

Skill-Shift 2026

Warum Agentic Coding
die neue Geheimwaffe
im Marketing ist



Günther Dietz
otago.at





Generate images from text phrases with VQGAN and CLIP (z + quantize method with augmentations).

Notebook by Katherine Crowson (<https://github.com/crowsonkb>, <https://twitter.com/RiversHaveWings>). The original BigGAN + CLIP method was made by <https://twitter.com/advadnour>. Translated and added explanations, and modifications by Eleiber # 8347, and the friendly interface was made thanks to Abulafia # 3734.

For a detailed tutorial on how to use it, I recommend [visit this article]

(https://yourcraitures.miraeheze.org/wiki/Help:Create_images_with_VQGAN+CLIP), made by Jakeukalane # 2767 and Avengium (Angel) # 3715

Modified by: Justin John

Licensed under the MIT License (*Double-click me to read the license agreement*)

[Code anzeigen](#)

Check GPU type

Factory reset runtime if you don't have the desired GPU.

V100 = Excellent (*Available only for Colab Pro users*)

P100 = Very Good

T4 = Good (*preferred*)

K80 = Meh

P4 = (*Not Recommended*)



Skillshift 2026

1

Meine AI-Reise

Prägende WTF-Momente:
Von Generative-Art zu Autonomen Coding-Agenten

2

Agentic hits different

Paradigmenwechsel: Warum autonomes Handeln
& Iterieren völlig neue Möglichkeiten eröffnet

3

In der Praxis

Echte Use-Cases: Reddit Research, Creative Automation,
Lokale Python Audits

4

Der Skill Shift

Welche Kompetenzen 2026 & darüber hinaus
echten Wert stiften

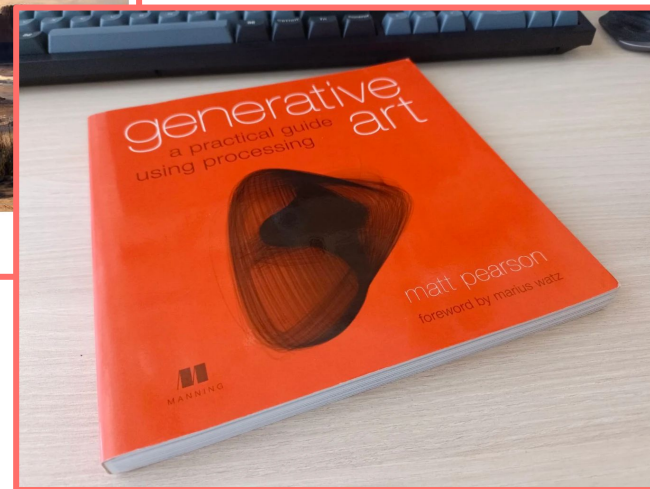
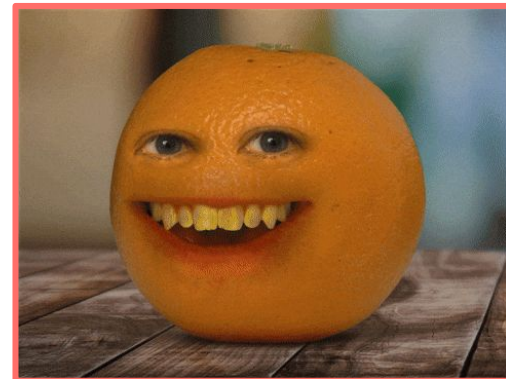
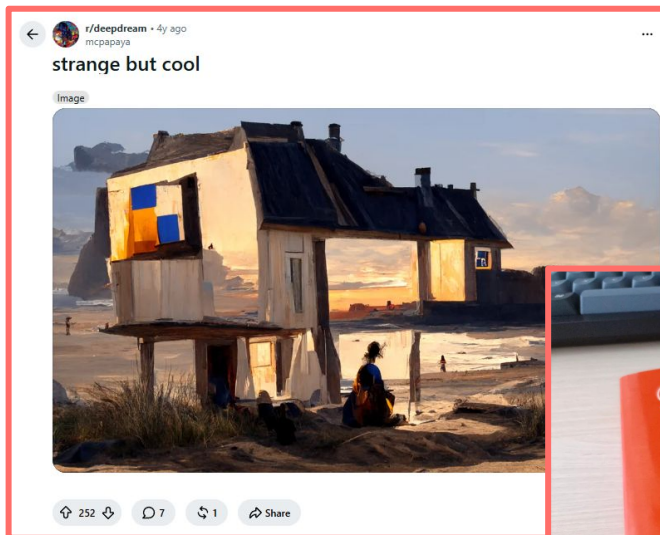
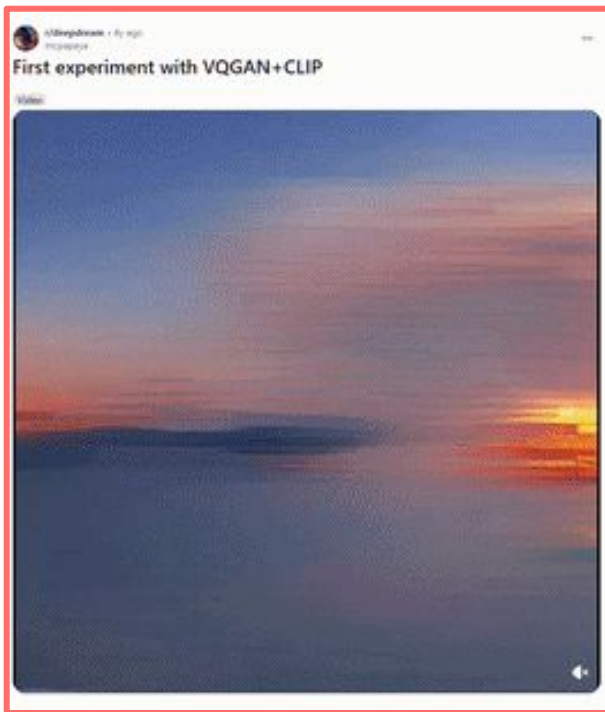
1

