

Erfahrungen mit dem Einsatz von KI in der Entwicklung

AI in code,
Creating with innovation,
Software evolves.
-- ChatGPT

Dr. Hans-Peter Störr, IST GmbH Dresden
www.ist-software.com , www.composum.com
www.stoerr.net

Erfahrungen mit dem Einsatz von KI in der Entwicklung

Roadmap - viele Anregungen, Details oft in Links

- Allgemeines zu LLM (großen Sprachmodellen)
- IDE Integration (Github Copilot u.ä.)
- Anwendung von ChatGPT: Erstellen von Scripten, Fachkonzeptionen, ...
- LLM als Tools:
 - auf der Kommandozeile
 - LLM als Codegenerator
- Experimente

Fokus: Sprachmodelle, meist OpenAI/ChatGPT/Github Copilot, aber meist übertragbar

Über mich

Dr. Hans-Peter Störr



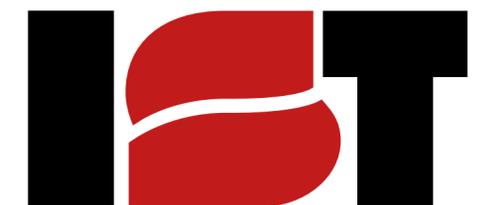
Software Architekt an der IST GmbH Dresden

Projektarbeit in vielen Projekten von Application Server bis Adobe AEM

Composum AI: LLM basierte Content-Erzeugung / Übersetzung für
Adobe AEM und Composum Pages CMS

www.ist-software.com / www.ai.composum.com

[www.stoerr.net/ai\(+ /blog.html\)](http://www.stoerr.net/ai(+ /blog.html))



