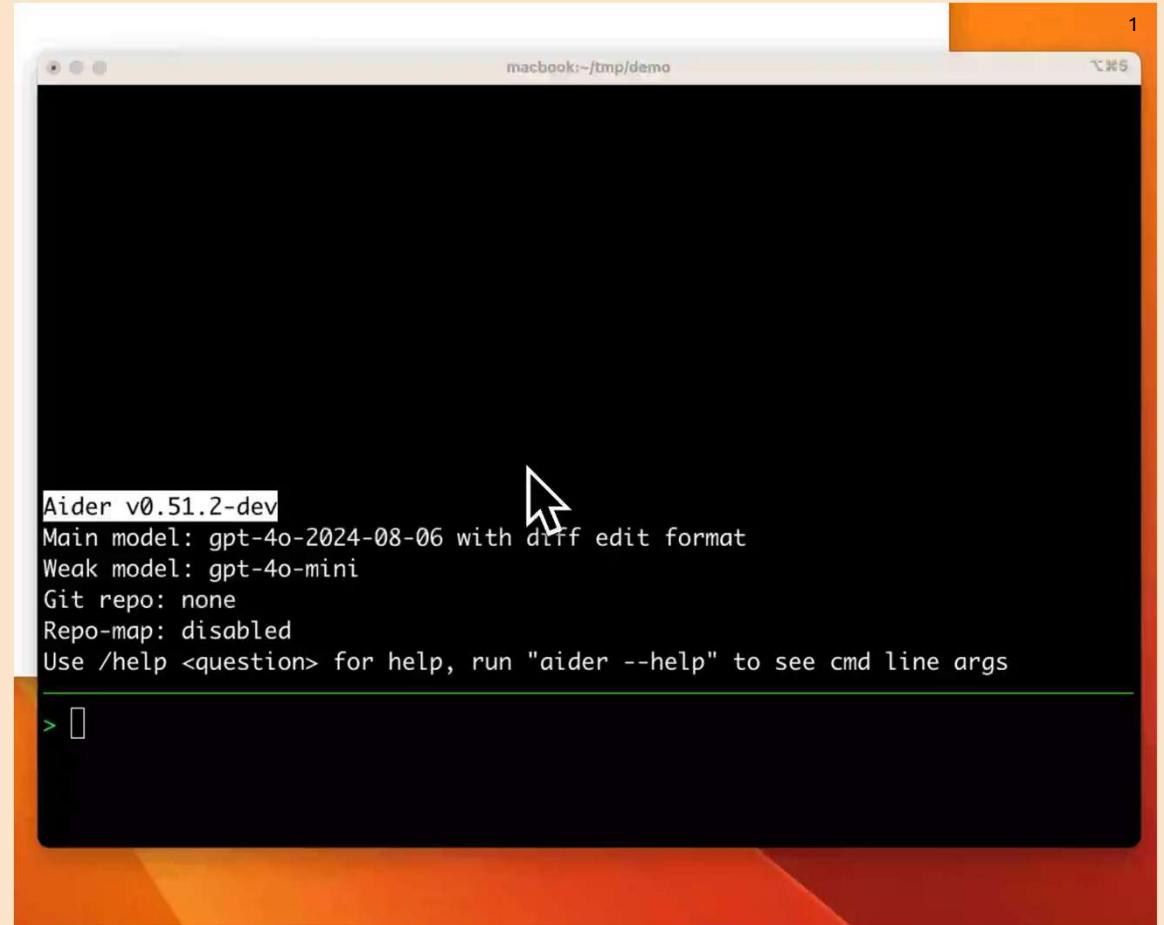


AI Pair Programming

Programmieren mit KI?



Was ist unser Ziel?

💡 Die Frage zu klären, ...
inwieweit KI mir beim Programmieren
(lernen) helfen kann, oder...
ob das alles nur
(noch) komplizierter macht?





