



# The GitOps Paradox

Why Your Devs Need an API You Don't Want to Build



## Who am I?

- Simon Koudijs
- ConfigButler

## Experience

- Software engineer / DevOps engineer

## Founder

- Quit my fulltime job to work on this fulltime
- Implementing this idear in the open
- ConfigButler



Will mine fly as  
well?

I ♥ Files

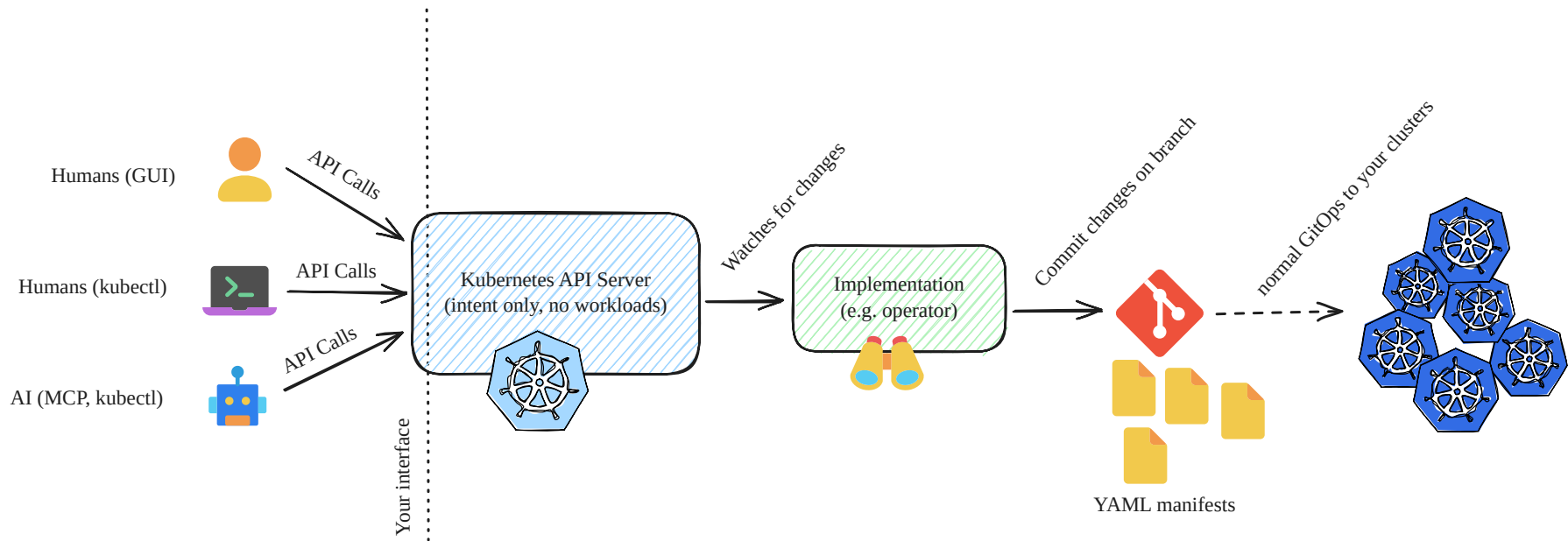
I ♥ Git

I ❤️ GitOps

Pfff now I need to  
create a PR with  
the right YAML 😞

# Why?

	GitOps	Reverse GitOps
Reproducible	✓	✓
Process	✓	✓
Full history	✓	✓
Rollbacks	✓	✓
API First 😊		✓
Fast type checking		✓



github-actions[bot] and sunib chore(main): release gitops-reverser 0.19.2 ✓		eb67c05 · last week	430 Commits
.devcontainer	chore: Get flux onboard to simplify the makefile and make it ...		last week
.github	chore: Fix linter and also run the bi-direction tests		last week
.vscode	fix: One edgecase where a secret is deleted (is now recreated)		2 weeks ago
api/v1alpha1	chore: refining age secret generation		last month
charts/gitops-reverser	chore(main): release gitops-reverser 0.19.2		last week
cmd	chore: First small step to HA		last week
config	chore: First small step to HA		last week
docs	docs: Improving docs		last week
hack	chore: Improving the test and keep forwards active		last week
internal	chore: Improving the test and keep forwards active		last week
test	chore: Lower wait times so that we can quickly validate it		last week
.dockerignore	feat: Use sops to encrypt secrets		last month
.gitignore	feat: Making big steps in Makefile understanding, but it's still ...		3 weeks ago
.golangci.yml	chore: Bump versions		last month
.release-please-manifest.json	chore(main): release gitops-reverser 0.19.2		last week
.yamllint.yml	chore: Refactoring away cyclomatic complexity		5 months ago
AGENTS.md	chore: e2e test for bi-directionality		last week
CHANGELOG.md	chore(main): release gitops-reverser 0.19.2		last week
CONTRIBUTING.md	docs: More info on deploying this thing		6 months ago
Dockerfile	chore: Get flux onboard to simplify the makefile and make it ...		last week

