

The Solo Fleet, Reviewed

CLI-First & Local-First — Strengths, Limits, and What Companies
Should Steal

An honest look at the playbook behind one developer maintaining many apps — including where
it breaks.

Uwe Jügel · Ubunatic Coding

The Setup

One Developer, Many Living Apps

1

developer (plus agents)

n

focused apps, all maintained

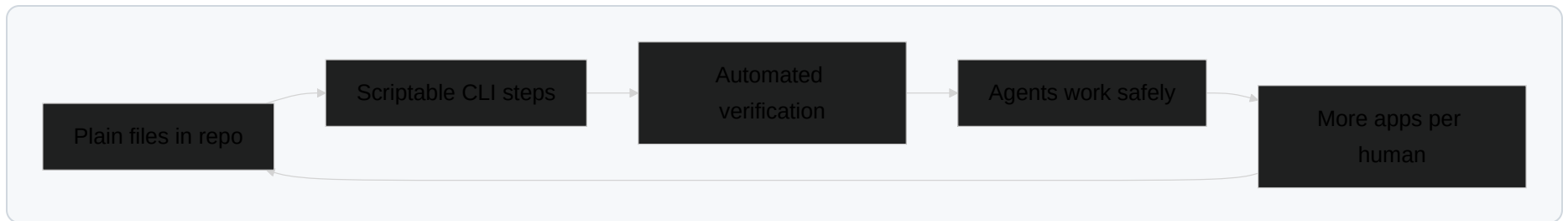
1

command to verify any repo

*Building was never the bottleneck. **Maintenance is where fleets sink** — unless verification is automated and context lives in files.*

Why It Works Here

The Solo Leverage Loop



- Zero coordination cost: every decision is a file edit away.
- CLI tools compose into canaries, gates, and pipelines. GUIs compose into stale wiki pages.
- Agents and future-you read the **same files**, get the **same truth**.

The Honest Part

Preconditions Disappear at Scale

Decisions Leave the Repo

Pricing, priorities, escalations happen in meetings and chats. Insisting on Markdown-only truth makes engineering invisible to PMs, sales, and leadership.

Files Don't Coordinate People

Ten people in flat `issues/` files recreate — badly — what trackers solved: assignment, notification, paging. Files don't wake anyone at 3 a.m.

The Honest Part II

Ownership, Governance & Money

Bus Factor

The conventions are documented; the **taste** that evolves them is not. One-author systems are liabilities with great test coverage.

Governance

"Human in control" assumes the operator owns the outcome. With 400 humans, control *correctly* becomes audit trails and approval layers.

Deadlines

Sometimes shipping the band-aid **is** the rational call. A method with no vocabulary for acceptable debt reads as perfectionism with a Makefile.

Meetings Are the API of Organizations

The Alignment Layer Is Synchronous

- The solo fleet has no meetings — organizations **are** meetings: demos, reviews, decisions in rooms.
- A purist who communicates only through versioned Markdown is, organizationally, **offline**.
- This repo ships slide decks for a reason. You are looking at one.

*Failure modes of local-first at scale are failures of **scope**, not substance.*

What Companies Should Steal

Six Transfers That Survive Contact with Reality

Verify First, as Policy

Fund preflight gates and golden suites **before** adopting agents. Never create faster than you verify.

ADRs in the Repo

Keep the tracker — stop losing the **why**. Two-paragraph decision files end code archaeology.

Agent-Ready Repos

`AGENTS.md` + spec + one verify command: the best onboarding doc ever written, for humans and agents alike.

One-Command Truth

`make preflight` — green or red, local or CI, no dashboard login on the critical path.

Platform Canaries

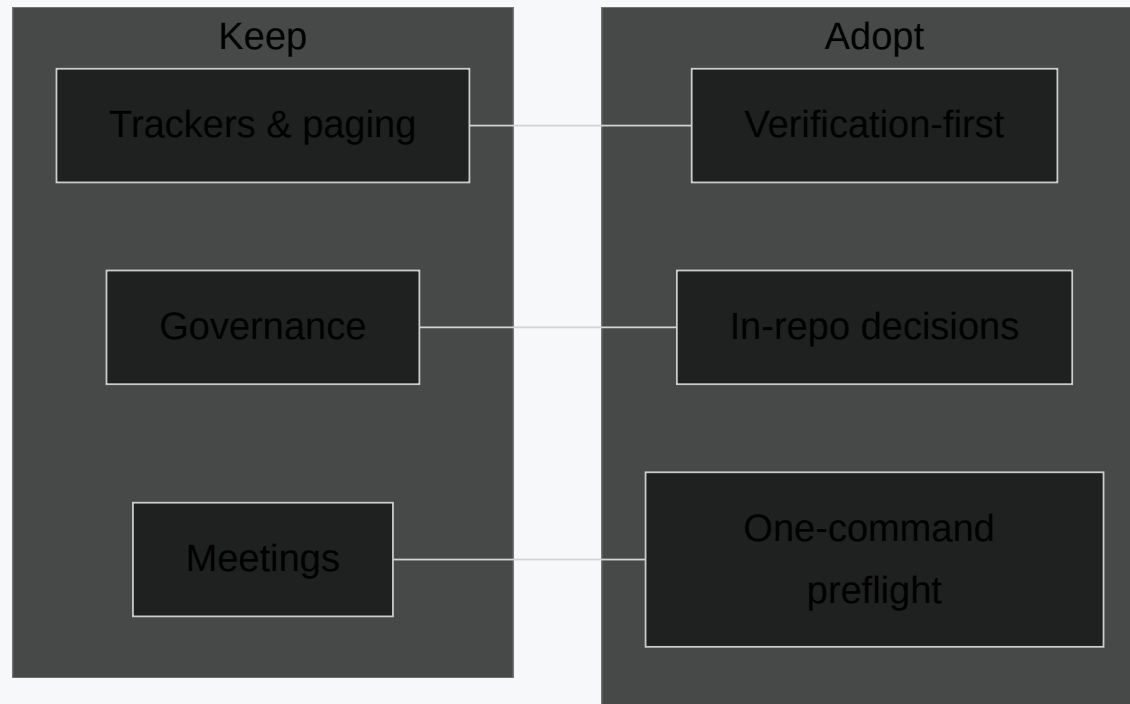
Thirty lines proving the bucket is writable and the token refreshes kill whole classes of cross-team tickets.

Spec-Driven Config

One schema-validated spec file beats "ask Dave which value is real." Reviewable, diffable, generatable.

The Substrate, Not the Operating Model

Keep Your Process. Fix Your Repos.



*CLI-first and local-first optimize the layer where code is **made and verified** — and at that layer, they win everywhere.*

The Closing Reflection

Repositories That Never Forget

- Every repo self-explaining, self-verifying, operable with nothing but a checkout.
- Checks that never sleep; agents that can finally be trusted with real work.
- If **one** developer with a terminal keeps a dozen apps alive —

imagine what a team could keep alive, once its repos can explain themselves.