Meine persönliche AI Reise



März 2022 – WTF-Moment(s) #1

Maschinen können aus vagen Beschreibungen visuelle Welten schaffen



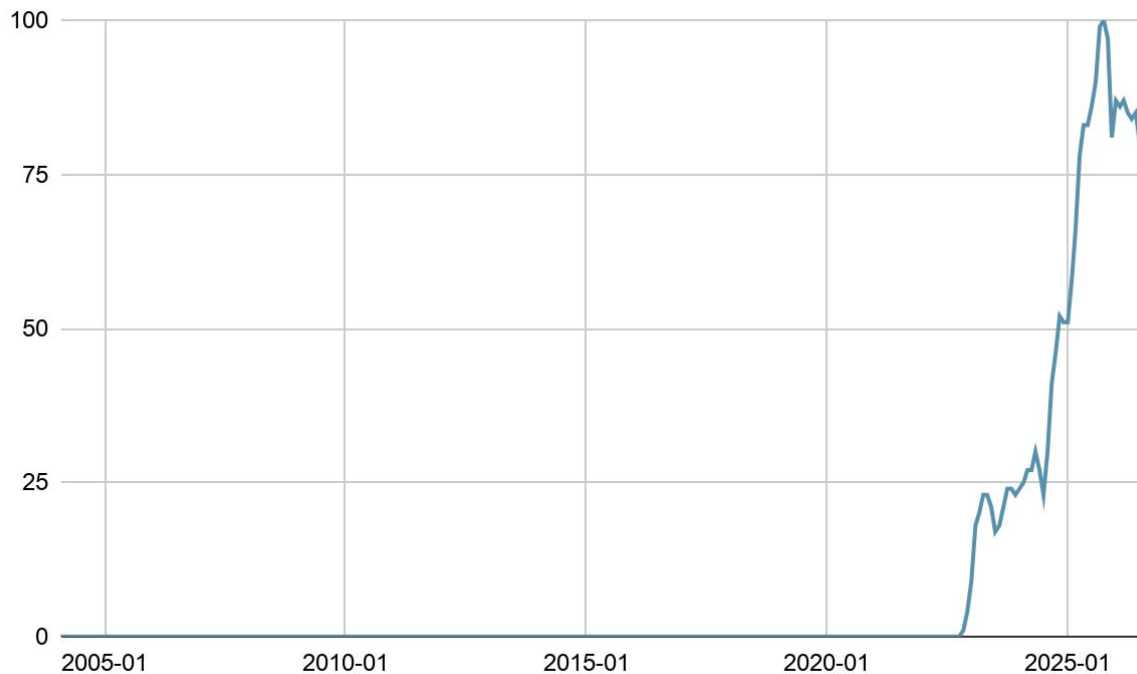
März 2022 – WTF-Moment(s) #1

Maschinen können aus vagen Beschreibungen visuelle Welten schaffen



November 2022 – WTF-Moment #2

Der erste weltweite "Aha-Effekt" – ChatGPT kommt auf den Markt





**A FEW
MOMENTS LATER**



Dezember 2025 – WTF-Moment #3

Der Schritt vom cleveren Chatbot zum produktiven Buddy

**Günther Dietz** • Sie
Business Development & Programmatic Advertising @ otago
5 Monate • Bearbeitet •

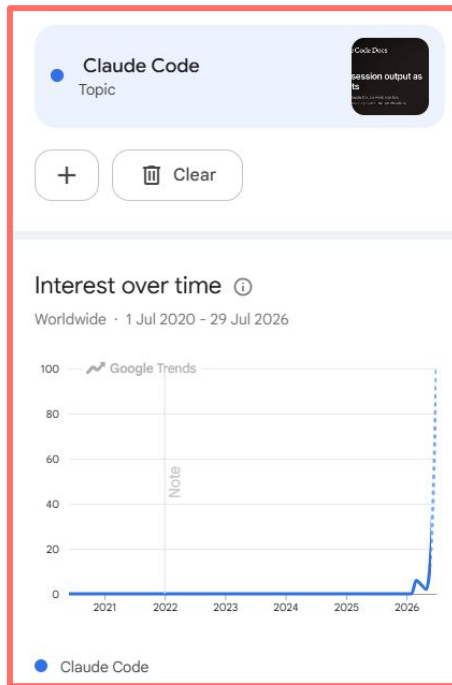
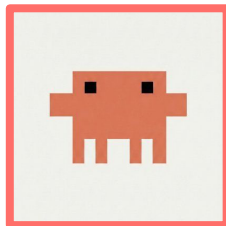
Guess one of the best investments at the moment is a **Claude** Pro/Max account + learning how to use/leverage that monster (mainly Claude Code) and then max out the rate limits until there is no tomorrow.

It's completely wild what this thing and the underlying models are already capable of.

