

A digital illustration of a swarm of bees flying over a field of flowers. The bees are depicted with detailed wings and bodies, some in flight and others on the ground. The background is a soft, hazy landscape with tall grasses and various flowers, creating a natural and serene atmosphere. The text is overlaid on the center of the image.

Directing a Swarm of AI Agents for Fun and Profit

Adrian Cockcroft – April 9th, 2026

Why did I choose this title?

Directing a Team of Agents - Management skill set

You aren't watching everything they do

You have to think and structure work like a director level manager

The agents build things you didn't ask for (or didn't know to ask for)

You have to make sure they remembered to run tests etc.

(Just like managing human dev teams)

But they do several days work in 15 minutes...



Fun and Profit

Coding with AI is fun!
Learn how to use the latest
tools earlier
Experiment on your own pet
projects
For a few \$\$, it's worth it to
help keep up



Code for work needs to be safe
Commercial deals take time
We need to be more careful and
responsible for paid work
*Speed of AI tool adoption is a
competitive advantage for your
workplace and career*

“The future is already here, it’s just unevenly distributed”

There is too much happening too fast

It will be confusing and complicated

Each month there is a new and better way to do AI

You can’t buy this expertise, you have to learn it in-house

Learning and sharing is the key...

Quote by William Gibson

NoOps to NoDevs to NoMans

Attempts to get VMware Ticket-and-ClickIt Ops people to use cloud failed, the Devs used cloud and it took the Ops people a few years to re-skill as DevOps to run the cloud platform...

Attempts to get developers to use AI coding tools run into the same problem. We need to teach the Product Managers to use the tools and let the Devs reskill as ProdDevs to run the platform..

Making it Work

Techniques for better coding results – BDD Testing

Ask for a step by step plan, a UX guide and human testing instructions and check them before coding

Ask for Behavior Driven Development (BDD) tests

Given, When, Then structured in PyTest - check they make sense

Use the same G/W/T for mocked unit tests and integration tests

(Note that async pytest-bdd doesn't work, use sync only)



Tricks for better coding results – Maintain Context

“Create and maintain a context block comment at the beginning of each file recording design intent, versions, bug fixes etc.”

The first thing an LLM reads when it looks at the file...

Improves code quality, reduces bugs, focuses effort

Portable context between different coding tools

Tricks for better coding results – Tidy First

Clean up each phase of work before adding more features (Kent Beck)

