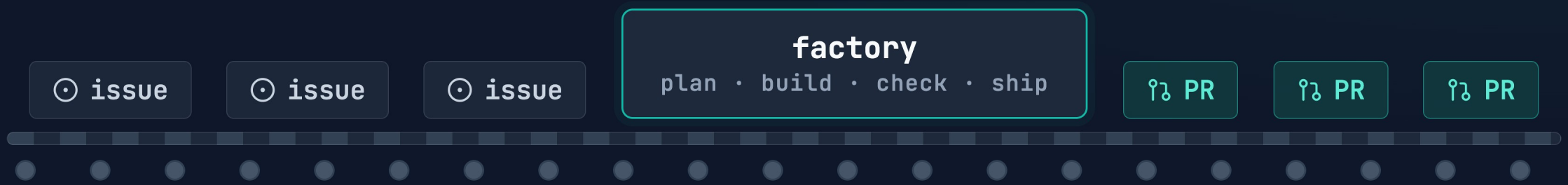
Meet the Software Factory

From ticket to PR while you're in the meeting



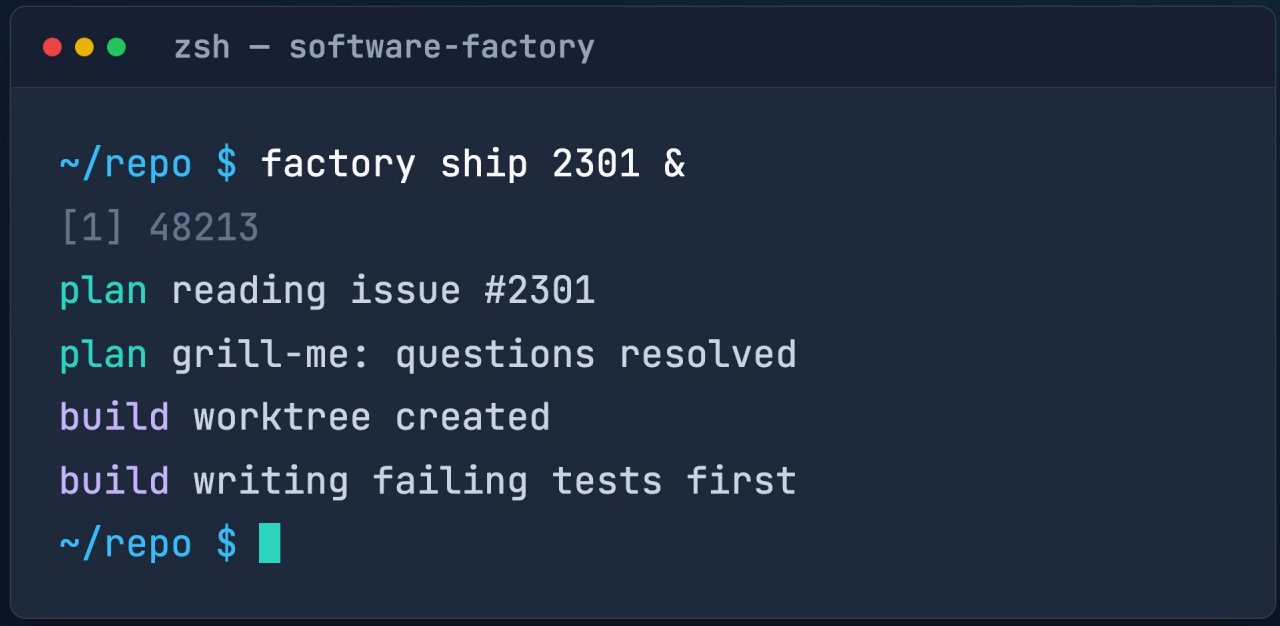
Patrick Robinson · On PAR Dev

github.com/on-par/software-factory

// LIVE RUN

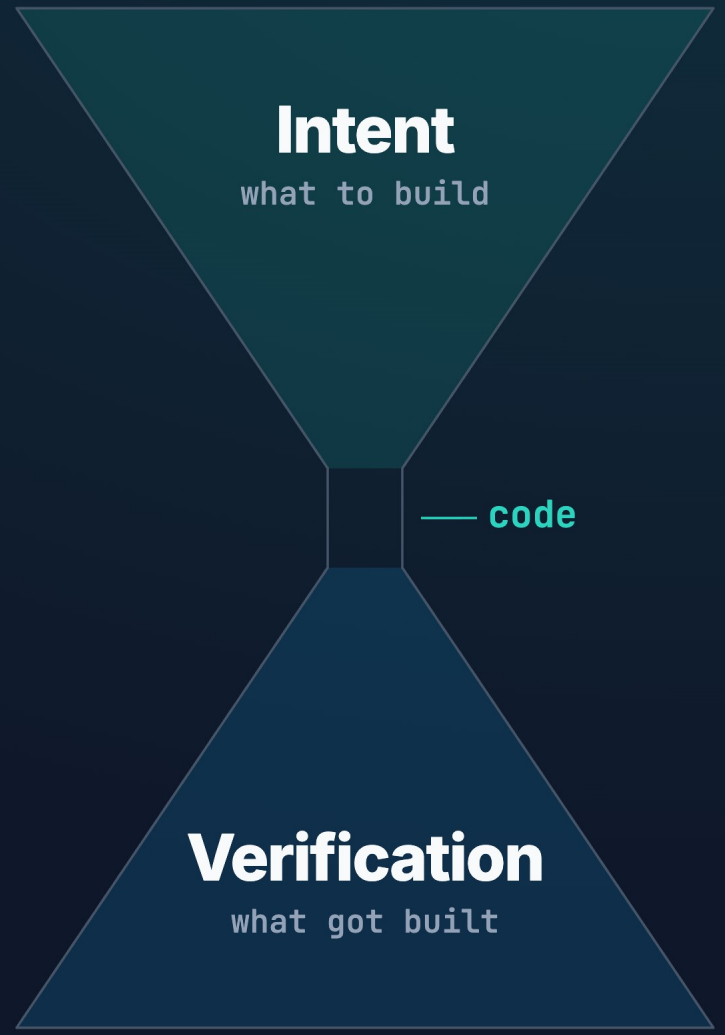
I started a run 2 minutes ago.

We'll check on it at the end.



// THE NEW BOTTLENECK

Writing code isn't the bottleneck anymore



Deciding what to build and verifying what got built is.

// THE NUMBER

3,900+ PRs

Most of them built by the factory, on the factory

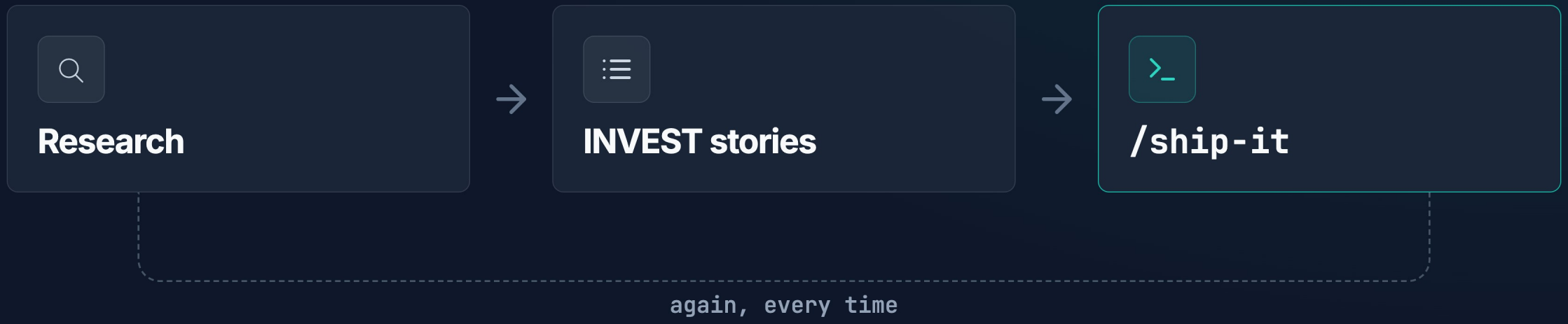
14,355

GitHub contributions in the last year



// HOW IT STARTED

I kept doing the same loop



If you do it the same way every time, codify it

// INSIDE THE LOOP

/ship-it

01 **grill-me** › 02 **code + verify loop** › 03 **AI code review** until clean › 04 **draft PR** › 05 **resolve comments** › 06 **CI green** › 07 **human approves**

Every arrow is a gate, not a suggestion

// THE BASH SCRIPT ERA

v1 was a bash script

It lasted 3–5 days

ship.sh

```
#!/usr/bin/env bash
set -euo pipefail
for issue in "$@"; do
claude -p "/ship-it $issue"
done
```

WHAT BROKE IT

- Parallel work
- Worktrees
- Resuming failures
- Multiple machines

// IT HAS A NAME

Harness engineering

a.k.a. building a Software Factory

// THE PIPELINE

Four stages, one direction



Issue in. Ready-for-review PR out. You decide what merges.

