktioniert?
- Ist jedem hier im Raum bewusst, dass selbst Java und C-Compiler nichtdeterministische Ergebnisse erzeugen?

Bildquelle: x.com

I don't understand this whole idiotic concept of "don't use AI for X".

Here's a new rule just for you: don't use compilers for any of your kernel submissions.

Because if you can't show that you really understand what the machine is actually doing by writing your code in assembler, then I don't want to see your code contribution. You clearly aren't competent if you can't write it that way.

And btw, you had better not use an editor to write that commit message either.

Because I need you to show me that you can actually write a good commit message without using tools that make it easy to backspace and hide the mistakes you made while typing.

Ok?

Because otherwise those tool-assisted commits of yours clearly aren't showing that you know what you are doing.

Do you not see how UTTERLY STUPID those kinds of rules would be?

So no. We're not going to make stupid rules that say "you can't use tools for X".

The kind of expectations *I* have is that people do their best and I expect people to understand what they are doing.

But that understanding has very little to do with what tools they use.

And quite honestly, in my experience, the people I respect and call intelligent are not the ones that shy away from using new and interesting tools.

Linus

# Aufschrei 4

*„Wir investieren hier so viel, wo ist die Produktivitätssteigerung?“*



# Kernthese – Wir füllen die entstehenden Lücken aus

Solow-Paradoxon (1987)

- *„Computer finden sich überall – außer in den Produktivitätsstatistiken.“*
- Quelle: Doppelgänger Podcast

Alles, was wir durch KI gewinnen, verlieren wir durch:

- „Was schon immer mal gemacht werden musste…“
- Neue GenAI „Bürokratie“
- Prozesse, die nicht für generative KI gemacht wurden




| Unternehmen | Umsatz pro Mitarbeiter | Zeitraum |
|---|---|---|
| **Anthropic** | > 6,6 Mio. USD | TTM, Sep. 2026 |
| **Microsoft** | 1,49 Mio. USD | GJ 2026 |
| **Siemens** | ca. 250.000 EUR | GJ 2025 |

# Aufschrei 5

*„Unsere Codebasis ist so komplex, da kann ich mich zurücklehnen“*



# Kernthese – Code ist nicht mehr der Moat/das Alleinstellungsmerkmal

- Früher war das einfach
- Reverse Engineer Anything (Link)
  - Aber will ich die Wartung stemmen?
  - Will ich mich um die Security kümmern?
  - Will ich den Service leisten?
- Kontext und Daten sind der Moat





Foto von damien dufour auf Unsplash

# Agenda

1. ✅ Wer ich bin
2. ✅ Warum glaube ich, darüber reden zu können
3. ✅ Kernthesen und warum ich daran glaube
4. ➡️ Was daraus folgt

# Kontext und Daten – Warum?

- Modelle haben grob zwei Phasen, in denen es „lernt“:
  - Trainingsphase
  - Inferenzphase („im Kontext lernen“)

- Die Trainingsphase ist sehr teuer
  - Selbst Finetuning ist teuer und muss mit jeder Modellgeneration erneut erfolgen