This goes way beyond coding btw.

Thank me later.

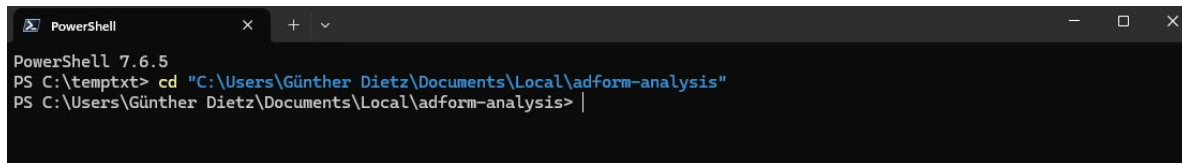
 18



Key Takeaway: Curiosity Wins

Einfach probieren – Agentic Hits Different

- Check dir ein Claude oder OpenAI Paid Abo privat für ein paar Monate, um die Systeme zu lernen
 - ein gutes Invest, ich verspreche es
- Verwende AI um AI zu lernen
- Verwende YouTube/Reddit/etc um AI zu lernen
- Lerne CLI Basics
 - z.B. Odin Project - Command Line Basics
- Teste Claude Code oder Codex (OpenAI) in der CLI
- **Enjoy your WTF Moment :-)**



```
PowerShell 7.6.5
PS C:\temptxt> cd "C:\Users\Günther Dietz\Documents\Local\adform-analysis"
PS C:\Users\Günther Dietz\Documents\Local\adform-analysis> |
```



2

Why “Agentic” hits different



Von Input/Output zu Input/Outcome

Vom Erklären ins Tun kommen

ChatGPT, Gemini & andere Tools im Web Interface:

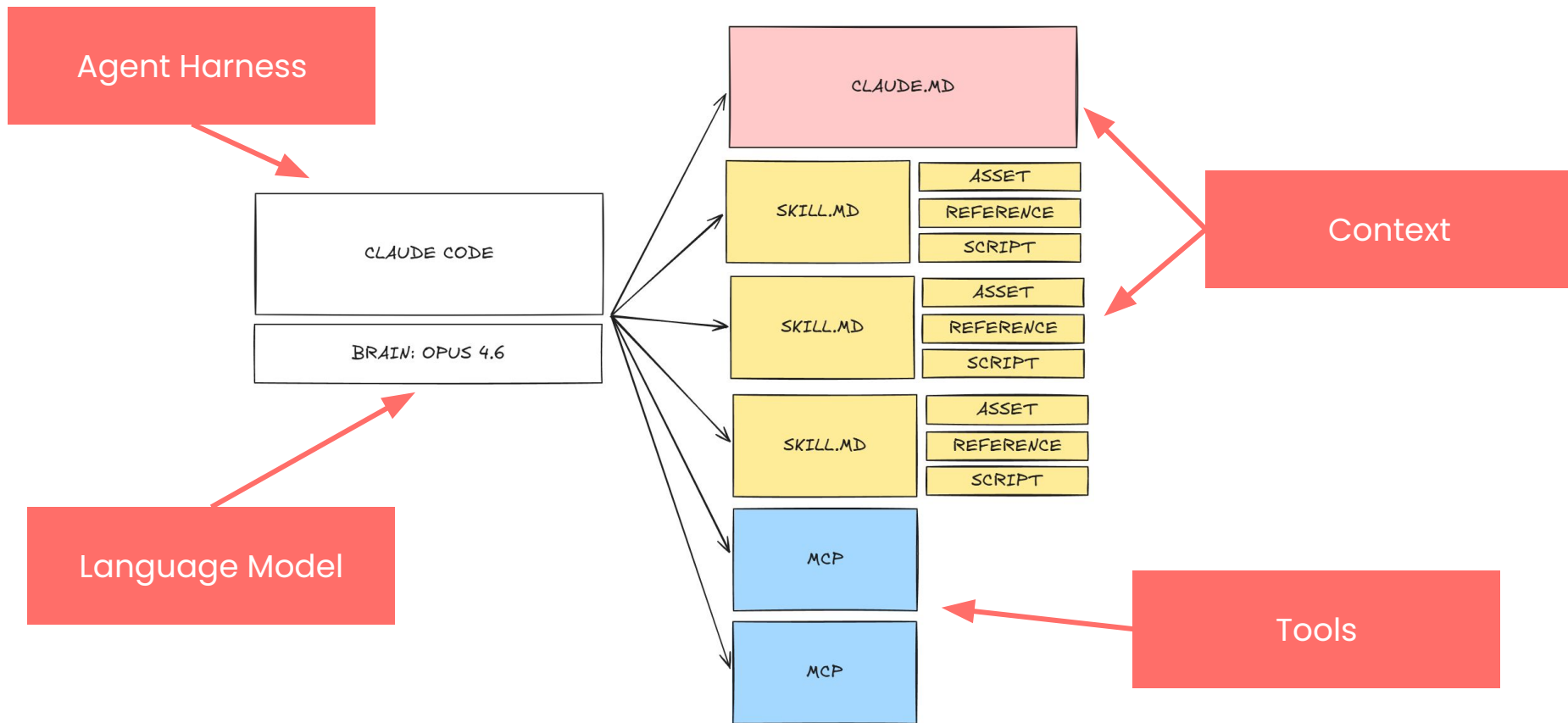
Prompt rein, Antwort raus – vielleicht Code + Erklärung, aber das wars