Allgemeines zur KI

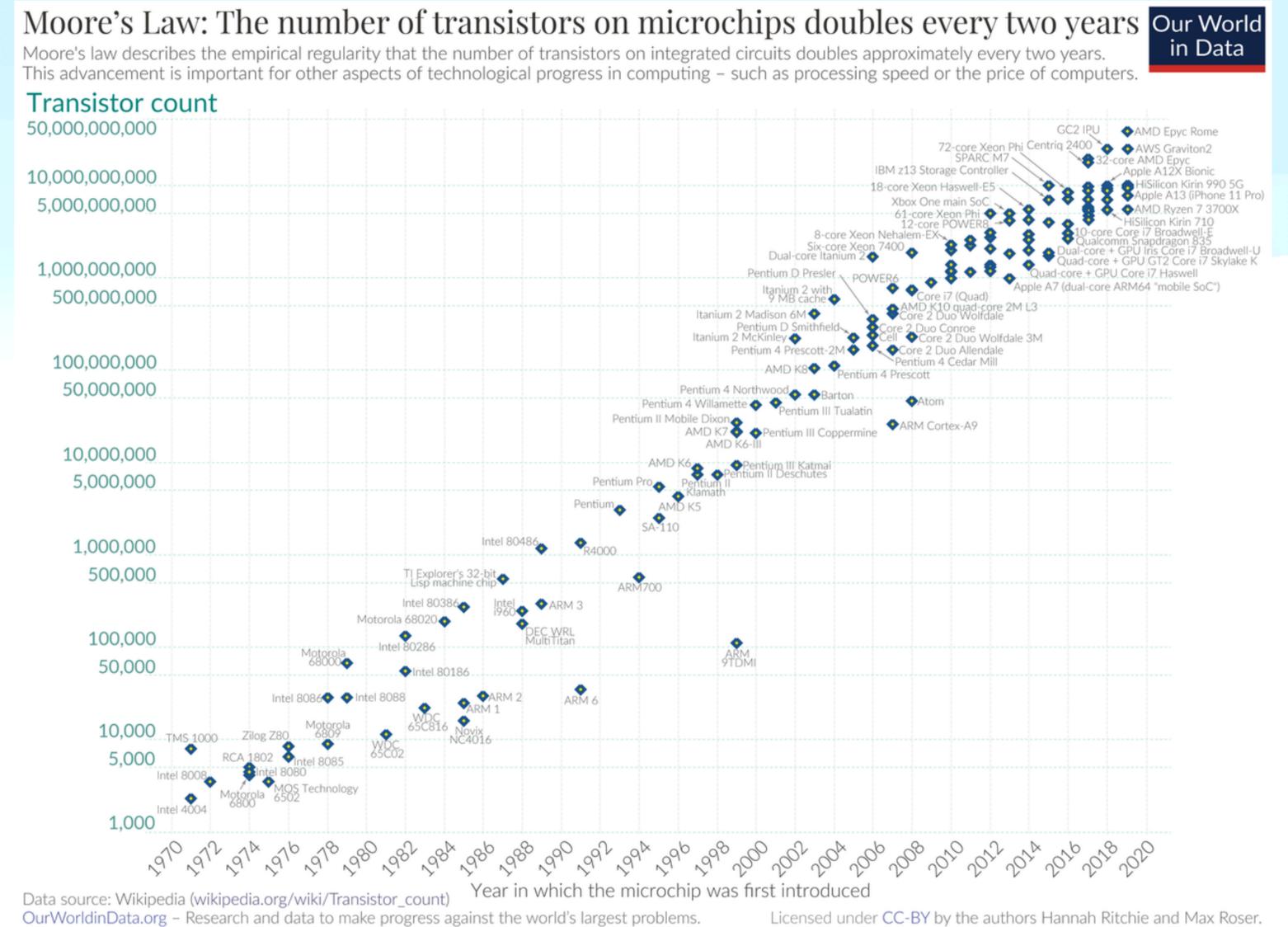


Warum KI

Einsatz von Sprachmodellen (Large Language Models)

- Steigerung von Komfort und Effizienz bei der Entwicklung
- Wissen, Kreativität
- Spassfaktor

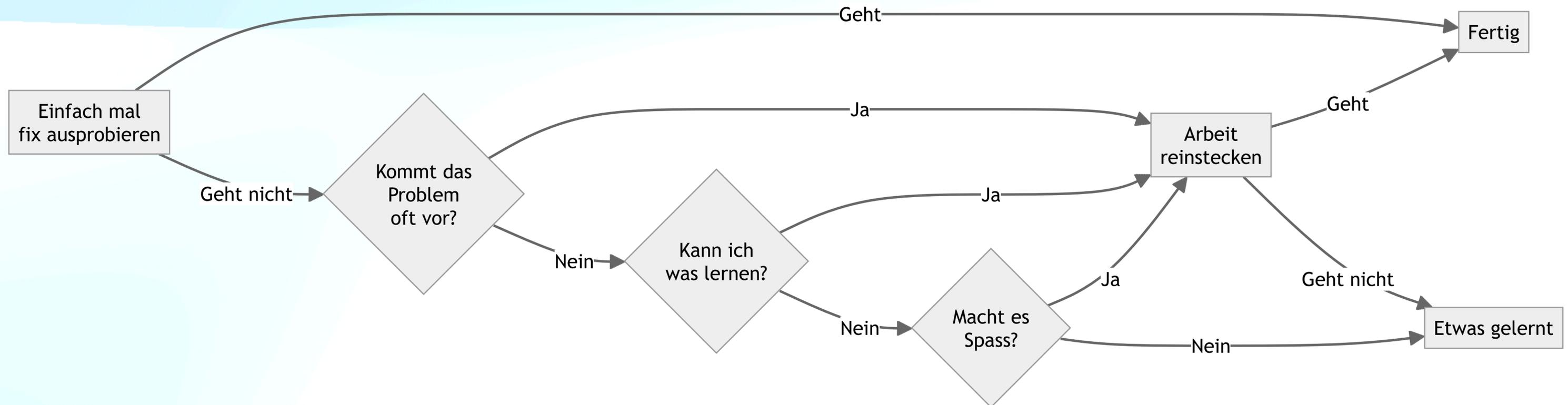
- für Zukunftssicherheit nötig



Wann KI einsetzen?

Intuition und Erfahrung entwickeln

Immer wieder Frage stellen: wo kann KI unterstützen?
The limit is your imagination.



"Eine KI ist auch nur ein Mensch"

Intuition zur KI Nutzung ist wichtig

- Mentales Modell "Mensch" ist besser als "Computer"
- Immenses Langzeitgedächtnis

ABER (zur Zeit):

- Keine Lernfähigkeit -> immer alle nötigen Informationen mitgeben
- Keine Persönlichkeit -> evtl. beschreiben was und wie gewünscht
- Keine Handlungsplanung -> führen der KI durch Prompts
- ähnlich zu menschlichen Fehlern: Halluzinationen, Prompt-Injection, ...

