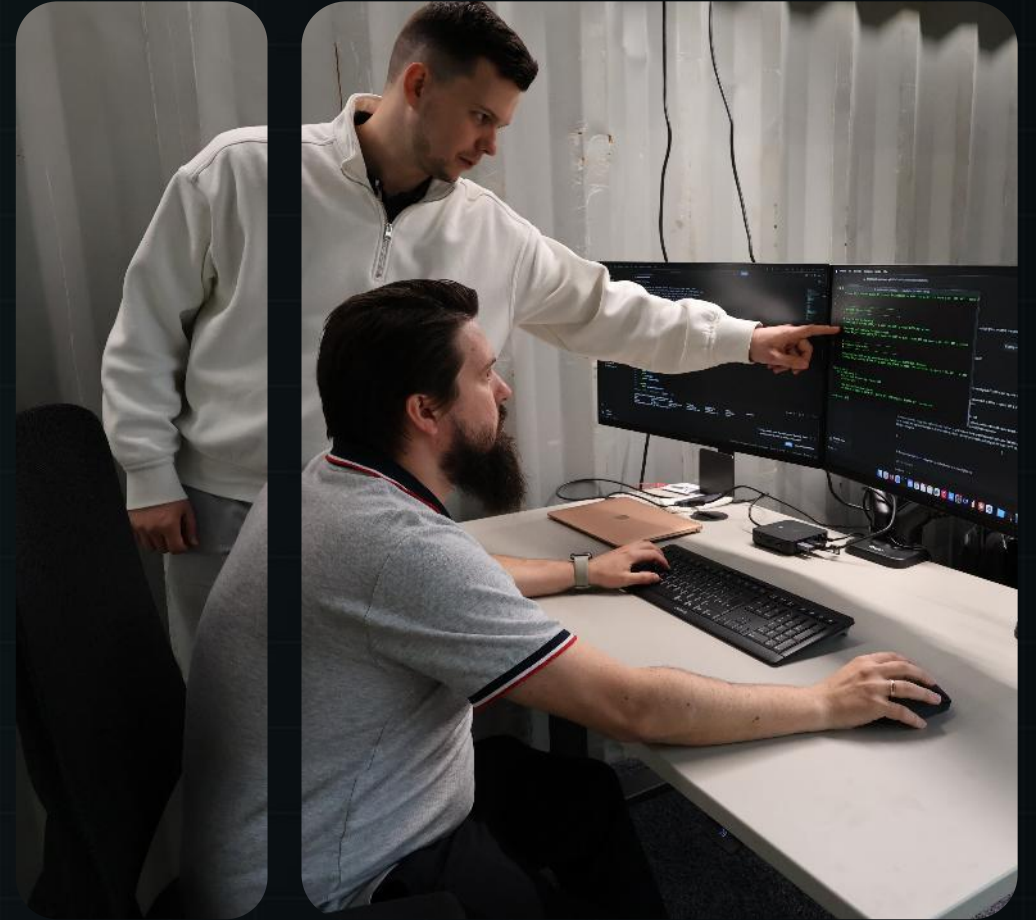


KI & Agentic AI im Unternehmen

Technische Grundlagen von KI

Einordnung · Skills & Tools · KI-Agenten

zelos.



Was ist eigentlich alles Künstliche Intelligenz?



Sprachmodelle (LLMs)



Deep Learning / Neuronale Netze

Maschinelles Lernen

Künstliche Intelligenz

Sprachmodelle (LLMs)

Auf Sprache spezialisiert; Basis heutiger KI-Assistenten. Emergente Intelligenz.

Beispiele: Transformer z.B. GPT, Claude, BERT

Deep Learning / Neuronale Netze

Vielschichtige Netze erkennen komplexe Muster in Daten.

Beispiele: CNN, GAN, Autoencoder, Diffusion Models

Maschinelles Lernen (ML)

Systeme lernen aus Daten; statt fester, programmierter Regeln.