// THE COMMANDS

Six commands run the floor

`factory triage`

AI proposes the queue

`factory ship`

One issue, end to end

`factory run`

Parallel lanes

`factory supervise`

Unattended — waits for usage headroom

`factory land`

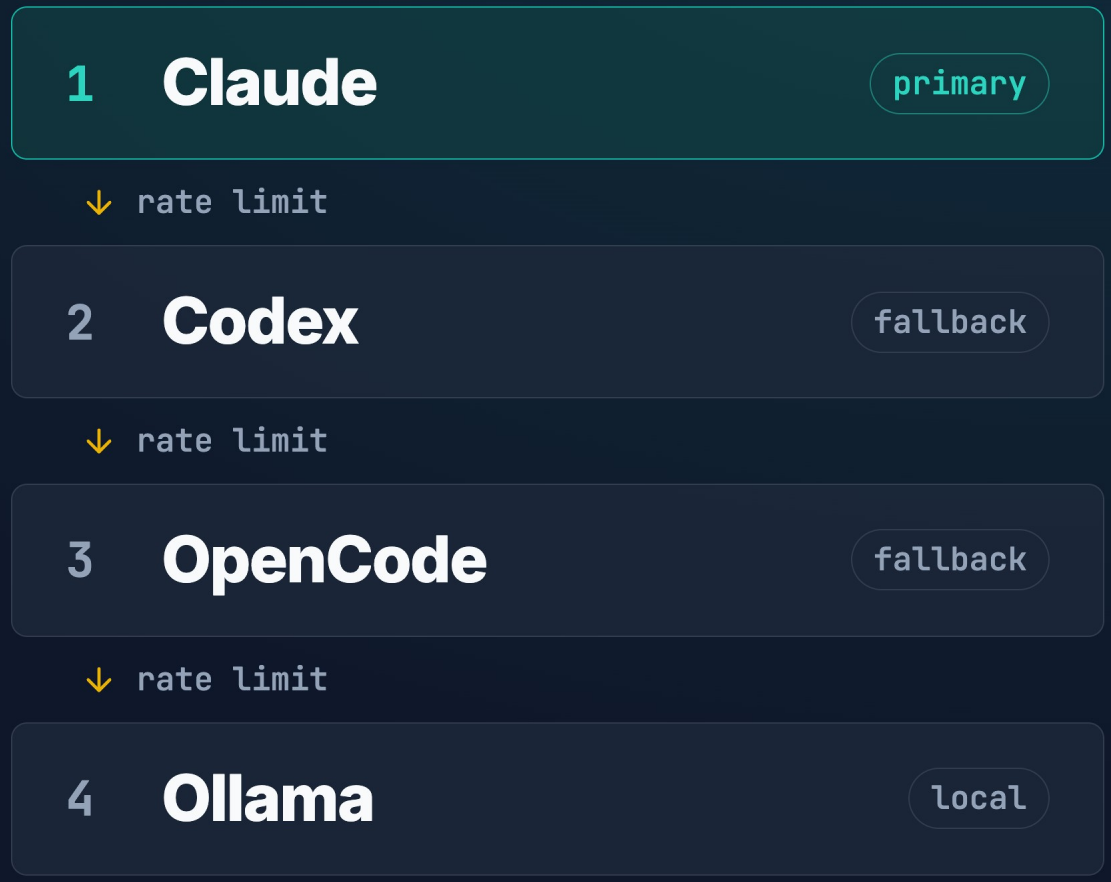
Merge on green

`factory cost`

Per-task spend

// MODEL ROUTING

Subscriptions as a failover chain



Hit a rate limit? It moves to the next model automatically.

// CONSTITUTIONS

Standards live in the repo



Every clone and every agent follows the same rules

// SAFETY

Prompts off. Guardrails on.