Kompensation im konkreten Einsatz nötig



KI in der IDE

Generat about

De intelcion

Conier niation

Commish

Girnal

Compsile

Críteie

Experimialing

Generi About I I

IDE mtegration

Qral

Amplones

Applicaion of

Chat ggpt

Chaits

Ergeriniels

Enpruue

IDE Plugins

Github Copilot u.a.

- Github Copilot o.ä. in der IDE ein Muss!
Diverse Alternativen: Tabnine, Jetbrains AI Assistant, ...
- Starke Unterstützung der "Coding" Phase beim Entwickeln
- "Do what I think": schlägt oft vor, was ich gerade tippen wollte

... Demonstration ...

Github Copilot

Demonstration Zusammenfassung

- gut beim Aufnehmen von Mustern
- Kommentare zuerst schreiben!
- Refactoring: Cursor direkt vor zu refactorenden Code setzen und mit Refactoring starten -> oft passende Vorschläge
- für unerfahrene und für erfahrende Entwickler extrem hilfreich
- Chat: nutzbar für Refactoring, Tests erzeugen, Fragen beantworten, ...
- ggf. keine Übersicht über Codebasis

Github Copilot / IDE Plugins

Mögliche Bedenken

- Datensicherheit:
 - Nutzung vertrauenswürdiger Firmen
 - Business Services erfüllen viele Anforderungen, zunehmend weitere Alternativen, evtl. local Hosting
- Urheberrechtsrisiko: sollte bei Generieren von angepassten kleinen Codeschnipseln eigentlich gering sein
- Code-Bloat: bekanntes Problem -> Lösung durch Disziplin
- Verwendung von unverstandenen Code-Vorschlägen -> Disziplin

Die vielen Nutzen von ChatGPT

- Alternativen verfügbar: Claude, Gemini, ..., (lokale Modelle)
- Fragen beantworten
- Diskussionspartner (modernes "Rubber Duck Debugging")
- für Coding, als Script Factory, Konzepte
- Triggern von Aktionen möglich (GPTs, tools)

ChatGPT als Scriptfactory

Kleinere Tools schreiben wird oft trivial einfach

Prompt-Fragment Beispiel:

Erzeuge ein bash script für MacOS mit arm64 Architektur.

Es kann MacOS Kommandozeilentools oder mit Package Manager homebrew installierbare Kommandos benutzen. Nutze getopt. Kommentiere den Script.

Nach Unix Konventionen soll es wenn ohne Argumente oder mit --help aufgerufen wird eine Kurzbeschreibung incl. Kommandozeilenargumente ausgeben.

+ Kurzbeschreibung vorgeben

Nacharbeit u.U. mit Github Copilot oder auch im Chat

www.stoerr.net/blog/scriptfactory

ChatGPT für Fachkonzepte

- Feature Spezifikationen von Composum AI stark ChatGPT unterstützt (Beispiel Composum AI [/featurespecs/\[0-5\]*.md](#))
- Schrittweiser Prozess:
 - Manuell geschrieben und schrittweise ergänzt: Grundidee, Entscheidungen, out of scope
 - Weitere Abschnitte erst automatisch auf Anforderung generiert, und dann nachbearbeitet
- Fühlt sich an wie "rapid prototyping" der Spezifikation
- www.stoerr.net/blog/writingSpecs

Prompt-Ideen für Fachkonzepte

(für Dialog, erzeugen jeweils eigenen Abschnitt der nacheditiert wird)

- 20 zusätzliche Dinge, die das Feature erweitern -> Scope abstecken
- typische Nutzer-Workflows ausgeben
- Liste von benötigten Dialogelementen
- Vorschlag für Dialoglayout
- Rendering Dialoglayout : Ascii-Art, SVG, HTML
- Interaktionsdiagramm, ...
- Testcase-Vorschläge

Prompt-Beispiel

Erstelle eine ASCII-Art des Dialogs, gerendert als Markdown-Codeblock mit 4 Leerzeichen Einrückung.

Buttons sollten mit ihrem Text wie [Abbrechen] gerendert werden.

Drop-Down-Listen können wie [V Vordefiniert] gerendert werden.

Textfelder, Textbereiche sollten mit einer Beschreibung des Inhalts gezeigt werden, und ihre Fläche mit _ ausgefüllt werden, um die Fläche, die sie einnehmen, zu zeigen. (Für Textbereiche wird das mehrere Zeilen sein.)