Beispiele: Decision Tree, Random Forest, Support Vector Machine, k-Nearest Neighbors, lineare & logistische Regression

Künstliche Intelligenz

Overbegriff: Maschinen lösen Aufgaben, die sonst „Intelligenz“ erfordern.

Beispiele: Expertensysteme, Suchalgorithmen A*/Minimax z.B. Deep Blue

Wenn heutzutage von „KI“ die Rede ist, sind oft Sprachmodelle (LLMs) gemeint.

Cloud versus Self-Hosted KI



Cloud Provider

Wo das Modell läuft?	Rechenzentrum des Anbieters
Wo die Daten hingehen?	Verlassen das eigene Netz
Wofür man zahlt?	Verbrauchte Tokens
Wer betreibt es?	Der Anbieter
Modellauswahl?	Was angeboten wird, provider kann jederzeit ändern
Skalierung?	Sofort, elastisch



Self Hosted

Wo das Modell läuft	Eigene GPU, Server oder gemietete Instanz
Wo die Daten hingehen	Bleiben im Haus
Wofür man zahlt	Hardware + Strom, unabhängig von der Nutzung
Wer betreibt es	Man selbst
Modellauswahl	Jedes Open-Weight-Modell, eingefroren auf der eigenen Version
Skalierung	Nächste GPU kaufen



Entscheidung?: Hängt stark von den Anforderungen des Use Cases ab. Cloud ist für die meisten Fälle die einfachere und günstigere Lösung, aber kein Datenschutz und volle Abhängigkeit von US-Firmen auch in ggf. kritischen Prozessen. Eigenes Hosting ist ggf. Teuer und Wissensintensiv.

Cloud versus Self-Hosted KI

Model	VRAM (Q4)	Hardware	Buy (GPUs only)
Kimi K3	~1.5-1.7 TB	12-16× H200	\$400-650k
DeepSeek V4 Pro	~900 GB	8× H200	~\$315k
GLM-5.2	~420 GB	6× H100	\$150-240k
Qwen3-Coder-480B	~265 GB	4× H100	\$100-160k
MiniMax M1	~250 GB	2× H200	~\$62k
DeepSeek V4 Flash	~158 GB	2× H200	~\$62k
Llama 4 Scout	~60 GB	1× H200 / M4 Max 128 GB	~\$31k / ~\$5k
Qwen3-Coder-Next	~46 GB	2× RTX 4090	\$3-4.5k
Qwen3.6-35B-A3B	~22 GB	1× RTX 4090 / 5090	\$1.5-4.3k



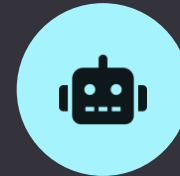
Provider	Models	Price \$ per M in/out
Anthropic	Claude Opus 5, Fable 5 etc.	Frontier~\$5/\$25
OpenAI	GPT-5.6 family, GPT-5.5 etc.	Frontier ~\$5/30
Google	Gemini 3.x Pro, Flash etc.	Pro ~\$2/\$12

Eigenschaften von KI Agenten



Software Agent

- Autonom, Sozial, Reaktiv, Proaktiv (Wooldridge & Jennings, 1995) ^[1]
- Dreiklang Autonomen Systeme: Perzeption, Kognition und Aktion (Russel & Norvig, 2021) ^[2]
- Spätere Ergänzung: Selbst lernendes System



Sprachmodell Agenten

- Large Language Models (LLMs) als Reasoning & Decision Engine
- Planning & Action Loop
- Tool/Function Calling
- Memory/Knowledge
- Heartbeat

LLMs eignen sich als „Gehirn“ für Autonome Agenten

[1] Jennings, Nicholas R., and Michael Wooldridge. "Applying agent technology." Applied Artificial Intelligence an International Journal 9.4 (1995): 357-369.

[2] Russell, Stuart, and Peter Norvig. "Artificial Intelligence: a modern approach, 4th US ed." aima: сайт. URL: <https://aima.cs.berkeley.edu/> (2021).