Worktree per build

one task, one checkout



OS sandbox

no ~/.ssh · no ~/.aws

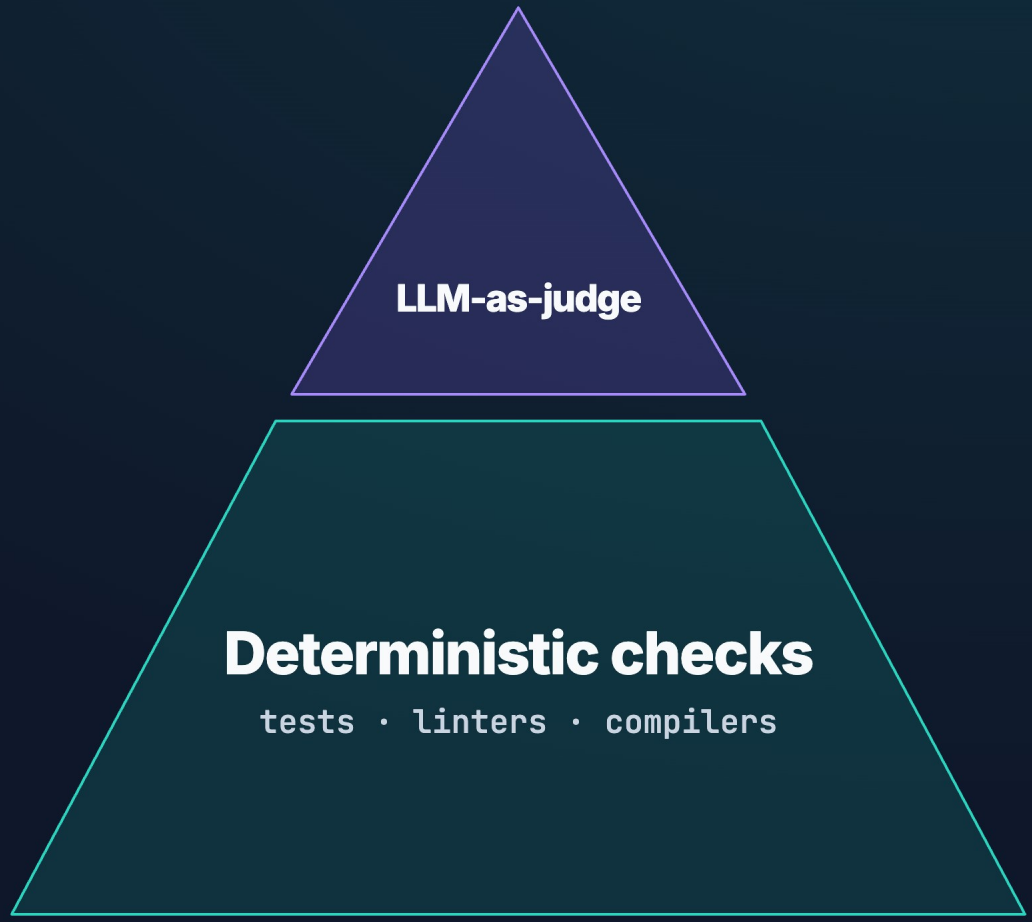


Never merges by default

a human decides

// THE THESIS

Delegate the work. Then delegate the oversight.