Ask for a code review, refactor if needed

Make all compile and run time errors and warnings go away

Optimize slow code paths and add structured logging if appropriate

Update block comments, README and other docs, archive out of date docs

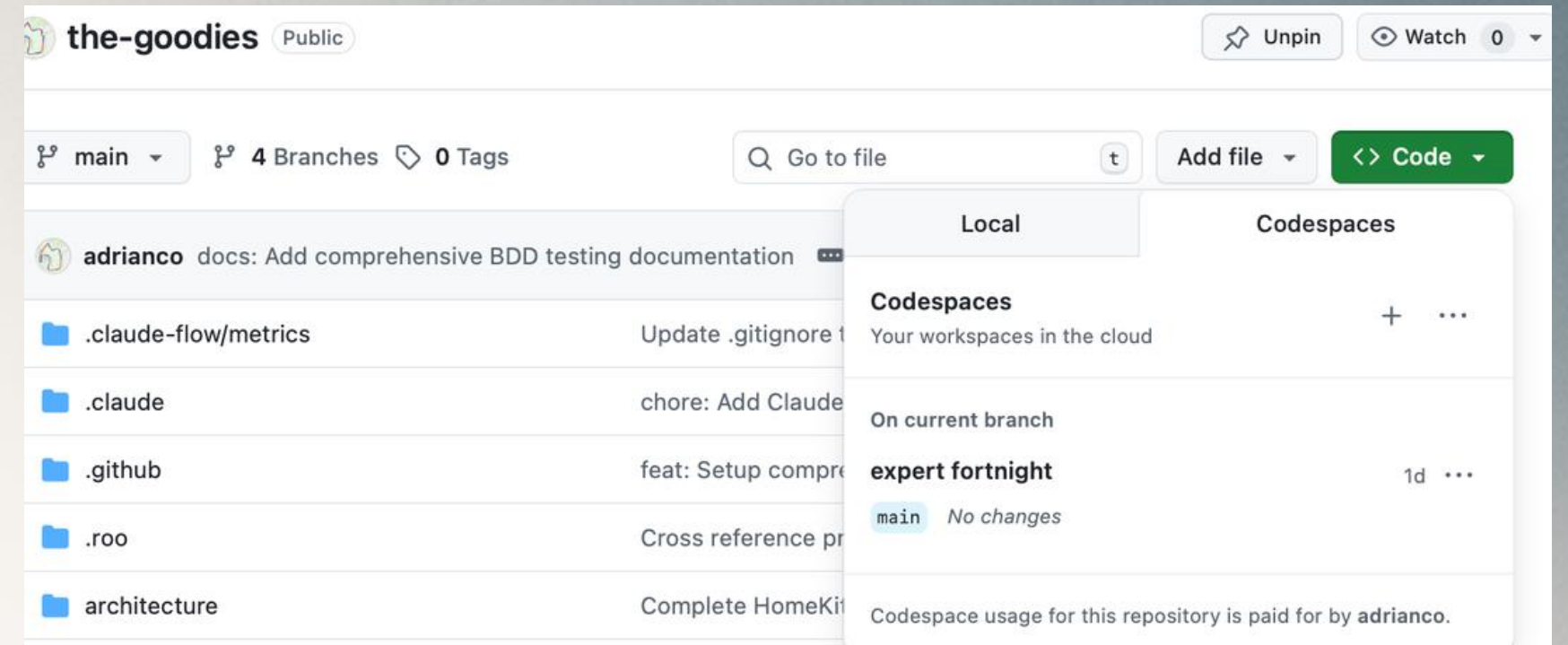


Github Codespaces

A safe place to run agents based on Azure/Linux

All the agents can do is commit to the repo, so its safe to let them run:
\$ `claude --dangerously-skip-permissions`

Monthly (\$20 worth) Codespaces free tier is plenty for fun projects
Safe container allows faster iteration and unattended agent work



Claude-Flow/RuFlow - Open Source by Reuven Cohen

Popular and powerful, but fast changing and too many options

Use to run an agent swarm in Github Codespaces

What is an Agent Swarm?

- 5-8 MCP specialized copies of Claude-code

- Architect, UX front end, Back end, Tester, DevOps etc.

- Communicating via ToDo lists and a shared memory service

Results are *significantly* better and faster for large projects

LLM as UX Designer

Ask Claude to write a UX guide

It will make up personas and onboarding flows for you

Provide feedback until it looks like what you want

Ask Claude to write a Human Testing guide, for someone coming across a github repo for the first time, what do they need to do to install and run this code, with example results. Test it yourself.

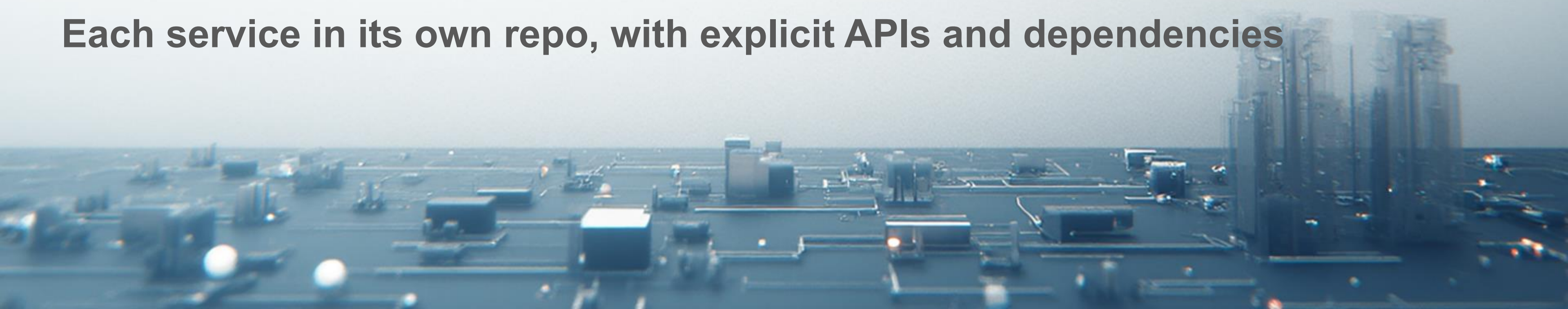
Microservices vs. Monoliths (redux)

The agents will build you a big messy “ball of mud” monolith If you let them...

Several people or agent teams will conflict working on the same code base

This is why we did Microservices in the first place, independently deliverable single function services. Easier to maintain developer context for changes.

Each service in its own repo, with explicit APIs and dependencies



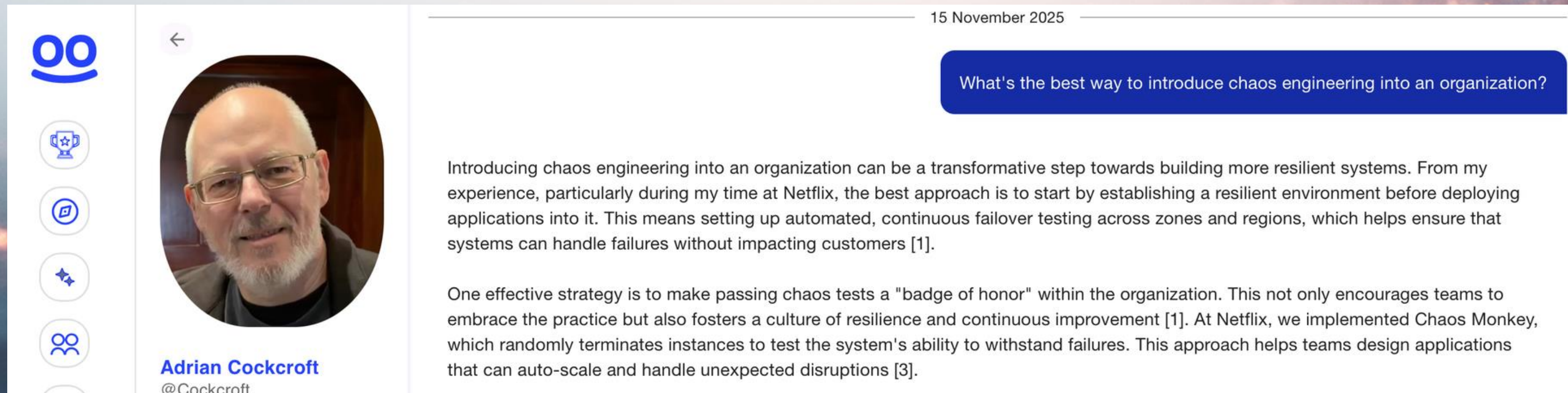
The “Hello World” of
LLMs

is to build an MCP server

Persona Based LLM Developments

All your content, blog posts, podcasts, books, social media, in an LLM Persona as a Service - Soopra.ai - see <https://soopra.ai/Cockcroft>

Build it into an MCP server? <https://github.com/adrianco/megpt>
One of my early attempts at vibe coding...