Claude Code, Codex & co:

Prompt rein, funktionierender Prototyp raus – inklusive Bug-Fixing & co.



```
* Building foundation layer... (36s · ↑ 651 tokens · thinking)
L ■ Phase 1: Foundation – config, models, schema, abstract parser
  □ Phase 2: DV360 parser + registry
  □ Phase 3: Analysis core – KPI, scoring, statistics
  □ Phase 4: Analysis modules – single-dim, pairwise, anomaly, efficiency, viewability
  □ Phase 5: Engine orchestrator
  □ Phase 6: Excel report generator
  □ Phase 7: HTML report generator + charts + templates
  □ Phase 8: CLI + end-to-end integration
```



Input

► [Image #1] - build a simple HTML5 Website that demonstrates the concept shown in the screenshot
└ [Image #1]

- I'll start by checking the project directory, since this appears to be a greenfield build.

```
Searched for 1 pattern (ctrl+o to expand)
└ 2 skills available
```

- The directory is empty - this is a clean greenfield build, so there's no codebase to explore. The screenshot depicts the anatomy of a coding agent (Claude Code specifically): an Agent Harness (Claude Code) driven by a Language Model (the "brain," Opus), fed by Context (CLAUDE.md + SKILL.md bundles with Asset/Reference/Script), and connected to Tools (MCP servers).

Let me clarify a couple of things so the plan matches what you want.

- Claude asked:
 - └ · How interactive should the website be? (Interactive diagram (Recommended) / Static replica / Scrollytelling page)
 - Which visual style should it use? (Hand-drawn / whiteboard / otago branding / Clean modern)
- No response after 60s - continued without an answer

Reasoning

- I've written the plan. Here's the summary:

What it is: A single self-contained `index.html` that recreates the whiteboard diagram from your screenshot and makes it interactive - clicking each block (Agent Harness / Claude Code, Language Model / Brain, Context / CLAUDE.md + Skills, Tools / MCP) reveals a plain-language explanation of that component's role in a coding agent.

Key decisions I made while you were away:

- Interactive diagram (not just a static picture) - best demonstrates the *concept*.
- Hand-drawn / whiteboard style matching the screenshot (same pink/yellow/blue/coral palette, sketchy borders, handwritten-style font).
- One file, zero dependencies - opens by double-click, works fully offline.

One thing I couldn't confirm: whether you'd prefer otago CI branding or a clean modern look instead of the whiteboard style. The plan assumes whiteboard (to match the screenshot) but notes it's easy to reskin.

Ready to code?

Here is Claude's plan:

Plan: "Anatomy of a Coding Agent" – Interactive HTML5 Site

Context

The user shared a whiteboard diagram (Excalidraw-style) that explains the anatomy of a coding agent, using Claude Code as the worked example. They want a simple HTML5 website that demonstrates this concept. The working directory is empty – this is a clean greenfield, single-deliverable build (no existing codebase to integrate with).

The diagram maps four labeled concepts onto concrete parts:

- Agent Harness → the CLAUDE CODE box (the orchestration/runtime layer)
- Language Model → the BRAIN: OPUS 4.6 box (the reasoning engine)
- Context → CLAUDE.md + three SKILL.md bundles, each containing ASSET / REFERENCE / SCRIPT
- Tools → two MCP boxes (Model Context Protocol servers)

Arrows fan out from Claude Code to every context/tool node; coral callout labels point at each group.

Chosen approach (User was away for clarifying questions): an interactive diagram in the hand-drawn / whiteboard aesthetic matching the screenshot. Clicking any component reveals a plain-language explanation of what it is and its role in the agent. This is the most faithful way to "demonstrate the concept" rather than merely picture it.

Deliverable

A single self-contained file: C:\Users\Günther Dietz\Documents\Local\summit\index.html

- All CSS and JS inline – no build step, no external network dependencies (offline-friendly, just double-click to open).
- Handwritten-style font via a system/generic fallback stack (e.g. "Comic Sans MS", "Segoe Print", cursive) so no font download is required and it still evokes the sketchy look.

Layout & Structure

Recreate the diagram faithfully, then layer interaction on top.

1. Header strip – title "Anatomy of a Coding Agent" + one-line subtitle.
2. Diagram canvas (CSS grid / flex, responsive):
 - Left column: stacked CLAUDE CODE box and BRAIN: OPUS 4.6 box, with the coral Agent Harness and Language Model callout labels + arrows pointing in.
 - Center→right: the fan-out nodes – CLAUDE.md (pink), three SKILL.md cards (yellow) each with ASSET / REFERENCE / SCRIPT sub-chips, two MCP cards (blue).
 - Right/side: coral Context and Tools callout labels + arrows.
 - Connector arrows drawn as an inline SVG overlay (single <svg> positioned absolutely over the canvas) so lines fan from the Claude Code box to each node, matching the screenshot.
3. Explanation panel – a side/bottom panel (or modal) that updates on click, showing the selected component's name + a short description of its role. Default state shows an intro.
4. Legend mapping the four coral labels (Harness / Model / Context / Tools) to their color.

Verification

1. Open index.html directly in a browser (double-click) – confirm it renders with no console errors and no failed network requests (fully offline).
2. Optionally drive it with the Playwright MCP tools: navigate to the local file, take a screenshot, and compare against the reference diagram for layout parity.
3. Click each of the four component groups and confirm the explanation panel updates correctly; tab through with the keyboard to confirm accessibility.
4. Resize the window to a narrow (mobile) width and confirm the layout reflows without horizontal overflow.