Die Namen von Gruppen und Untergruppen sollten nicht gezeigt werden, außer wenn sie im vollständig implementierten Dialog erscheinen sollen.



LLM als Tool

Einsatz von LLM für eigene Tools

Chat Completion API - einfach aber mächtig

"Liefere eine zu dieser Unterhaltung passende Antwort"
- nicht auf reale Unterhaltung festgelegt!

- Integration Hintergrundinformationen, Beispiele für erwartete Antworten, Instruktionen, Daten

- Pattern zur Trennung Daten / Instruktionen:
"die Daten der KI in den Mund legen"
-> saubere Trennung, stabiler gegen prompt injection

Nutzer: Hallo!
Modell: Hallo! Wie kann ich ihnen helfen?
Nutzer: Schreibe ein Gedicht über die Compositum AI!
Modell: ?

Nutzer: Gib den Seitentext aus
Modell: {Seitentext}
Nutzer: {Nutzerinstruktionen}
Modell: ?

LLM auf der Kommandozeile

- Github Copilot für Kommandozeile: erstellt Kommandos aus Beschreibung
- Tools zur Nutzung der OpenAI API, lokaler Modelle o.ä.
 - LLM <https://github.com/simonw/llm> incl. vieler Plugins, Nutzung lokaler und anderer Modelle, ...
 - chatgpt <https://github.com/stoerr/chatGPTtools>
- Ergänzung der Kommandozeile durch Aufgaben wie Übersetzung, Formatänderungen, Auszüge, ..., die sich leicht verbal beschreiben lassen

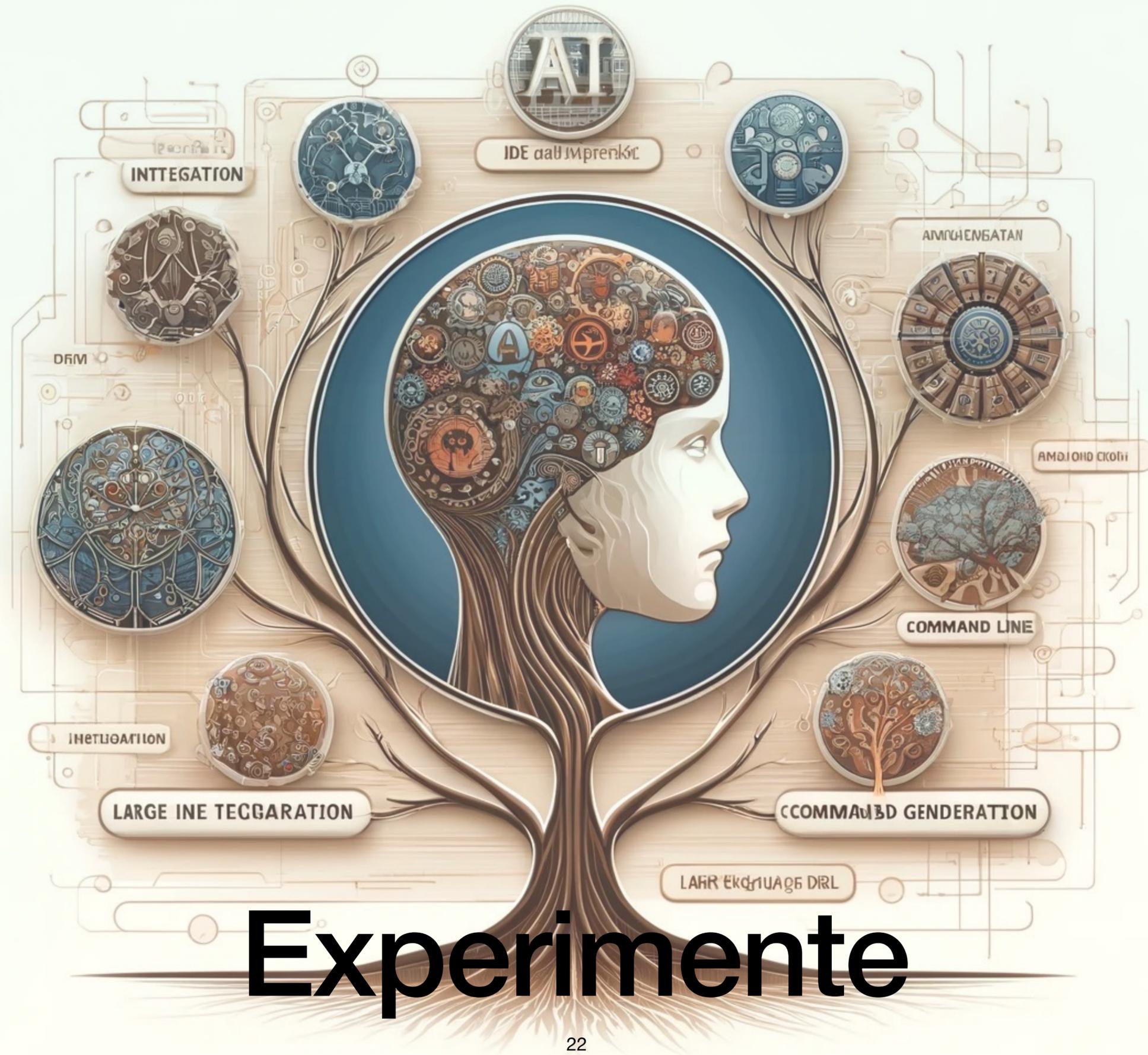
Umgang mit automatisch KI-generierten Files