The screenshot shows a social media post from Adrian Cockcroft (@Cockcroft) dated 15 November 2025. The post features a blue header with the question: "What's the best way to introduce chaos engineering into an organization?". The main text discusses the importance of building a resilient environment before deploying applications, mentioning his experience at Netflix and the use of Chaos Monkey. It also describes a strategy of making chaos tests a "badge of honor" to foster a culture of resilience and continuous improvement.

15 November 2025

What's the best way to introduce chaos engineering into an organization?

Introducing chaos engineering into an organization can be a transformative step towards building more resilient systems. From my experience, particularly during my time at Netflix, the best approach is to start by establishing a resilient environment before deploying applications into it. This means setting up automated, continuous failover testing across zones and regions, which helps ensure that systems can handle failures without impacting customers [1].

One effective strategy is to make passing chaos tests a "badge of honor" within the organization. This not only encourages teams to embrace the practice but also fosters a culture of resilience and continuous improvement [1]. At Netflix, we implemented Chaos Monkey, which randomly terminates instances to test the system's ability to withstand failures. This approach helps teams design applications that can auto-scale and handle unexpected disruptions [3].

Adrian Cockcroft
@Cockcroft

Consciousness as an Observability

Model

[Medium Post](#) by adrianco



Photo taken by Adrian at the Park Hyatt NingBo, China

Perhaps Consciousness is Just Human Observability?

 adrian cockcroft  6 min read · Jun 29, 2019

My definition of consciousness: **The things that go away when you are unconscious**

Conscious things can interact and share their state: How are you? What are you thinking about?

We can use LLMs to make things conscious

My experiment is based on my own house I want a knowledge graph MCP service as a mobile app that knows how to do stuff...

Consciousness Coding Projects

Reuven Cohen showed me...

Vibe Coding is so “Last Month...” — My First Agent Swarm Experience with claude-flow

adrian cockcroft 20 min read · Jun 26, 2025

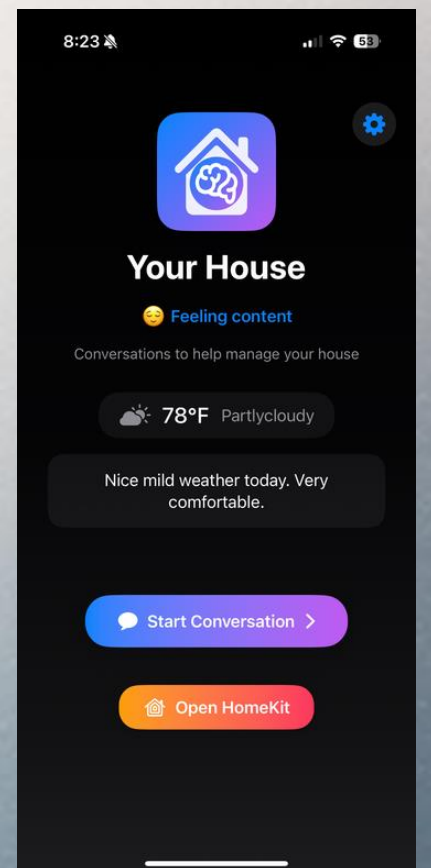
139 9



First attempt to use Claude-code Python based Consciousness Service
Tried to build too much in one go
Ran but didn't work - June 2025

Second attempt to use Claude-code Native Swift iOS app - Sept 2025
Harder to get working than Python
Includes Weather, HomeKit, TTS and voice recognition chat