● Plan approved. I'll build the single self-contained index.html now.

```
• Write(index.html)
  L Wrote 509 lines to index.html
    1 <!DOCTYPE html>
    2 <html lang="en">
    3 <head>
    4 <meta charset="UTF-8">
    5 <meta name="viewport" content="width=device-width, initial-scale=1.0">
    6 <title>Anatomy of a Coding Agent</title>
    7 <style>
    8   :root {
    9     --pink: #f9ccd6;
   10     --pink-2: #f8c9d4;
    ... +508 lines (ctrl+o to expand)
```

Action

- Now let me verify it in a real browser with Playwright – render it, screenshot, and test the interactions.

Called playwright (ctrl+o to expand)

- File protocol is blocked – I'll serve it locally over HTTP.

```
• Bash(cd "/c/Users/Günther Dietz/Documents/Local/summit" && python -m http.server 8931 >/tmp/httpsrv.log 2>&1 &
  sleep 1.5 && curl -s -o /dev/null -w "%{http_code}" ht...)
  L 200
```

Called playwright 3 times (ctrl+o to expand)

- Only a harmless favicon 404. Let me screenshot the default view.

Searched for 1 pattern, read 2 files, called playwright 2 times (ctrl+o to expand)

- This matches the reference diagram closely. Now let me test the interaction – click "Context" and verify the panel updates with dimming.

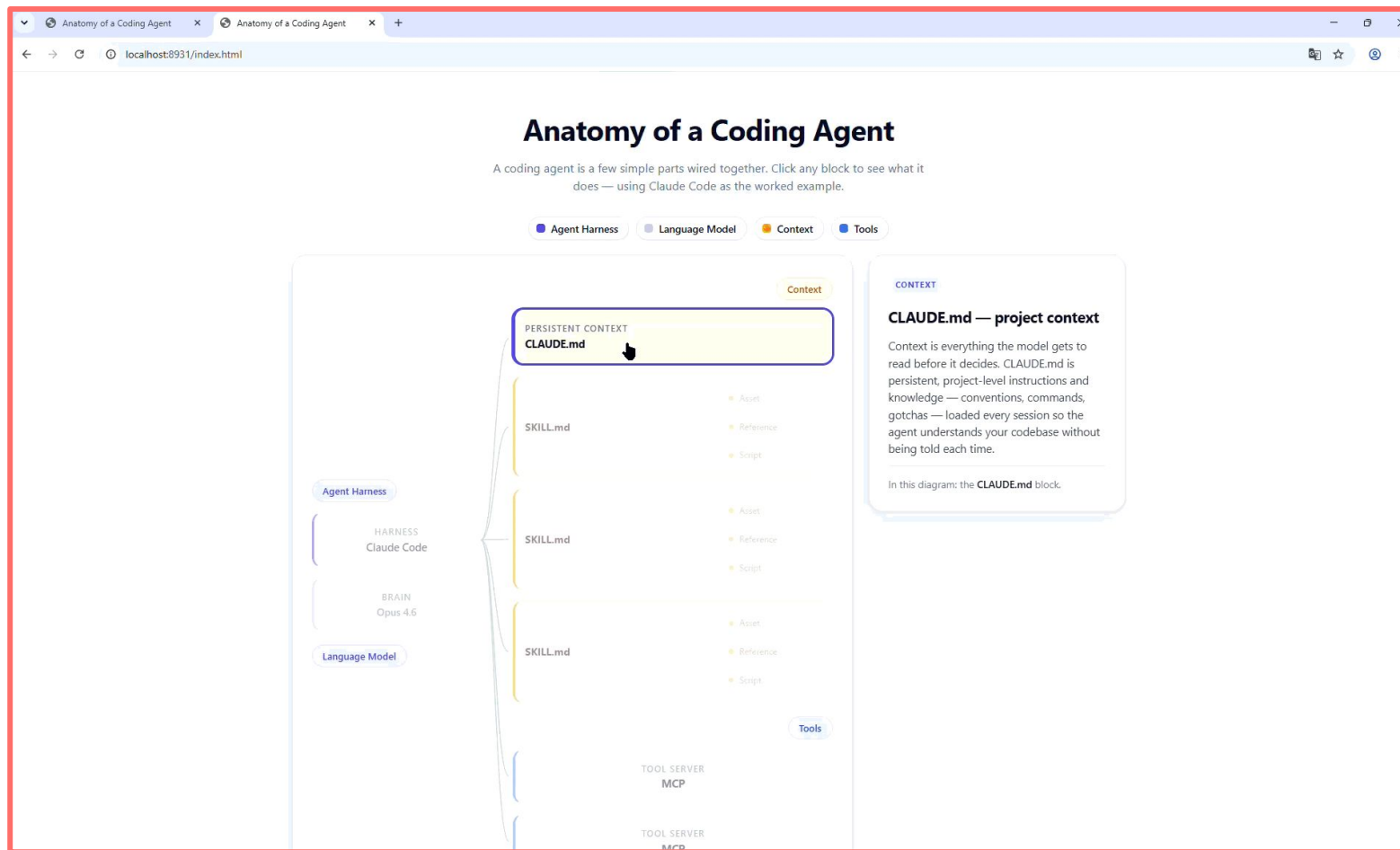
Read 1 file, called playwright 3 times (ctrl+o to expand)

- Interaction works – the selected block highlights with a coral outline, others dim, and the panel updates. Now the mobile reflow check.