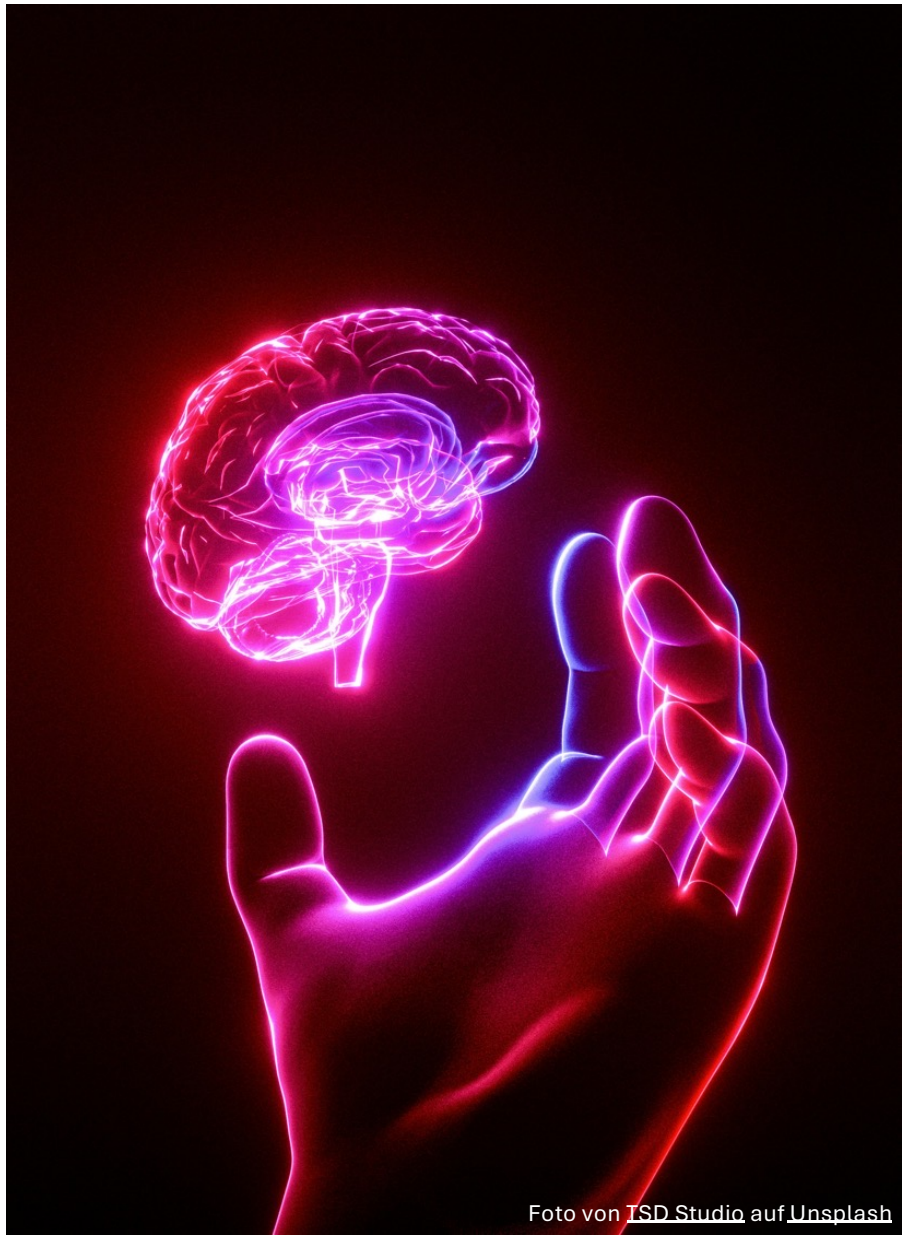
# Kontext und Daten – Warum?

Ein Beispiel mit „Spezialwissen“:

- Wie teuer ist die Fahrradmitnahme im ICE?

Was sagt ein neuer, sehr kluger, Praktikant zu der Frage?

- Das Wissen wird „aus dem Gedächtnis“ abgerufen
- Hohe Gefahr von falschen Informationen & Halluzinationen