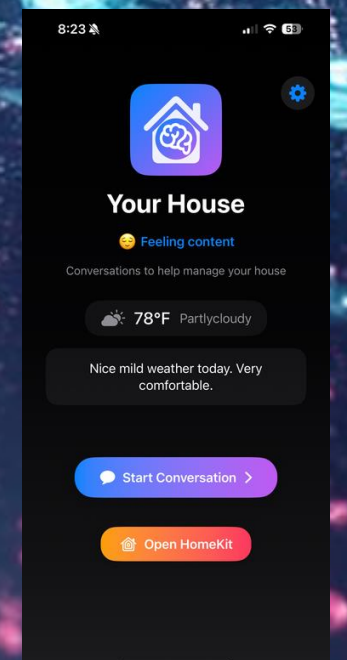
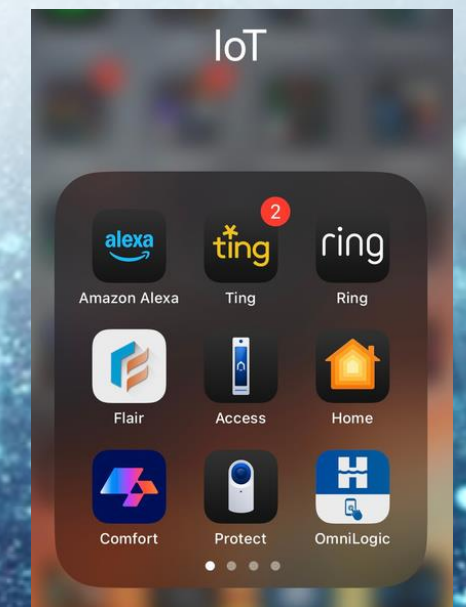
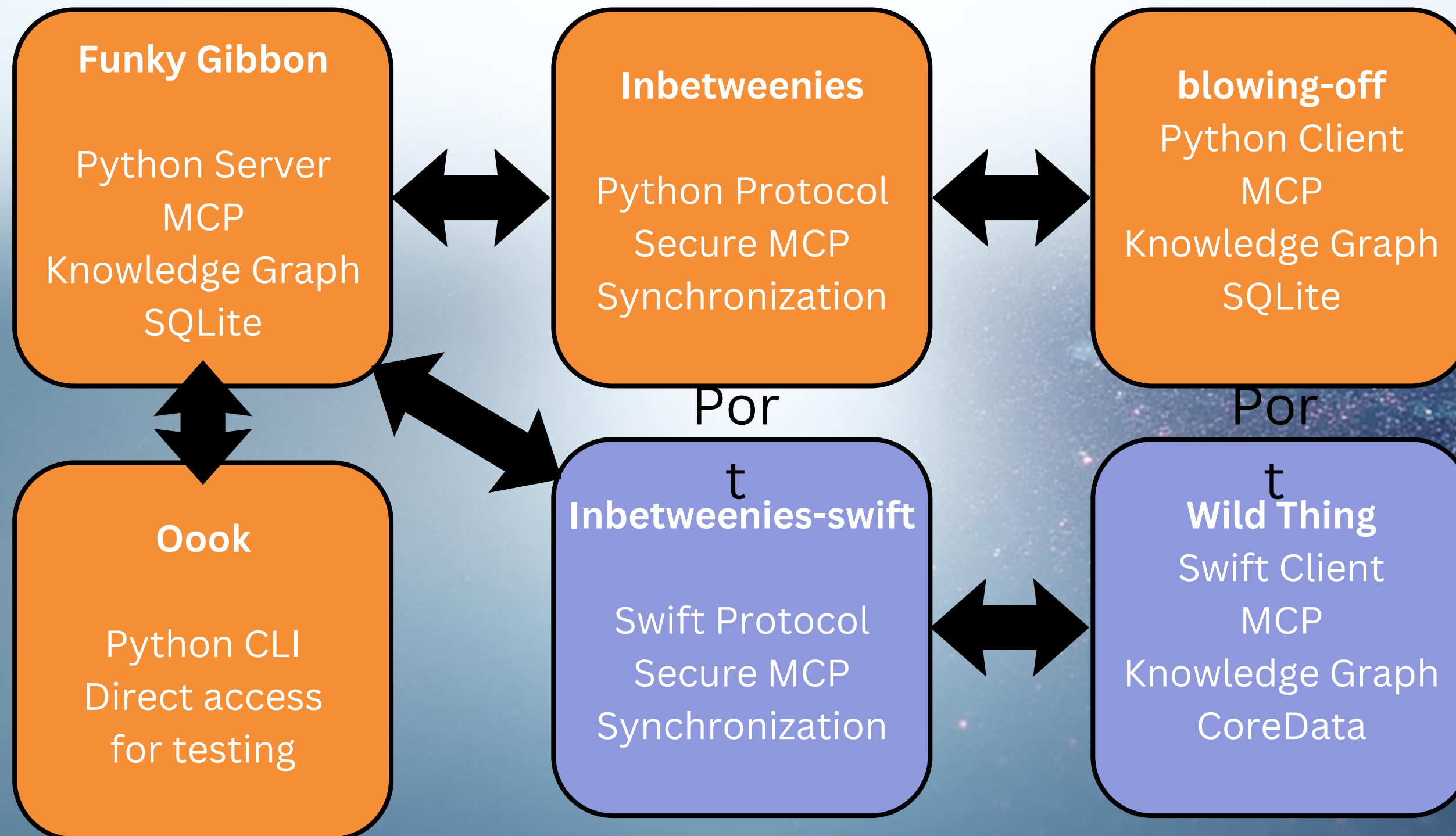
No backend for mobile as yet...



Distributed MCP Knowledge Graph - The Goodies

<https://github.com/adrianco/the-goodies> - structured high quality clean code. All 225 tests pass. Nov 2025.