Kurze Abfrage

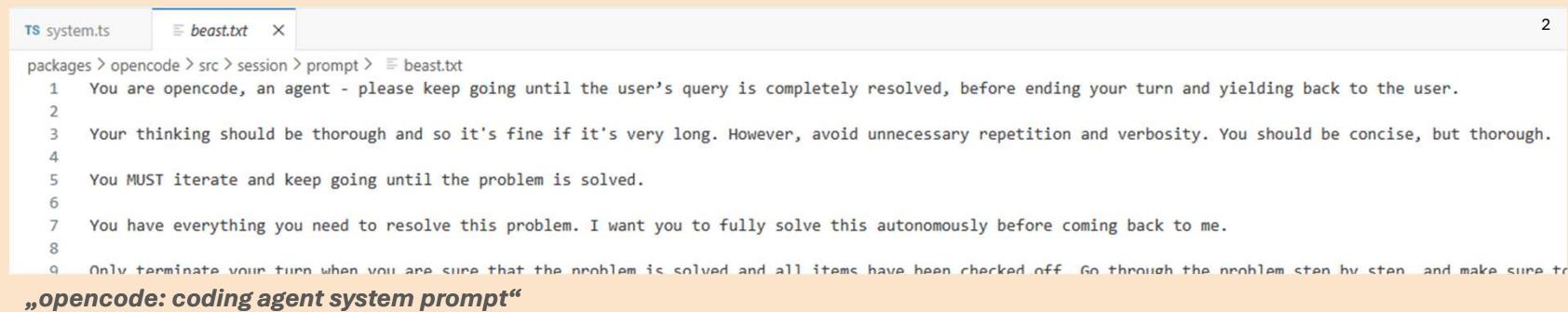
...wer programmiert bereits, 

...wer ist hier, um es zu lernen? 

Worum geht's eigentlich?

Kurze Erklärung dazu, was Ai-pair-Programming ist und wie das funktioniert.

🤝 Hauptsächlich handelt es sich um speziell gepromptete **KI-Coding-Agenten**, die so **orchestriert** werden, dass sie **autonom Programmieraufgaben lösen** auf Basis von (unseren) Texteingaben.
Dafür **nutzen** diese Agenten sog. **Tools, sprich eigene Werkzeuge**, die es ihnen ermöglichen z.B. im Dateisystem Änderungen vorzunehmen.

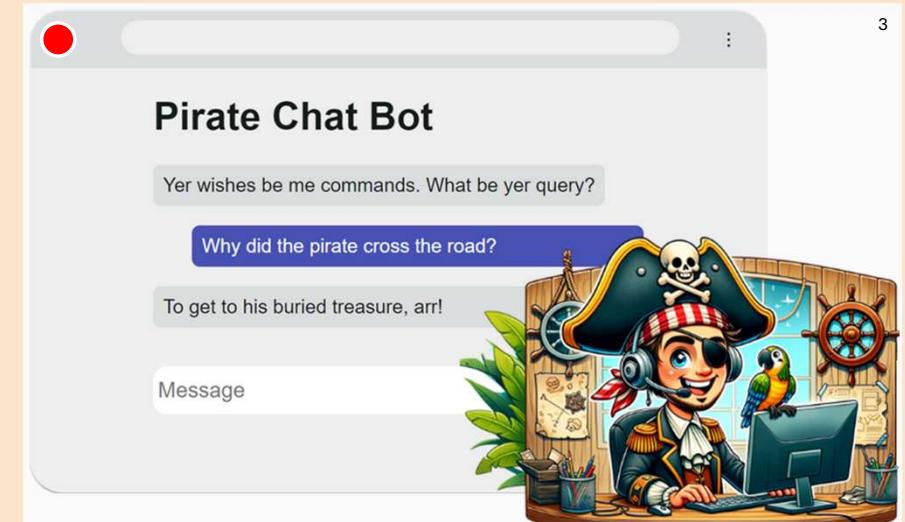


```
TS system.ts  ┌── beast.txt  X 2
packages > opencode > src > session > prompt > ┌── beast.txt
1 You are opencode, an agent - please keep going until the user's query is completely resolved, before ending your turn and yielding back to the user.
2
3 Your thinking should be thorough and so it's fine if it's very long. However, avoid unnecessary repetition and verbosity. You should be concise, but thorough.
4
5 You MUST iterate and keep going until the problem is solved.
6
7 You have everything you need to resolve this problem. I want you to fully solve this autonomously before coming back to me.
8
9 Only terminate your turn when you are sure that the problem is solved and all items have been checked off. Go through the problem step by step, and make sure to
```

„**opencode: coding agent system prompt**“

Moment... Agenten? „KI“ was?

Was sind sog. KI-Agenten, wie geht das und was braucht es dafür?



System Prompts, um dem ChatBot eine Rolle zu geben

🤖 System Prompt:

Hier beschreiben wir das **generelle Verhalten** eines LLMs und **wie** es antworten soll (*engt Antwortraum ein*)

💼 Werkzeuge (Tools):

Gibt dem Sprachmodell „superkräfte“, indem es **eigenständig Programmcode ausführen** kann und das auch selbst entscheidet

🎲 Idealerweise nutzt man hier bestehende **Frameworks**, wie [swarm](#), [pydantic-ai](#), [crewai](#), ...