Machines check the machines



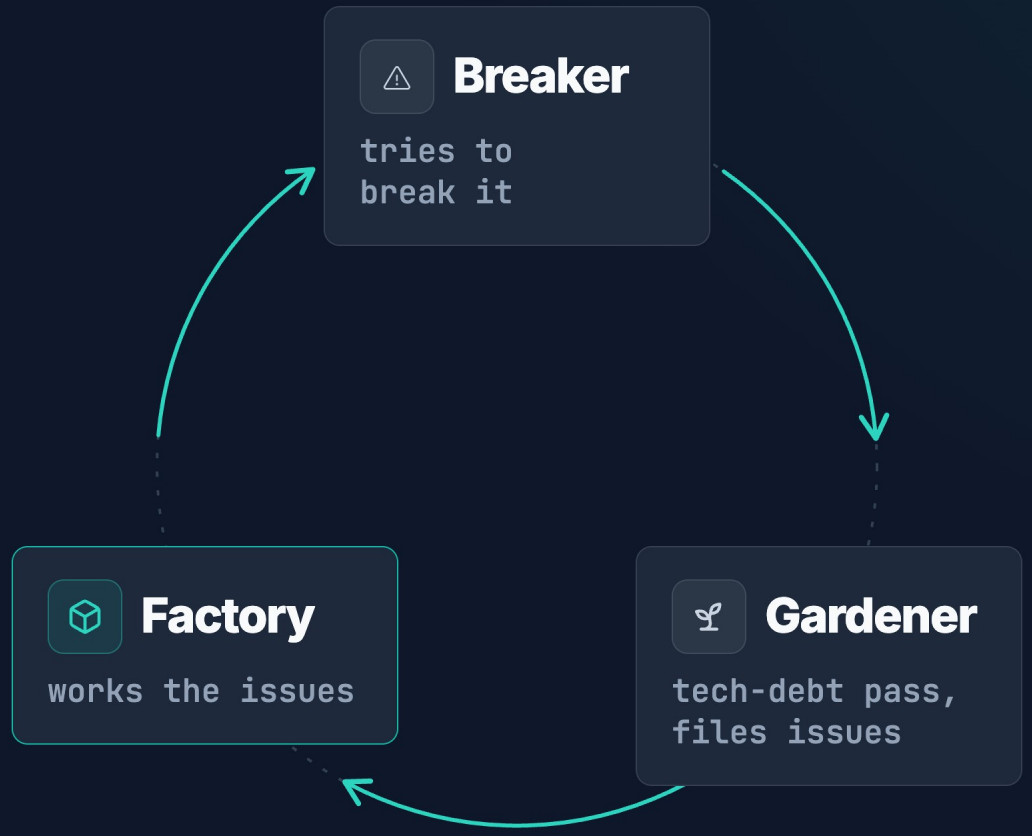
// DEMO · ROUGH TRUCK DERBY

Demo: a real game

Guardrails live in ADRs, and agents obey them

Godot · C# · Jolt physics · 71 ADRs

// DEMO · THE FACTORY BUILDS ITSELF



The factory builds itself

EXAMPLE

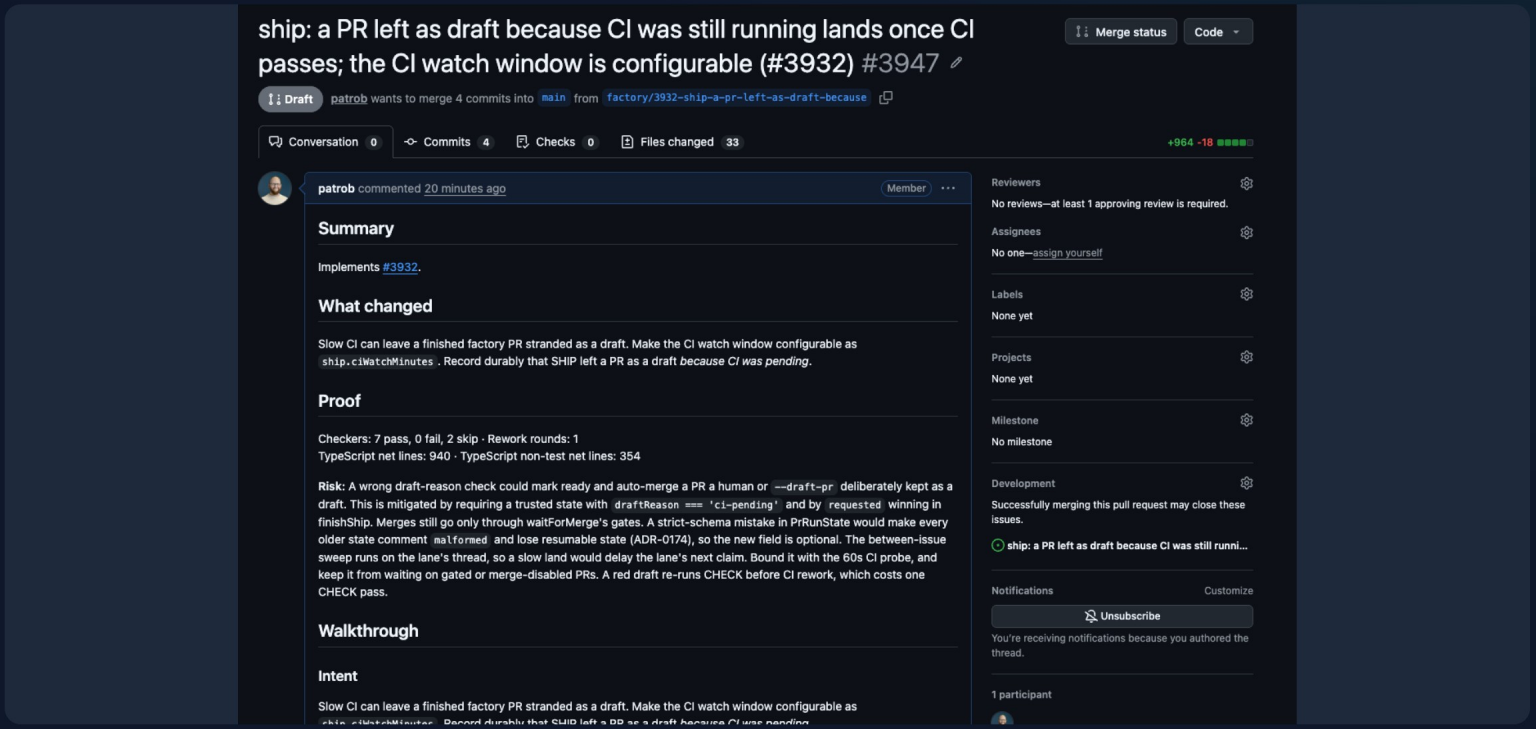
PR #2288

7 checks passed

1 rework round

// BACK TO THE RUN

Here's a run I started right before the talk



1. Plan
2. Checker results
3. Rework round
4. Evidence pack
5. CI green

// WHAT'S NEXT · FOREMAN

Foreman

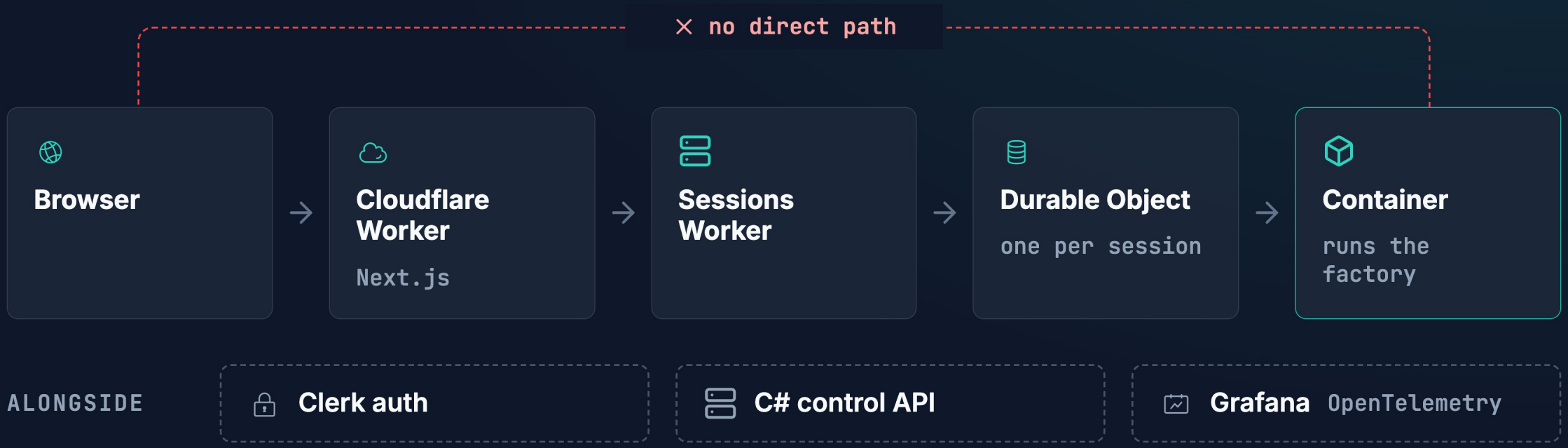
The factory, hosted. One ticket, one container.

foreman.onpardev.com

in progress

// FOREMAN ARCHITECTURE

One session, one container



The browser never talks to a container directly

// BRING YOUR OWN SUBSCRIPTION

Your Claude or Codex plan, my infrastructure



Credentials go into your container only and are never stored in plain text



Every run metered from day one

// MACHINES ENFORCE, NOT REVIEWERS

Every rule has a gate

- oxlint
- strict TS
- slice boundaries
- knip
- duplication
- max tokens/file
- no comments
- mutation testing
- preview deploy + smoke test

If a machine doesn't check it, it isn't a rule

// LESSONS

What I'd tell you before you build one

01 **Communication skills beat prompting skills**

02 **Small vertical slices beat big asks**

03 **Unchecked rules don't exist**

04 **Building the factory is more fun than shipping** (watch that)

// Q&A

Questions

github.com/on-par/software-factory

onpardev.com

patrickarobinson.com



scan for the repo