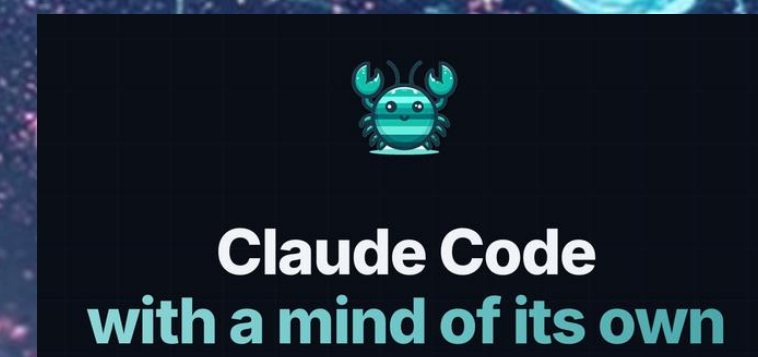
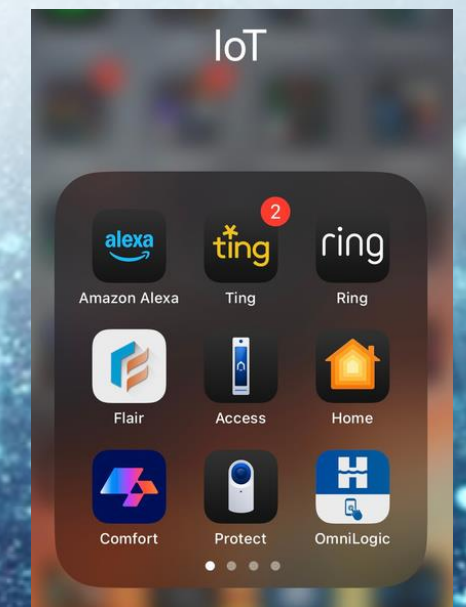
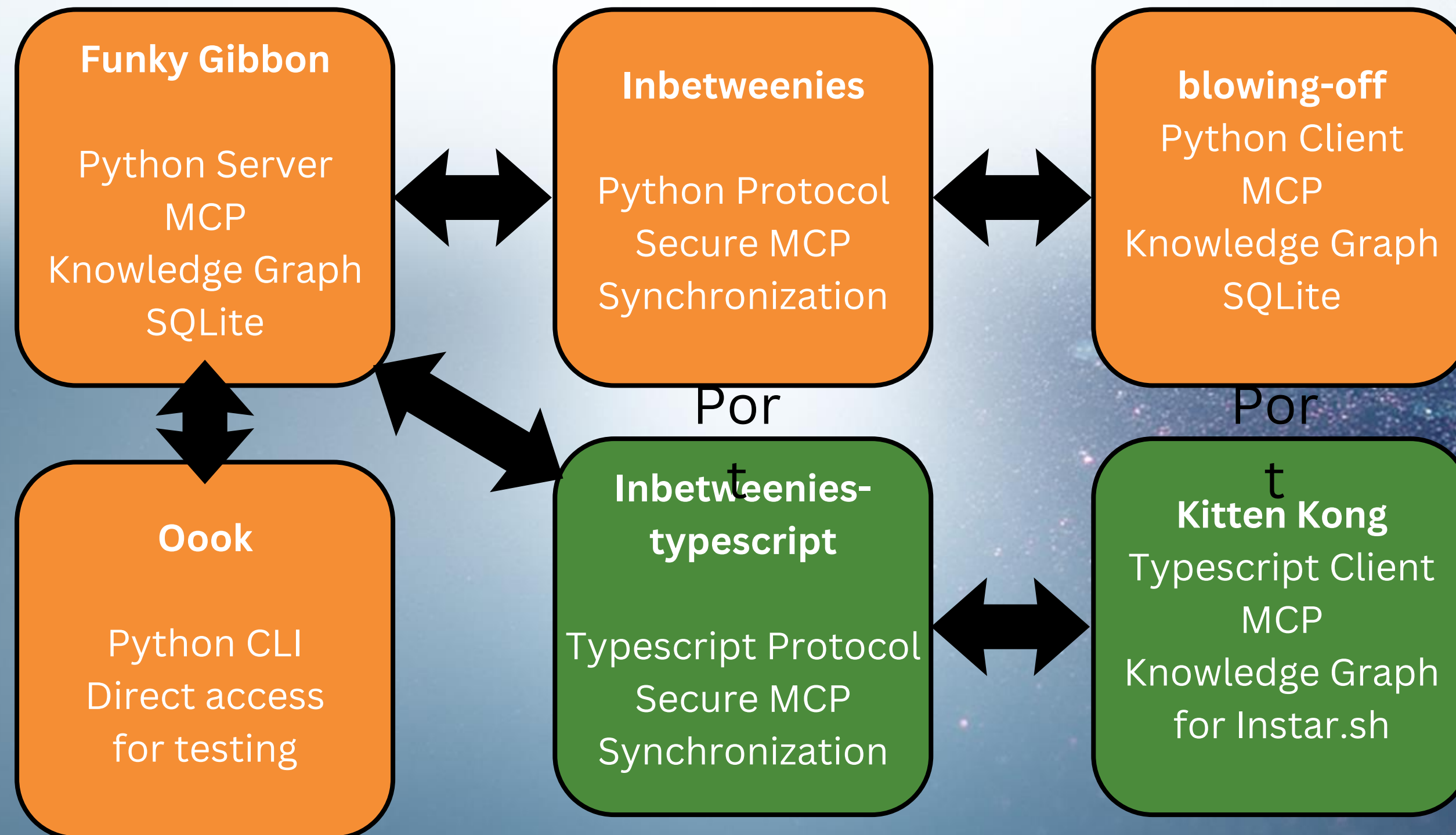
Security: JWT authentication, rate limiting, audit logging. Access Control: Admin and guest role QR code access