Read 1 file, called playwright 2 times (ctrl+o to expand)

- Everything works: the mobile layout reflows to a single column with no horizontal overflow, wires hidden, and the panel moves below. Let me clean up the test artifacts and stop the server so the contains just the deliverable.

Verification



Claude Code Setup

Beispiel

CLAUDE.md

mit allgemeiner Info über das Projekt

Skills

HTML5 Builder Skill (SKILL.md, Reference Files, Assets)

Spec Guard Skill (SKILL.md, Reference Files, Assets)

Frontend-End-Design Skill (SKILL.md, Reference Files, Assets)

MCPs

playwright MCP (Browser Use Access)

Google Ads Library MCP (Sample Banner Access)

3

In der Praxis

Reddit Audience Research



Ars Technica

As Reddit stock falls, CEO questions value of Google's AI Overviews

Reddit may still be considering ending its licensing deal with Google.

2 days ago



Search Engine Journal

Reddit CEO Says LLMs 'Would Not Exist' Without Reddit Data

Reddit CEO Steve Huffman calls user content "modern oil" for AI, discusses deals with Google and OpenAI, and outlines why some companies...

25 May 2026



Reuters

Reddit sues AI startup Anthropic for allegedly using data without permission

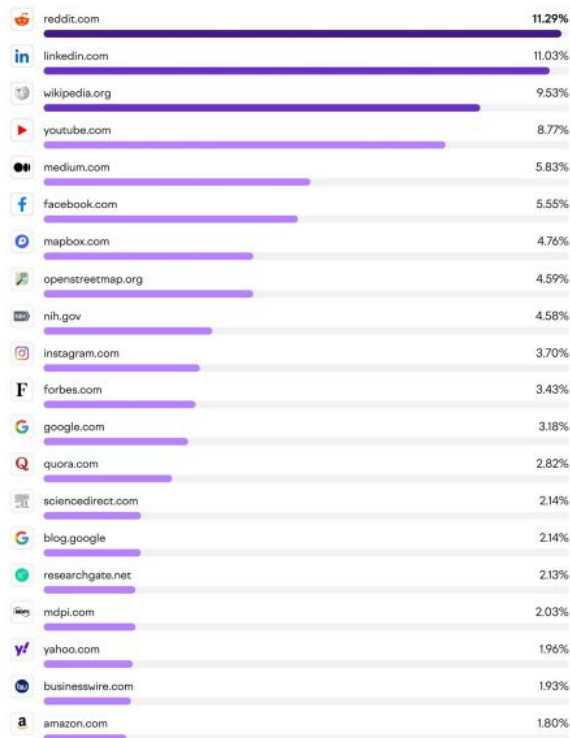
Reddit sued the artificial intelligence startup Anthropic on Wednesday, accusing it of stealing data from the social media discussion...

5 Jun 2025



Top Domains Cited by LLMs (ChatGPT Search, Perplexity, Google AI Mode)