- Vielseitig: DSL, Code von Dokumentation oder umgekehrt, ...
- Generieren dauert etwas -> nur bei Bedarf anstossen
- Bei komplexeren Manipulationen Nachkontrolle nötig (manuell oder durch gute Tests)
 - Idee: Einchecken in Git -> automatisch manuelle Kontrolle bei commit
- aigenpipeline.stoerr.net :

```
aigenpipeline -p prompt.txt -o output.txt input.txt ...
```

Generieren nur bei Änderungen durch Einbetten Versionskommentar z.B.

```
// AIGenVersion(4c07e3b5, prompt.txt-1.5, input.txt-7e10dd2a)
```

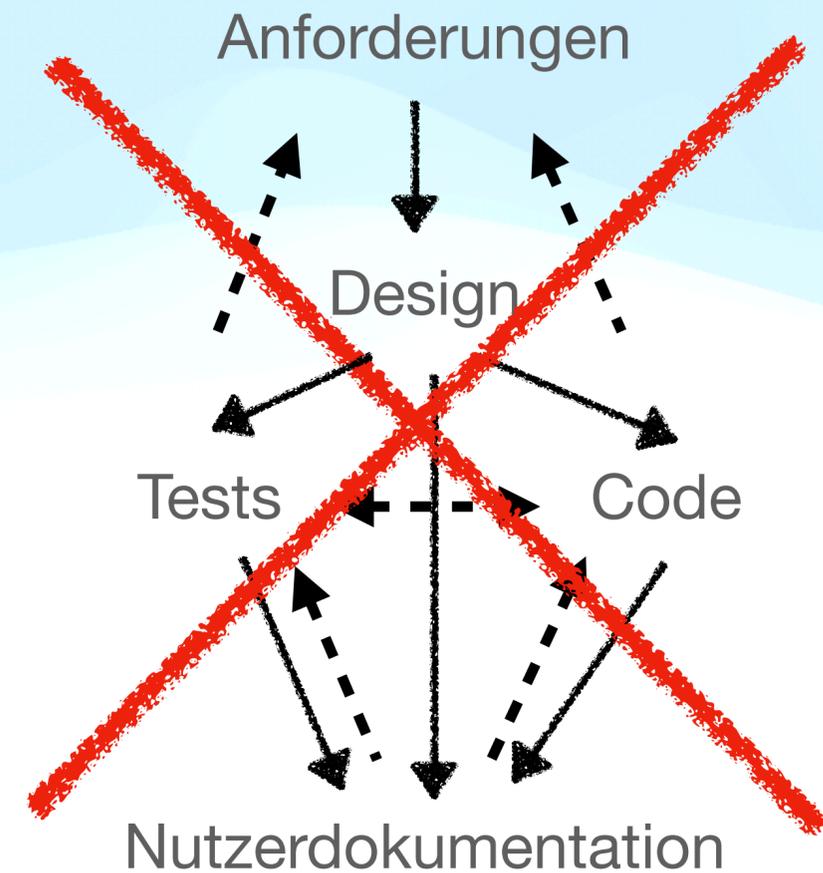


Experimente

Generieren von ganzen Programmen?