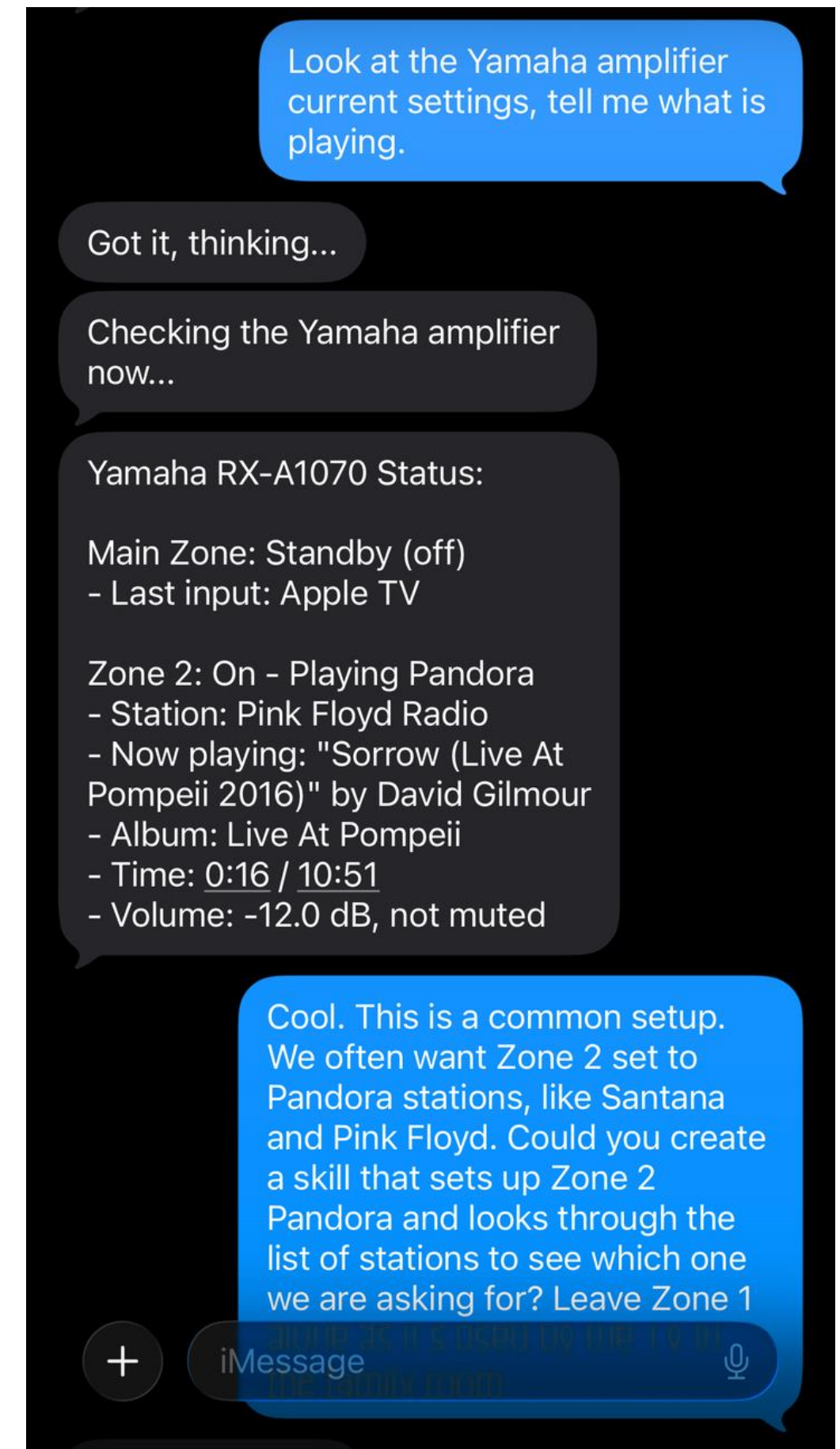
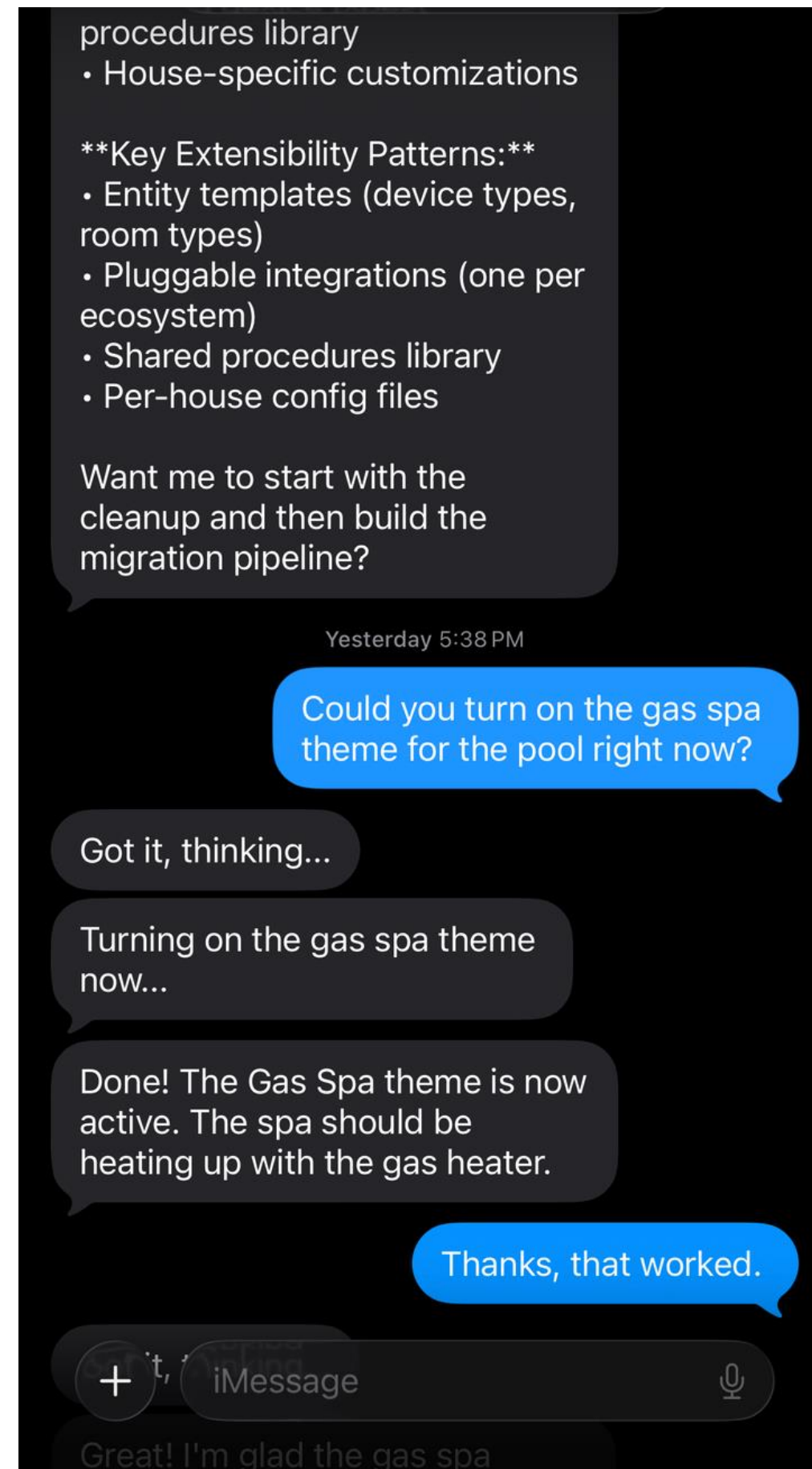
