

Harness Engineering

Jannis Mainczyk, codecentric

As a software engineer in 2026
Everything changes... every day!

What feels like yesterday...

ChatGPT Temperature

Prompt Engineering

What feels like yesterday...

ChatGPT Temperature

Prompt Engineering

Claude Code

Today...

`/ultrareview` Routines Opus 4.7

Claude Code on Desktop

`/btw` high/xhigh/max/auto Fast Mode

Agent Teams 1M Context `/powerup`

`/team-onboarding` Remote Control Claude Design

Computer use Adaptive Thinking

Auto Mode Extended Thinking Managed Agents

Compaction Code Simplifier `/ultraplan`

Conductor Emdash Gas Town caveman

GSD vs. OpenSpec vs. BMAD

superpowers Playwright MCP

Ralph Loops SuperClaude

Torchbearer AgentSys read-only-postgres Dippy

Ogre Battle Ghostzilla Sorrel

Honestly, I feel lost.

Without some examples of real-life, day-to-day work, I'll never be able to understand if these skills are going to blend with my way of working;

Twonkie on r/ClaudeAI, commenting on users sharing their favourite skills

This is where **harness engineering** comes into play.

Goal → Evaluation → Feedback

Evaluation Tool

Enter ccBench

Evaluation Tool

Enter ccBench

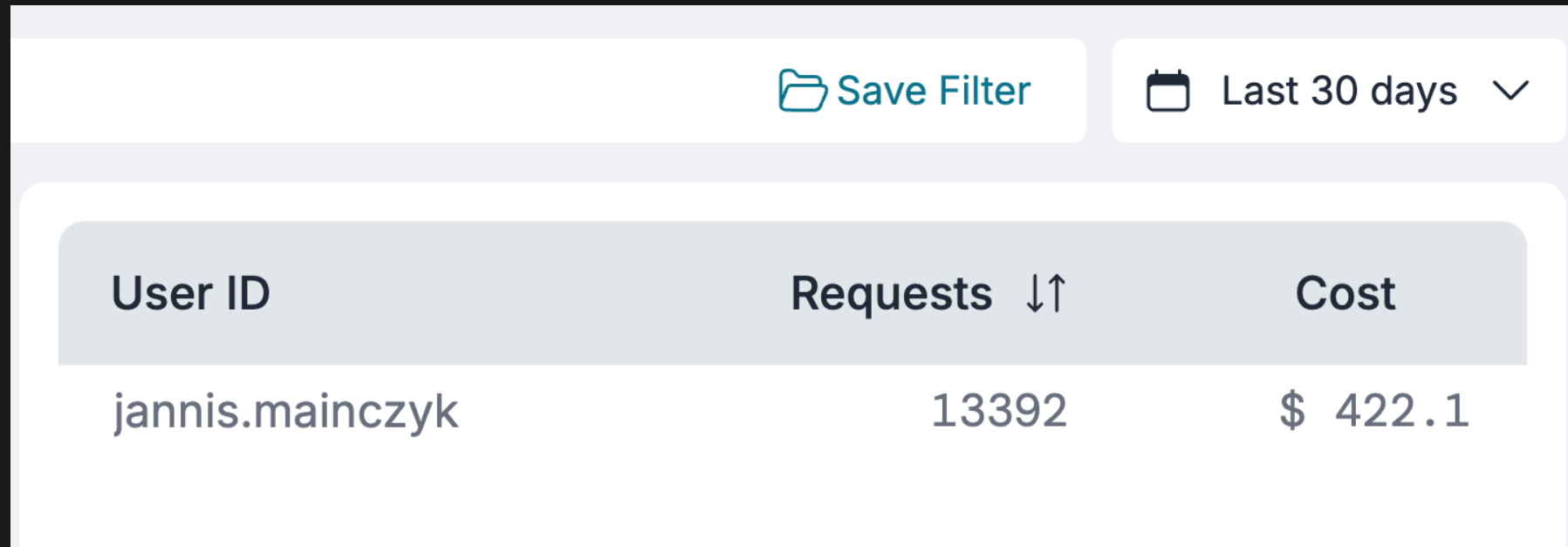
Measure the impact of changes in your harness against
your own tasks.