Grenzen der AI

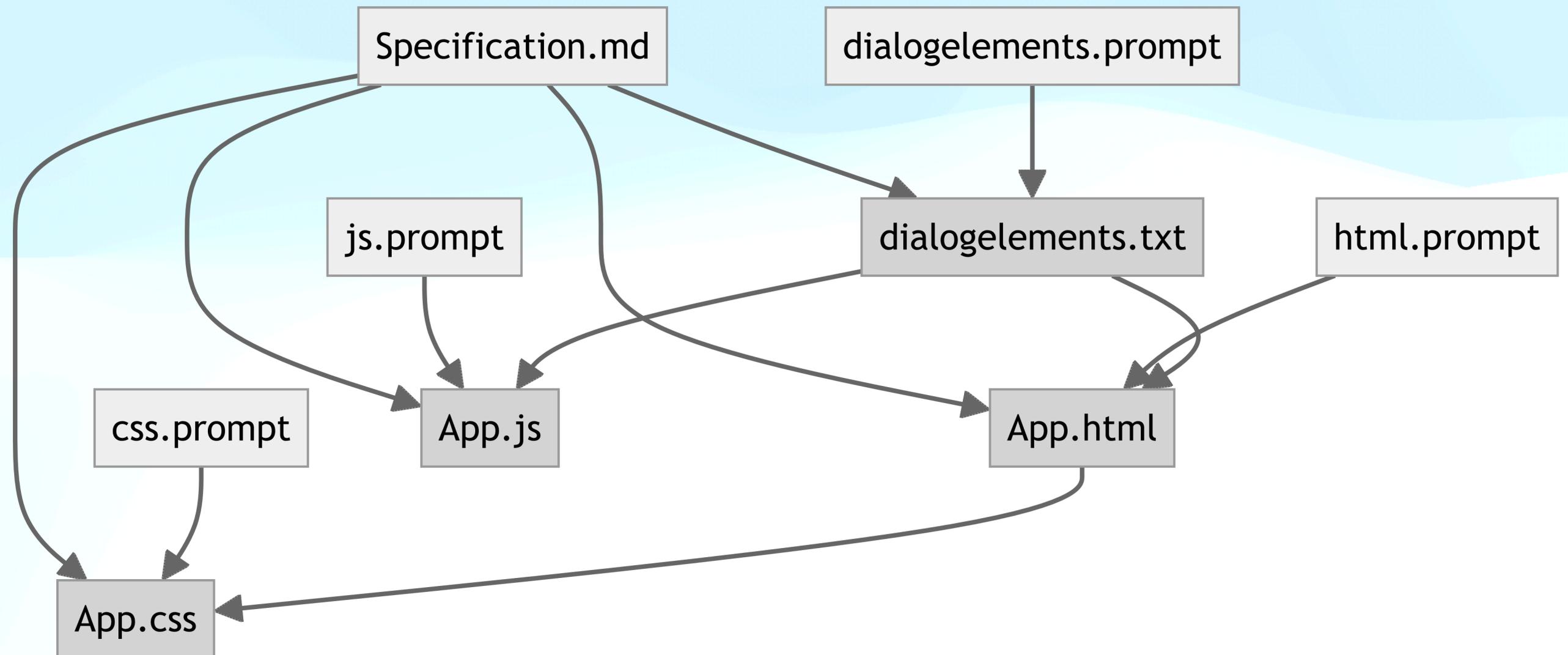
- Für einfache Fälle schrittweises Generieren von Dokumenten / Code je nach Komplexität möglich
- Tatsächlicher Ablauf ist aber nichtlinear und mit Iterationen und viel Kommunikation verbunden -> ggf. schwer möglich.
- Besser: Unterstützung bei Einzelaufgaben?