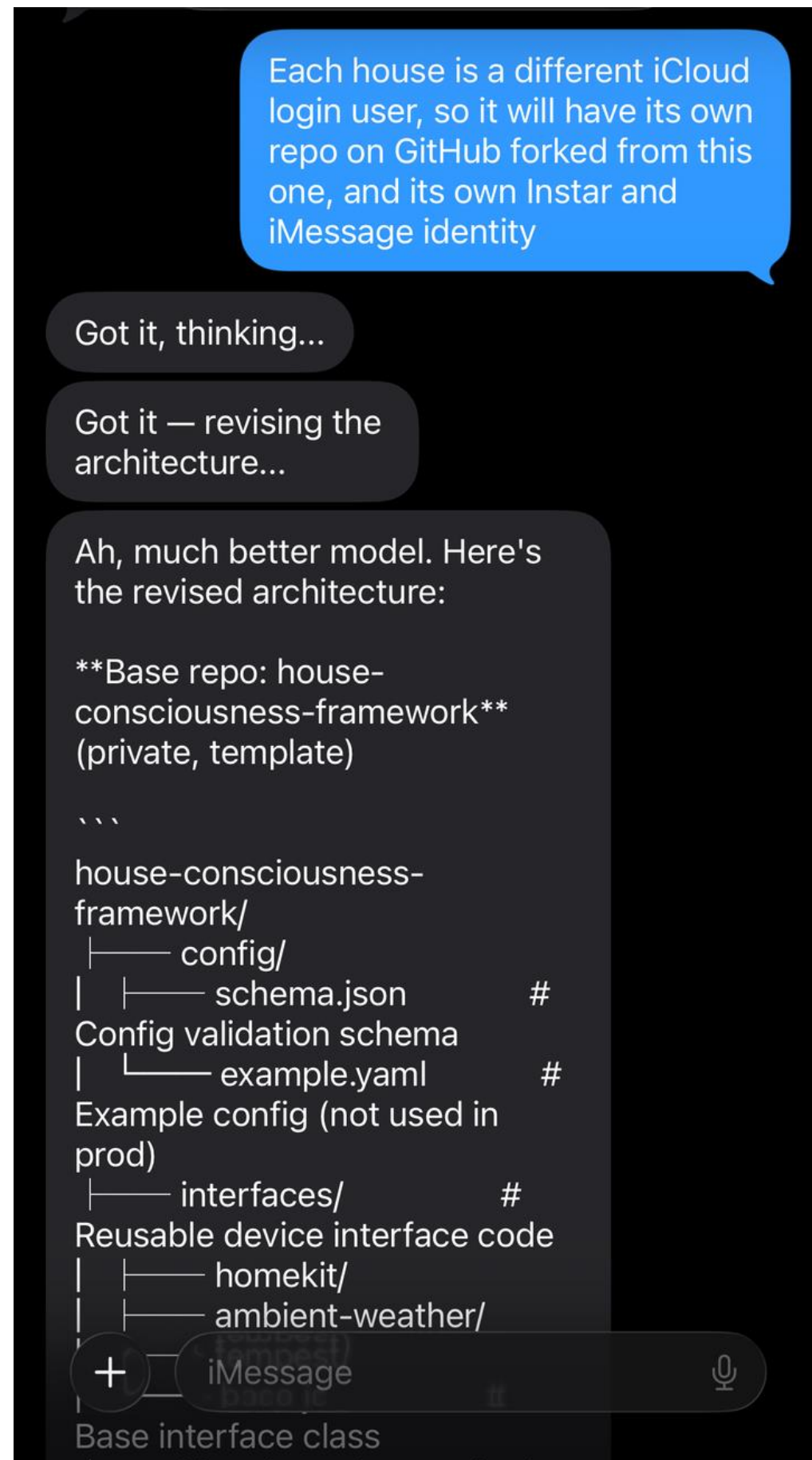
Work
in
Progress