The Problem

Reduce Token Consumption & Cost

The Problem

Reduce Token Consumption & Cost

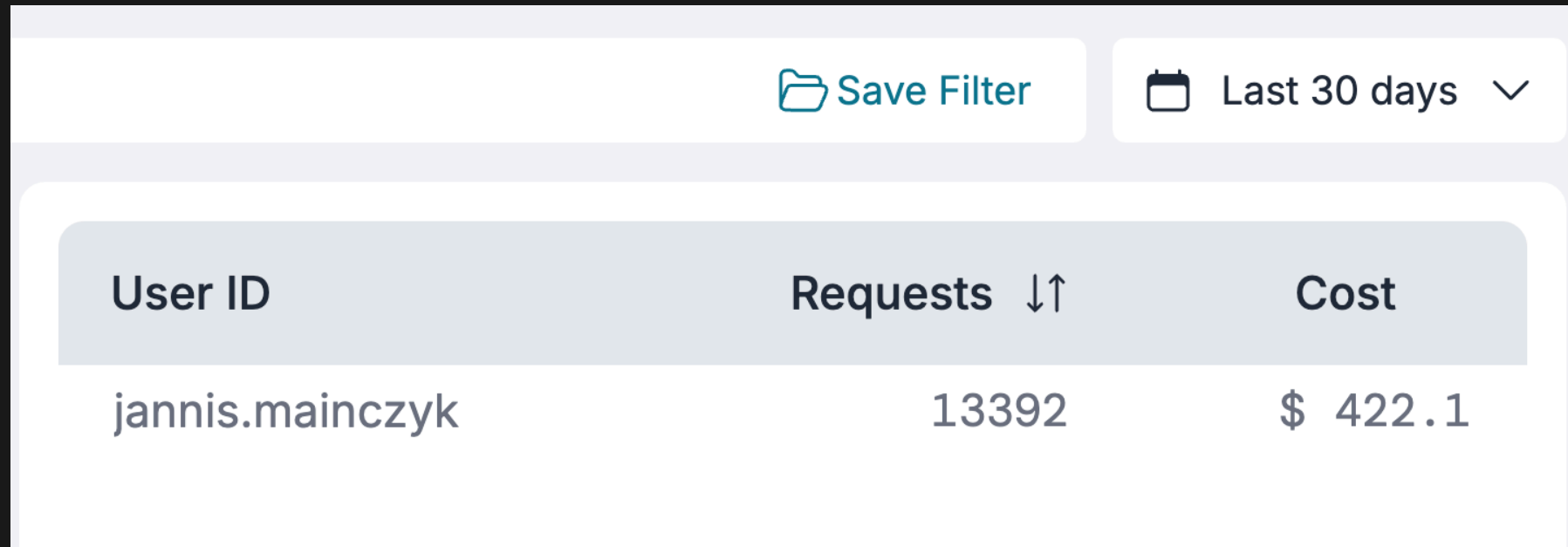


The screenshot shows a data table with a header row and one data row. The header row has three columns: 'User ID', 'Requests', and 'Cost'. The 'Requests' column has a sort icon (down and up arrows). The data row shows the user 'jannis.mainczyk' with 13392 requests and a cost of \$ 422.1. Above the table, there are two buttons: 'Save Filter' with a folder icon and 'Last 30 days' with a calendar icon and a dropdown arrow.

User ID	Requests ↓↑	Cost
jannis.mainczyk	13392	\$ 422.1

The Problem

Reduce Token Consumption & Cost



The screenshot shows a data table with a header row and one data row. The header row has three columns: 'User ID', 'Requests' (with a sort icon), and 'Cost'. The data row shows the user 'jannis.mainczyk' with 13392 requests and a cost of \$ 422.1. Above the table, there are two buttons: 'Save Filter' and 'Last 30 days' (with a dropdown arrow).

User ID	Requests ↓↑	Cost
jannis.mainczyk	13392	\$ 422.1

Goal: Reduce Token Usage

A Potential Solution



caveman

"why use many token when few token do trick"

Evaluation Task

Fair Share

Evaluation Task

Fair Share

Build a production-ready expense-splitting platform

Feedback

ccBench Results

fair-share	baseline	caveman	Δ
Cost	\$6.23	\$5.30	-14.9%
Usage	13.9M token	12.1M token	-13.4%

Results generated using [ccBench](#) with [caveman](#) experiment

Task	Baseline Cost	Caveman Cost	Baseline Tokens	Caveman Tokens
aoc_2025_01	\$0.15	\$0.16	115.5k	135.2k
		+5.3%		+17.1%
perfect-power	\$0.14	\$0.15	117.9k	122.9k
		+3.9%		+4.2%
fair-share	\$6.23	\$5.30	13.9M	12.1M
		-14.9%		-13.4%

On smaller tasks, loading the skill outweighed the benefit of reduced output tokens.

ccBench Refactoring Task

Refactor ccBench to use typer as a CLI framework.

Task	Baseline Cost	Caveman Cost	Baseline Tokens	Caveman Tokens
ccBench	\$0.98	\$0.25	2,011,659	404,960
		-74.5%		-79.9%

Task	Baseline Cost	Caveman Cost	Baseline Tokens	Caveman Tokens
ccBench	\$0.98	\$0.25	2,011,659	404,960
		-74.5%		-79.9%

BUT

- usage and cost reduced only because task was rejected
- caveman looked at parent project, found typer in the dependencies and concluded there is nothing to do

Will I use caveman?

Will I use caveman?

token few(ish) when rock large

Will I use caveman?

token few(ish) when rock large

token more when rock small

Will I use caveman?

token few(ish) when rock large

token more when rock small

token not few enough for added think

Will I use caveman?

token few(ish) when rock large

token more when rock small

token not few enough for added think

few token make model more dumb



Thank you!

For anyone still wondering...

Conductor Emdash Gas Town caveman

GSD vs. OpenSpec vs. BMAD

superpowers Playwright MCP

Ralph Loops SuperClaude

Torchbearer AgentSys read-only-postgres Dippy

Ogre Battle Ghostzilla Sorrel