January 2026



Average share of responses with domain citation

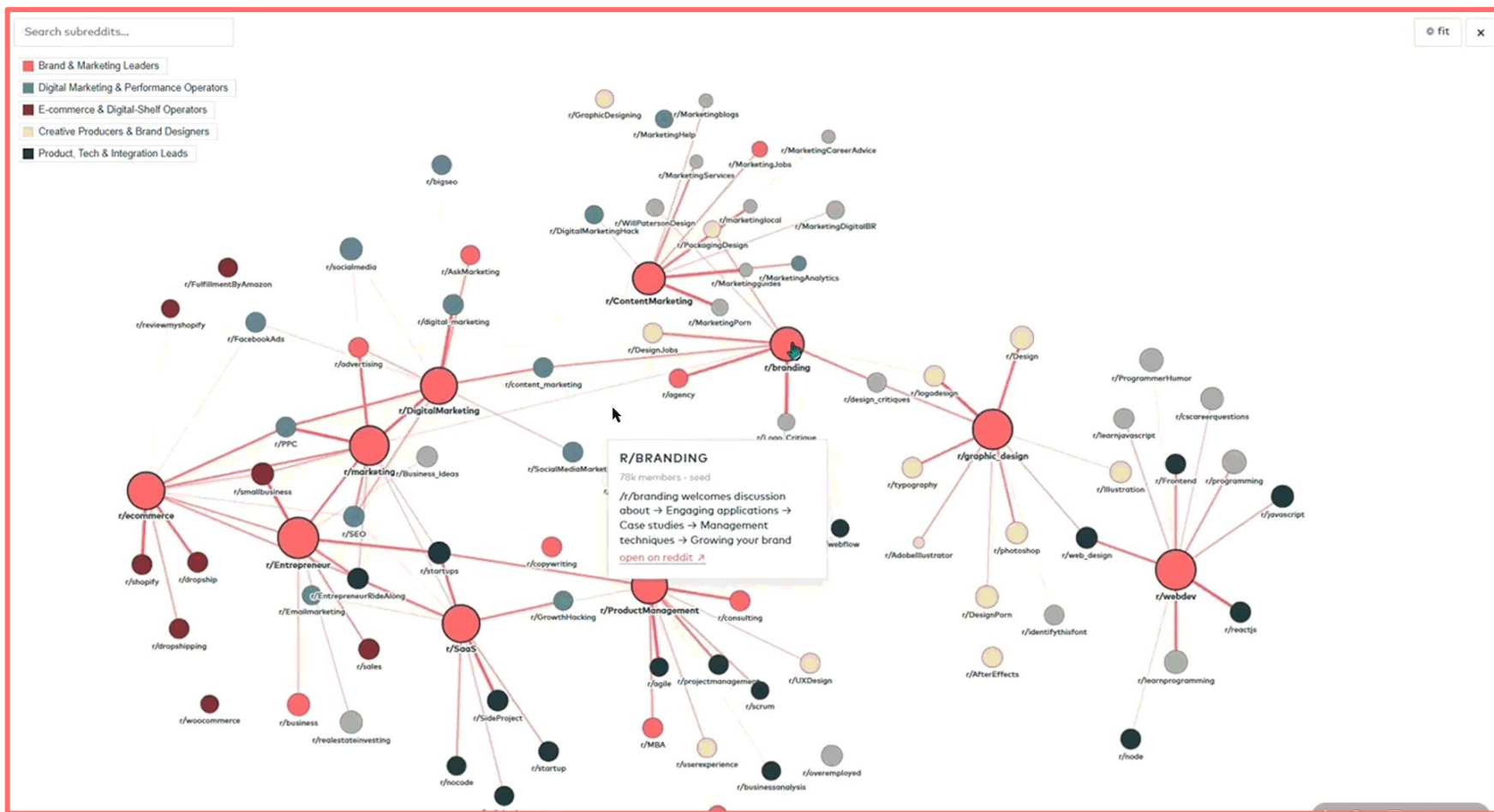


Enter subreddit name

[about](#)
[source code](#)

Welcome!

This website renders graph of related subreddits. [Click here](#) to learn more, or [try demo](#).



Das Ergebnis

otago Kunden-Case

Research Automation

- Findet passende Subreddits
- Nützt Open Source Tools um Verknüpfungen zu scannen
- Findet Wege um zusätzliche Daten auszulesen

Deep Insights

- Erstellt interaktives Tools um die Findings visuell darzustellen
- Schreibt Summary der Findings in unserer CI in ein PDF

Leverage Output

- Kann mit dem Output weiterarbeiten und Reddit Ads Strategie Input liefern (zB Creative Angles und mehr)

3

In der Praxis

Creative Automation



PMAX Auto-Generated Video Assets

Status Quo

expecting nice
on-brand assets



PMAX





So gut wie kein großer (globaler)
Advertiser würde jemals Auto-Generated
Video Assets von PMAX freigeben.

Dieter Günz

Ausgangslage

otago Kunden-Case

- Es wurden mehrere Spots gesichtet, die überhaupt nicht der Kunden-CI entsprachen.
- Das wurde entsprechend bei der Kundin geflagged und über eine Problemlösung nachgedacht.

Expectation



Reality



Brainstorming – Programmatic 🤝 Google Ads

otago Kunden-Case

- Kunde hat eine Vielzahl an tollen Bild-Assets
- Kunde hat einen Dynamic Feed mit vielen Anlaufpunkten (Texte, CTAs, Assets, ...)
- Kunde kann exakt definieren was On-Brand bedeutet



Die Lösung: Claude Code meets Remotion

otago Kunden-Case



Agentic Coding Tool

CI + Remotion Skill

Open Source Video Editor

```
Claude Code v2.1.31

Welcome back Otago!

Opus 4.5 · Claude Max · rechnung@otago.at's
Organization
~\Documents\Local\

> ready to build?
```

X

Make videos programmatically.

Create real MP4 videos with React.
Parametrize content, render server-side and build applications.

`$ npx create-video@latest` Docs Discord GitHub 40k Prompt a video

Das Ergebnis

otago Kunden-Case

Assets at Scale

- 40x On-Brand 15-Sekünder
- Individualisiert nach Produktkategorien
- Hoch- und Querformat

Powerful Engine

- nach Freigabe eines Sample-Assets pro Größe wurde der Rest in wenigen Minuten ausproduziert
- Fehlerquote minimal

Repeatable

- basierend auf dem freigegebenen Schema können jederzeit weitere Creatives in kurzer Zeit und im Bulk ausproduziert werden