Generieren von Programmen: was geht

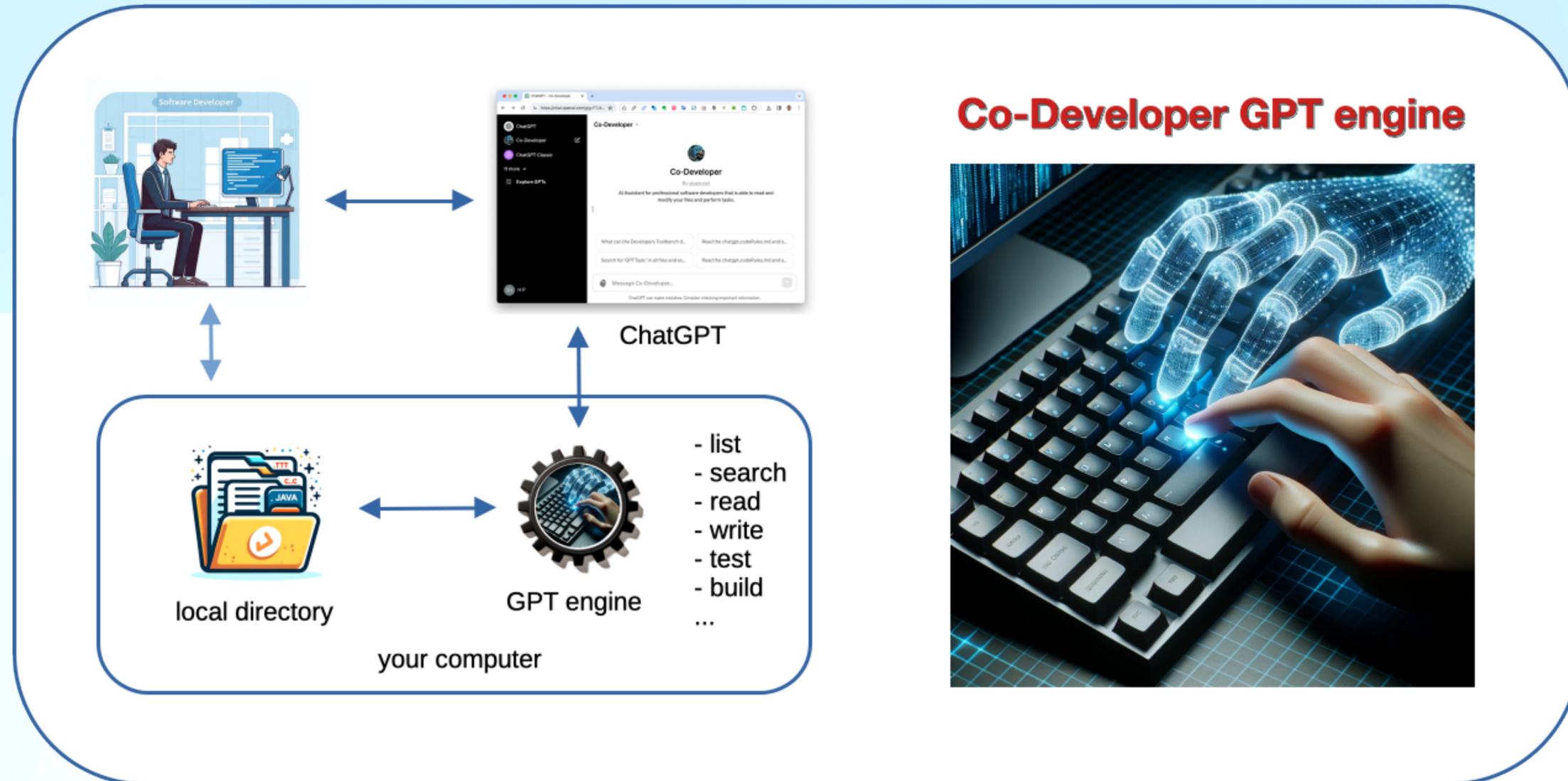
Für kleine Prototypen: Generieren aus Spezifikation

(Beispiel mit
eigenpipeline)



"Spielwiese" - ChatGPT als Entwickler

Chat mit autonomen Zugriff auf lokales Verzeichnis



Co-Developer GPT engine



Entwicklungsunterstützung CoDeveloper GPT engine

codevelopergptengine.stoerr.net

- Start in beliebigem Verzeichnis
- Zugriff durch ChatGPT innerhalb eines Chats auf dieses Verzeichnis
 - suchen, lesen, ändern von Dateien
 - Webzugriff (URL -> Markdown für ChatGPT)
 - Ausführen konfigurierter Aktionen (z.B. Build, Testlauf)
- Beispiele: Git Commits in engine source, die mit "ChatGPT: " anfangen
- OpenAI ChatGPT Plus nötig

Vielen Dank!

<http://www.stoerr.net/ai> Links, Blog

Austausch zum Thema gesucht!

www.composum.com / ai.composum.com

Viel Spaß!