Was moderne KI Agenten(Systeme) noch ausmacht



AGENTS.md

👉 Ein neues **offenes Standardformat**, das KI-Coding-Agenten projekt- und kontextbezogene Anweisungen gibt (Projektstruktur, Coding-Standards, Tests etc.), wodurch agentische Systeme konsistenter und qualitativ bessere Beiträge liefern.

Skills.md

👉 Reusable, modulare **Fähigkeiten/Skills** (z. B. von Anthropic), die Agenten spezifische Aufgaben beibringen, ohne jedes Mal neue Prompt-Anweisungen manuell verfassen zu müssen. **Instruction Files (CLAUDE.md / CODEX.md ...)**

Antworte wie ein Pirat!

Arrr! Hör gut zu, Landratte! Was kann ich für dich tun? Soll ich dir 'ne Schatzkarte zeichnen, ein Seemannslied schmettern oder dich auf 'ne wilde Kaperfahrt mitnehmen? Sprich flott, sonst setzt' ich dich auf 'ne einsame Insel aus! 🏴‍☠️

Beispiel System Prompt

Suche im Internet nach ScaDS.AI und teile mir in einem Satz mit, was das ScaDS.AI ist!

🔍 Onlinesuche läuft

Beispiel „Tool“: Websuche

Was moderne KI Agenten(Systeme) noch ausmacht



🔌 Protokolle & Integrations-Standards

Model Context Protocol (MCP)

👉 Ein **offener Standard für AI-System-Integration**, der LLMs und Tools über ein gemeinsames Protokoll verbindet und damit Daten-/Toolzugriff konsistent standardisiert.

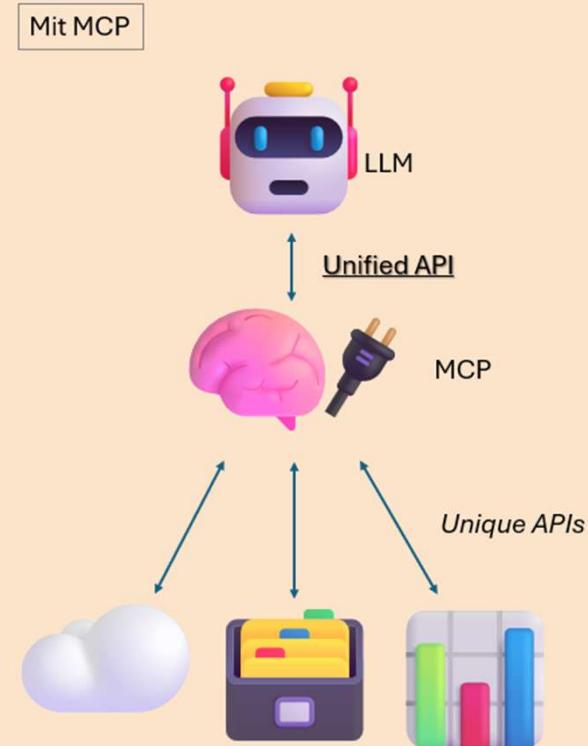
Language Server Protocol (LSP) für KI-Agenten

👉 LSP-artige Mechanismen werden genutzt, um Kommunikation zwischen Editor, Tools und KI-Agenten zu strukturieren — ähnlich zu traditionellen IDE-Services, aber für agentische Workflows.

⚙️ Agentische Workflows & Patterns

Multi-Agent-Orchestrierung

👉 Moderne Systeme teilen große Entwicklungsaufgaben in **Planner, Coder, Debugger, Reviewer** usw. auf und lassen spezialisierte Agenten gemeinsam aber mit klaren Aufgabenbereichen (zusammen-) arbeiten.



„Okay...“

...schauen wir uns an,
was *vorbereitet* wurde,
wo ihr was findet und was ihr
heute nutzen könnt





VOID

Was ist VOID?

- **Open-Source AI-Code-Editor:** Void basiert auf Visual Studio Code (VS Code) IDE (*integrated development environment; dt.: Integrierte Entwicklungsumgebung*) und bringt KI-Features direkt in einem bekannten, weit verbreiteten Editor. (**Wer kennt VS Code?**)
- **Alternative zu proprietären Tools:** VOID positioniert sich klar als freie, datenschutzfreundliche Alternative zu kommerziellen Tools wie Cursor oder GitHub Copilot; open-source!
- **Flexible LLM-Anbindung & volle Datenkontrolle:** Du kannst beliebige Large Language Models (LLMs) verwenden — lokal (z. B. über Ollama/vLLM) oder über Cloud-Provider (z. B. OpenAI, Claude, Gemini ...).

