File 2604.14228, filed 14 April 2026, section: Coding agents

Dive into Claude Code: The Design Space of Today's and Future AI Agent Systems

Jiacheng Liu, Xiaohan Zhao, Xinyi Shang, Zhiqiang Shen

In brief

A teardown of Claude Code's architecture from its public source, compared with two open-source agent systems, OpenClaw and Hermes Agent.

Why builders should care

The clearest public look at how a top coding agent is actually built. Most of the engineering lives around the loop, not in it.

The paper, in plain English

The authors analyze Claude Code's publicly available source and compare it with two open-source agent systems, OpenClaw and Hermes Agent, which answer many of the same design questions.

They identify five values that drive the architecture: human decision authority; safety, security and privacy; reliable execution; capability amplification; and contextual adaptability. These are traced through thirteen design principles down to implementation choices.

The core is a simple while-loop: call the model, run tools, repeat. Most of the code lives around that loop: a permission system with seven modes and an ML-based classifier, a five-layer compaction pipeline for context, four extensibility mechanisms (MCP, plugins, skills and hooks), subagent delegation, and append-oriented session storage.

Across systems, the same questions get different answers: Claude Code emphasizes per-action safety, OpenClaw perimeter-level access, and Hermes per-action approvals across many surfaces. The paper ends with six open design directions.

How to apply it

1. Keep the agent loop boring. Put your engineering effort into permissions, context compaction and extension points.
2. Decide early whether safety is per action or at the perimeter; it shapes your whole UX.
3. Treat session storage as append-only so runs are auditable and resumable.
4. Use the thirteen principles as a design review checklist for your own agent.

Key results

Quoted from the abstract

> *The core of the system is a simple while-loop that calls the model, runs tools, and repeats.*

> *Most of the code, however, lives in the systems around this loop: a permission system with seven modes and an ML-based classifier, a five-layer compaction pipeline for context management, four extensibility mechanisms (MCP, plugins, skills, and hooks), a subagent delegation and orchestration mechanism, and append-oriented session storage.*

> *Claude Code emphasizes per-action safety, OpenClaw emphasizes perimeter-level access, and Hermes renders per-action approvals across many surfaces.*

Summary written from the arXiv abstract. Read the original: https://arxiv.org/abs/2604.14228

Curated by Atharv Pawar