3

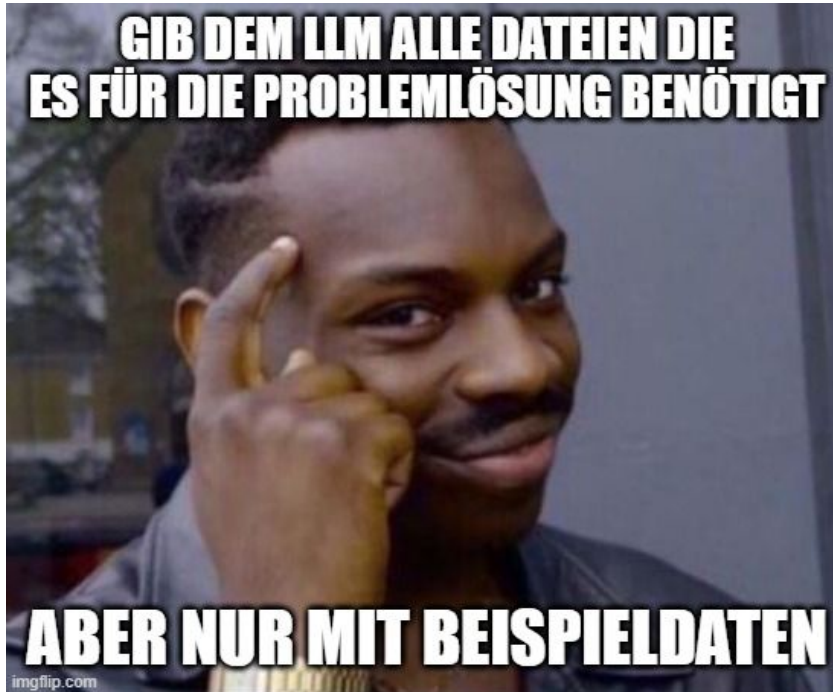
In der Praxis

Lokale Python Audits



Sende keine Unternehmensdaten, tu nur so

otago interner Case





RAW
CSV WITH
200k+ ROWS



NICELY
FORMATTED
INSIGHTS BASED
ON OUR NEEDS

Das Ergebnis

otago Kunden-Case

Automatisierung

- Python-Scripts die lokal laufen
- Zeitaufwendige Tasks die in Sekundenschnelle durchgeführt werden können

Data Protection

- sensible Daten verlassen den Rechner nicht und werden somit an keine LLM-Provider versendet

Improvements

- das System kann laufend optimiert/erweitert werden, man muss nur die lokalen Daten sauber vom LLM/Agent trennen

4

Der Skill Shift 2026



Skill Shift 2026

Was jetzt zählt

1.

Ins Tun kommen

Learning by doing: einfach ausprobieren und parallel Inspiration bei YouTube, Reddit, LinkedIn & Co. holen.

2.

Limits kennen

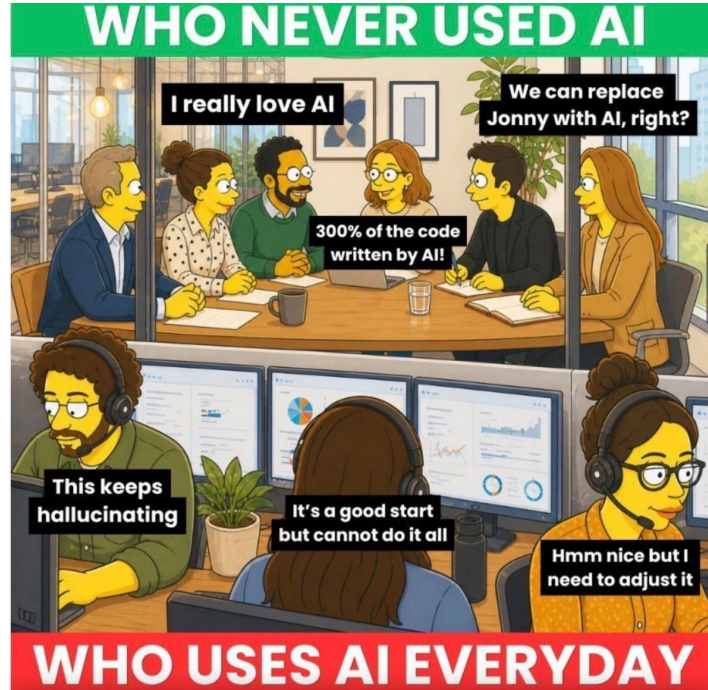
Die Grenzen kennen und akzeptieren lernen – und sie trotzdem immer wieder herausfordern.

3.

Expertise ausbauen

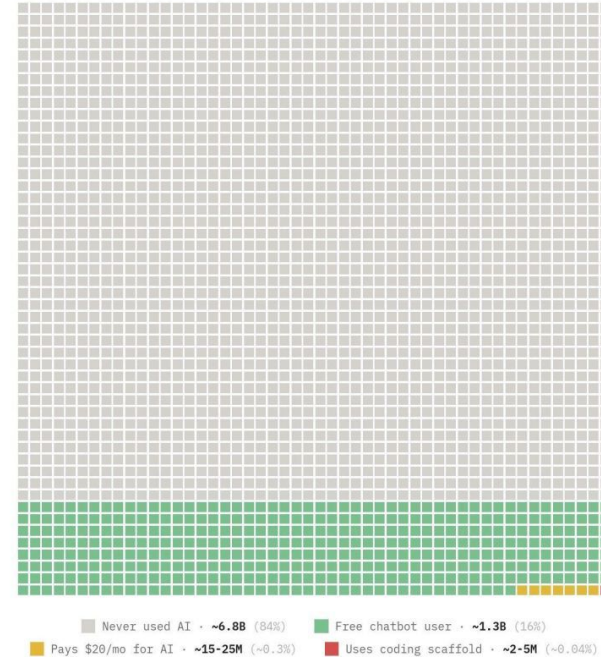
Fachwissen gekonnt einsetzen und weiter aufbauen:
Es macht auch künftig den Unterschied.

We're still early..



Each dot is ~3.2 million people

2,500 dots = 8.1 billion humans. Color = most advanced AI interaction, Feb 2026.



The people doing good writing with AI, are *writers*.
The people doing good coding with AI, are *coders*.
The people doing good strategy with AI, are *strategists*.
The people doing good sales with AI, are *salesmen*.
The people doing good design with AI, are *designers*.

Don't become mediocre at someone else's gifts.
Accelerate your own.



Was hindert euch
am probieren?
Commitment.

Vielen Dank!

Fragen?



Günther Dietz
guenther.dietz@otago.at

Follow me on 