Agent Harness Frameworks

Etwas schnell zum laufen bringen



Claude Code



Codex

- Gute allround Performance
- Zuverlässig
- Weniger Kontrolle/Anpassungen

Etwas neues und spezifisches bauen



Hermes



Pi

- Open Source
- Große Community
- Volle Kontrolle/Anpassungen



Entscheidung?: Hängt auch hier vom Use Case ab. In der Regel mit Claude oder Codex starten und testen. Nur wenn spezifische Anpassungen notwendig sind umsteigen auf Hermes oder Pi.

Coding- versus General Purpose Agents

Werkzeuge: read,
write, bash



Werkzeuge: read, write,
bash, git, web search,
browser usage, office
tools etc.



Werkzeuge: read, write, bash,
git, web search, browser
usage, office tools, cron,
memory, kanäle (Slack,
Whatsapp, Telegram ..) etc.



Eine Domänen & Wenige Werkzeuge

Automatische Feedback (Compiler, Lintner,
Statische Analyse, Tests)

Viele Domänen & Viele
Werkzeuge

Feedback durch den Nutzer



Takeaway?: Coding-Agenten funktionieren heute deutlich besser als General-Purpose-Agenten. Automatisches Feedback aus Compiler, Tests und Statischer Analyse ist der Grund, warum Agenten zuerst in der Softwareentwicklung angekommen sind. Für alles andere fehlt der Compiler. Deshalb ist Verifikation das eigentliche Problem der nächsten Jahre.



```
Eingabeaufforderung
Microsoft Windows [Version 10.0.26200.8875]
(c) Microsoft Corporation. Alle Rechte vorbehalten.
C:\Users\Robin>

# 1. Installieren
npm install -g --ignore-scripts @earendil-works/pi-coding-agent

# 2. Authentifizieren
export ANTHROPIC_API_KEY=sk-ant-...
# oder im Tool: /login (Claude Pro/Max, ChatGPT, Copilot)

# 3. Starten
cd mein-projekt
pi
```

- Minimaler Harness, MIT-Lizenz
- Vier Werkzeuge: Read, Write, Edit, Bash + winziger System-Prompt
- 15+ Provider, hunderte Modelle
- Sessions als Baum: zu jedem Punkt zurück, verzweigen
- Kontextsteuerung: AGENTS.md, SYSTEM.md, anpassbare Compaction, Skills
- Extension API (Pi baut auf nachfrage sich selbst Extension)
- Nicht enthalten: MCP, Sub-Agenten, Plan-Modus, Permission-Popups, To-dos



```
Eingabeaufforderung
Microsoft Windows [Version 10.0.26200.8875]
(c) Microsoft Corporation. Alle Rechte vorbehalten.
C:\Users\Robin>

# 1. Installieren
curl -fsSL https://hermes-agent.nousresearch.com/install.sh | bash

# 2. Setup-Wizard
hermes setup --portal # OAuth: Modell + Web, Bilder, TTS
# Alternativ: hermes setup -> Full Setup (eigene Keys) | Blank Slate

# 3. Starten
hermes
```

- Maximaler Harness, MIT-Lizenz
- Vollausrüstung: Terminal, Web, Vision, Bildgenerierung, TTS, Code-Ausführung etc.
- 20+ Kanäle: Telegram, Slack, WhatsApp, E-Mail, CLI etc.
- Persistentes Gedächtnis: lernt Projekte, generiert Skills selbst, vergisst Lösungswege nicht
- Cron in natürlicher Sprache: Reports, Backups, Briefings laufen unbeaufsichtigt
- Sub-Agenten & Sandboxing über fünf Backends: lokal, Docker, SSH, Singularity, Modal
- 300+ Modelle über Portal



Thank You For Your Attention!

✉ Kontakt@Zelos-Digital.de

☎ +49 7161 62720-40

Zelos Digital Solutions UG
Bosslerstrasse 12
73098 Rechberghausen
Deutschland