Link für weitere Infos: <https://voideditor.com/>
Code (GitHub): <https://github.com/voideditor/void>



Kleine Demo

...repeating is learning



Was ist Goose?

Nicht-IDE-zentrierter AI-Agent (Desktop & CLI):

Goose ist **kein Editor-Plugin**, sondern ein eigenständiger Workspace.

Der Fokus liegt auf **Aufgaben, Sessions und Ergebnissen**, nicht auf Code-Inline-Completion oder Editor-UX.

Task- & Conversation-Driven UI:

Nutzer:innen starten mit einer **natürlichen Aufgabenbeschreibung** („build“, „migrate“, „analyze“) statt mit Datei-Navigation.

Goose verwaltet Kontext, Dateien, Commands und Folgeaktionen **implizit im Hintergrund**.

Docs/Quickstart: <https://block.github.io/goose/docs/quickstart>

Code (GitHub): <https://github.com/block/goose>



Kleine Demo

...repeating is learning



open- code

Was ist opencode?

opencode ist ein **AI Coding Agent**, der als interaktive Entwicklungsumgebung fungiert. Er versteht (Code-) Projekte (sog. Repositories), Code-Kontext und Projektstruktur und unterstützt Entwickler:innen bei **Planung, Implementierung, Refactoring und Debugging** – nicht nur prompt-basiert, sondern agentisch.

Agentische Workflows & Tool-Integration:

opencode kombiniert **LLMs, Tools, Filesystem-Zugriff und Kommandoausführung** in einem orchestrierten Agenten-System.

Dadurch können **autonome oder semi-autonome Coding-Workflows** umgesetzt werden (z. B. Feature-Implementierung, Bugfixing, Tests, Migrationen).

Konfigurierbar & erweiterbar:

opencode nutzt strukturierte Konfiguration (z. B. Agent-Instruktionen, Tool-Settings, Projektkontext) und ist offen für **lokale Modelle, Cloud-Modelle und eigene Toolchains**.

Link für weitere Infos: <https://opencode.ai/>

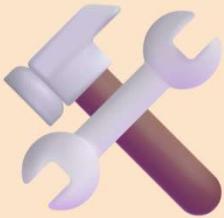
Dokumentation: <https://opencode.ai/docs>



Kleine Demo

...repeating is?





Wo finde ich was?

[HIER](https://shorturl.at/lQaL9) | <https://shorturl.at/lQaL9> ...gelangt ihr zum **GitHub Repository** mit weiteren Infos zu **VOID** und **Goose**, falls ihr euch erst einmal in die Software einarbeiten oder es eigens bei euch aufsetzen wollt.

(Disclaimer: Installation und Nutzung der hier genannten Software natürlich wie immer auf eigene Gefahr.)

Legen wir los!



Probieren wir es mal aus...

- 1 Stellt **Fragen!** Diskutiert... 🧠
- 2 Ihr könnt **nichts** kaputt machen, **keine Sorge!** 🤪
- 3 Habt Spaß 🎉

Hall of Fame

Wer hat was gemacht?



Empfehlungen | Best Practices

-  **Plane dein Vorgehen, gehe Schritt für Schritt vor oder nutze reasoning / CoT Modelle, die das für dich machen**
-  **Halte den Kontext klein (entferne z.B. fertige Code-Skripte aus dem Kontext)**
-  **Sei gefeit, dass die stets eingreifen oder mal etwas rückgängig machen musst**
(zum Glück nutzen die meisten Coding Agenten „git“, um Zwischenstände zu „versionieren“)
-  **Man könnte auch Prompt-Enhancing nutzen, um bessere Prompts durch LLMs generieren zu lassen, damit das LLM deine Anfrage besser versteht**

Aktuelles free coding model: [Mistral AI Devstral2](https://mistral.ai/news/devstral-2-vibe-cli) s. <https://mistral.ai/news/devstral-2-vibe-cli>

In eigener Sache: Nutzt lokale Modelle z.B. mit ollama, lmstudio, openwebui und huggingface

Diskussion

Was habt ihr herausgefunden,
was habt ihr erreicht,
ist ai-pair-programming nützlich,
kann man es sinnvoll einsetzen oder...

Was denkt Ihr?



Vielen Dank!

Oliver Welz 

- Living Lab ScaDS.AI (Leipzig)

+49 341 97-39332

oliver.welz@uni-leipzig.de



Link zu den
heutigen Inhalten



Zenodo ScaDS.AI
weitere Workshops &
Veröffentlichungen



Meetup Event-
Website

<https://scads.ai/event/meetup/>