Autonomous Agent - Instar via iMessage chat

Instar is a simple OpenClaw like system based on Claude-code that I'm customizing into house consciousness
See <https://instar.sh> and <https://github.com/rolandcanyon-cmd/the-goodies-typescript> - April 2026



Chatting with my house on iMessage while sitting in the hot tub listening to Pink Floyd...



Demo Agentic Coding at Nubank - This is all I said



Adrian: If I wanted to create an MCP knowledge graph for Brazilian soccer players, teams and games, is the data available to populate the graph?

Claude:... <too much output to show>

Adrian: don't need real time, need free access, this is for a demo, not commercial use

Claude:... <too much output to show>

Adrian: I will want to connect the MCP server to Claude and have a set of sample questions for the demo that Claude looks up via MCP to come up with the answers

Claude:... <too much output to show>

Adrian: save this entire conversation as a document that I can implement elsewhere



Test Questions from the doc Claude wrote

Simple Lookups:

1. "Who scored the most goals for Flamengo in 2023?"
2. "What teams has Neymar played for in his career?"
3. "When was the last match between Palmeiras and São Paulo?"

Relationship Queries:

1. "Which players have played for both Corinthians and Palmeiras?" (rival teams)
2. "Show me all Brazilian players who moved to European clubs in 2024"
3. "What teams did Ronaldo Nazário play for during his career?"

Aggregate/Analysis:

1. "Which team has won the most Brasileirão titles?"
2. "Who are the top 5 goal scorers in Brazilian Serie A history?"
3. "What's Flamengo's win rate against Internacional in the last 10 years?"

Complex Multi-hop:

1. "Find players who scored in a Copa do Brasil final and later played in Europe"
2. "Which coaches have managed multiple championship-winning teams?"
3. "Show me teammates of Neymar at Santos who also made it to European leagues"



Brazilian Soccer MCP

Benchmark

Run a swarm with the same objective to see how the swarm results evolve...

Setup as Github organization [GitHub.com/brazil-bench](https://github.com/brazil-bench) with bench template
Attempts are evaluated, compared, issues filed using Pourpoise SOP scripts
There are 9 attempts in the repo, including Claude-flow, Beads and Gastown

Top 10 Leaderboard

Rank	Attempt	Pattern	Score	Spec	Tests	Integration	Skip%	Issues
1 🏆	2025-12-14-python-claude-beads-2	Beads v3	91.8	16/16	59	× External	20%	2 open
2 🥈	2025-10-30-python-hive	Hive v1	90.9	15/16	64	× External	0%	3 open
3 🥉	2025-12-15-python-claude-ruvector	RuVector	90.1	16/16	61	× External	0%	2 open
4	2025-12-13-python-claude-hive	Hive v2	86.0	16/16	43	✓ testcontainers	0%	0
5	2026-01-13-python-gastown	Gastown	85.4	16/16	40	× External	0%	1 open
6	2025-09-30-python-swarm	Swarm v1	83.5	16/16	63	✓ testcontainers	0%	0
7	2025-12-13-python-claude-swarm	Swarm v2	82.8	16/16	37	× Skip	3%	1 open
8	2025-12-14-python-claude-beads	Beads v2	78.4	16/16	25	✓ testcontainers	0%	0
9	2025-12-01-python-claude-beads	Beads v1	75.5	16/16	18	× Mock	0%	1 open

- **Best Overall:** Beads v3 (100% compliance, 59 effective tests)
- **New Entry:** Gastown (#5) - fastest implementation (~18 min), zero fix commits
- **testcontainers leaders:** Hive v2, Beads v2, and Swarm v1 have self-contained integration tests
- **Most Tests:** Hive v1 (64), Swarm v1 (63) - both with 0% skip ratio
- **Full Compliance:** 8 of 9 attempts achieve 16/16 (Hive v1 at 15/16)
- **Cleanest Code:** Gastown (0 fix commits, 40 tests pass first try)
- **6 of 9 fail integration:** 3 use testcontainers (Hive v2, Beads v2, Swarm v1)



YMMV: Questions?

**Ask: Join Brazil-bench
and contribute attempts
for frameworks and agents**