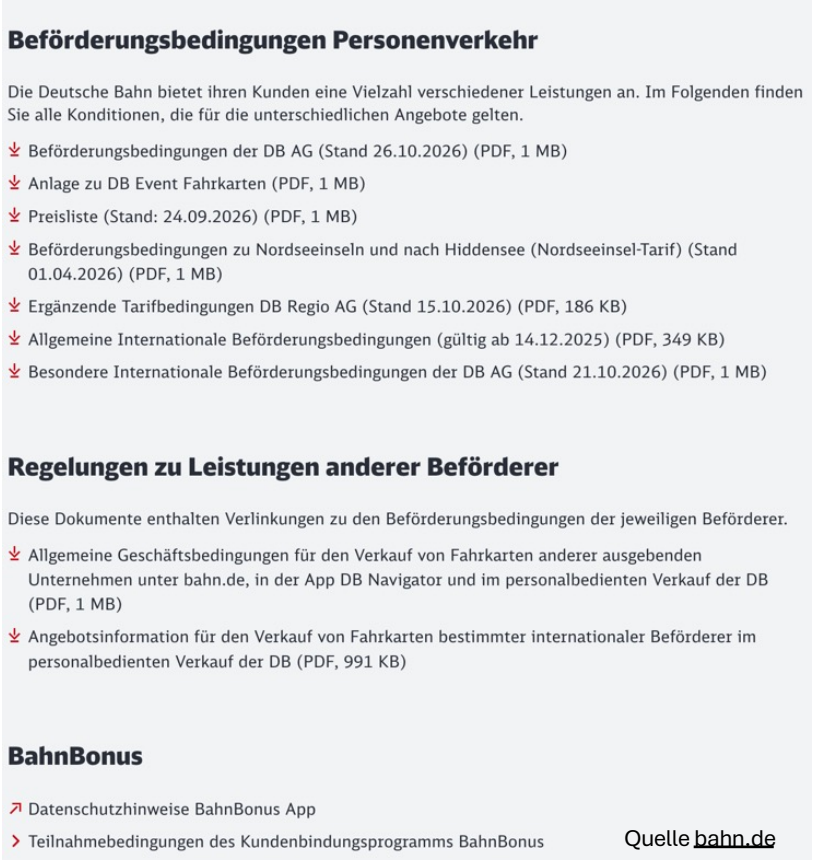


# Kontext und Daten – Warum?

Ein Beispiel mit „Spezialwissen“:

- Wie teuer ist die Fahrradmitnahme im ICE?

Was, wenn ich dem Praktikanten die folgenden Informationen dazu lege?



**Beförderungsbedingungen Personenverkehr**

Die Deutsche Bahn bietet ihren Kunden eine Vielzahl verschiedener Leistungen an. Im Folgenden finden Sie alle Konditionen, die für die unterschiedlichen Angebote gelten.

- Beförderungsbedingungen der DB AG (Stand 26.10.2026) (PDF, 1 MB)
- Anlage zu DB Event Fahrkarten (PDF, 1 MB)
- Preisliste (Stand: 24.09.2026) (PDF, 1 MB)
- Beförderungsbedingungen zu Nordseeinseln und nach Hiddensee (Nordseeinsel-Tarif) (Stand 01.04.2026) (PDF, 1 MB)
- Ergänzende Tarifbedingungen DB Regio AG (Stand 15.10.2026) (PDF, 186 KB)
- Allgemeine Internationale Beförderungsbedingungen (gültig ab 14.12.2025) (PDF, 349 KB)
- Besondere Internationale Beförderungsbedingungen der DB AG (Stand 21.10.2026) (PDF, 1 MB)

**Regelungen zu Leistungen anderer Beförderer**

Diese Dokumente enthalten Verlinkungen zu den Beförderungsbedingungen der jeweiligen Beförderer.

- Allgemeine Geschäftsbedingungen für den Verkauf von Fahrkarten anderer ausgebenden Unternehmen unter bahn.de, in der App DB Navigator und im personalbedienten Verkauf der DB (PDF, 1 MB)
- Angebotsinformation für den Verkauf von Fahrkarten bestimmter internationaler Beförderer im personalbedienten Verkauf der DB (PDF, 991 KB)

**BahnBonus**

- Datenschutzhinweise BahnBonus App
- Teilnahmebedingungen des Kundenbindungsprogramms BahnBonus

Quelle bahn.de



Foto von Andreas Fickl auf Unsplash

# Kontext und Daten – Warum?

Zentrale Frage:

Wie bekomme ich die richtigen Dokumente/Daten in den Kontext?

Zwei Antworten:

1. Entweder ist klar, wo Informationen drin stehen – dann fragen wir das Modell, was es lesen will
2. Oder es gibt einen zugehörigen Prozess, in dem klar ist, was der nächste Schritt ist



Foto von damien dufour auf Unsplash

# Agenda

1. ✅ Wer ich bin
2. ✅ Warum glaube ich, darüber reden zu können
3. ✅ Kernthesen und warum ich daran glaube
4. ✅ Was daraus folgt

# Und jetzt?

- Der Traum: „Die Software Factory“
- Der Weg: „???“
- Unsere ersten Schritte:
  - Frage den Code
  - Anforderungsassistenz
  - Automatische Code Reviews
  - „freies Lernen“



# Und jetzt?

- Der Traum: „Mein Agenten Team“
- Der Weg: „???“
- Unsere ersten Schritte:
  - Agenten Harnesses und Guardrails
  - Zugriff auf APIs im User-Kontext
  - „freies Lernen“



# Generative KI ist gekommen um zu bleiben – das geht nicht mehr weg



Foto von Kevin Grieve auf Unsplash

Wir müssen also anfangen,

- das selbst zu verstehen
- damit umzugehen
- und in eine von uns gewünschte Richtung zu kanalisieren

# Und das wird nie wieder schlechter werden