About ⚙️

Reconciles Kubernetes API resources into a git destination as clean deployable YAML files (audit trail through git commits).

- audit-log
- gitops
- config-as-data
- kubernetes-resource-model
- reverse-gitops

- Readme
- Apache-2.0 license
- Contributing
- Activity
- Custom properties
- 3 stars
- 0 watching
- 0 forks
- Audit log
- Report repository

Releases 30

gitops-reverser: v0.19.2 Latest  
last week

+ 29 releases

Packages 4

- gitops-reverser-ci
- gitops-reverser
- gitops-reverser-devcontainer
- charts/gitops-reverser

Contributors 3

# Voting demo

- A live voting app is always nice
- This one is truly cloud native
- The presentation as well



# Someone votes

The API receives a

`QuizSubmission`.

The synced repo turns that into a normal file.

```
apiVersion: examples.configbutler.ai/v1alpha1
kind: QuizSubmission
metadata:
  name: example
  namespace: vote
spec:
  answers:
    - questionId: q1
      singleChoice: Software engineer
    - questionId: q2
      singleChoice: Git
    - questionId: q3
      singleChoice: No
    - questionId: q4
      singleChoice: Curious
  sessionRef:
    kind: QuizSession
    name: demo
  submittedAt: "2026-02-05T12:34:00Z"
```

```
$ kubectl get quizsessions -n vote
NAME      STATE  TITLE
demo     live   KubeCon 2026 Demo
$ kubectl get quizsubmissions -n vote --watch
NAME      SESSION  AGE
example   demo     2m
$ git log --oneline --decorate --all -5
* 65b3c71 [CREATE] examples.configbutler.ai/v1alpha1/quizsubmissions/example
* e518c65 [CREATE] examples.configbutler.ai/v1alpha1/quizsessions/feedback
* fb999fd [CREATE] examples.configbutler.ai/v1alpha1/quizsessions/demo
```

Rendered output - on stage this was a live terminal

# Demo takeaway

Not workloads. Just editable intent objects in Git.

Plain object

QuizSession

Editable

with YAML

Plain object

QuizSubmission

Trackable

as files in Git



```
demo-repo/  
adam/rai/8f/  
  examples.configbutler.ai/  
    v1alpha1/  
      quizsessions/vote/  
        demo.yaml  
        feedback.yaml  
      quizsubmissions/vote/  
        example.yaml
```

PDF snapshot - on stage this was audited  
the API

# The paradox

- We shield people from the Kubernetes API.
- We force them to use Git as the write path.
- Then we build custom APIs and portals anyway.
- So why not use Kubernetes itself as the typed API for intent?
- Keep GitOps. Change the front door.



# reversegitops.dev



The image shows a dark-themed website for 'Reverse GitOps'. At the top, the title 'Reverse GitOps' is displayed in a light blue font, with a logo consisting of three squares and arrows. Below the title, the tagline 'API-first. Git-recorded. GitOps-compatible.' is centered. Two buttons are visible: 'Read the manifesto ↓' and '★ Star on GitHub'. A horizontal bar below the buttons contains a 'LIVE' indicator, the text 'I'm speaking at KubeCon Co-located 2026', the location 'Amsterdam', and the date 'March 23, 2026'. The main content area features a paragraph of text, a section header 'THE PROBLEM WITH GIT AS A FRONT DOOR', and another paragraph of text.

## Reverse GitOps

API-first. Git-recorded. GitOps-compatible.

[Read the manifesto ↓](#) [★ Star on GitHub](#)

**LIVE** I'm speaking at KubeCon Co-located 2026 @ Amsterdam March 23, 2026

*An API-first operating model where users and agents make validated changes through a typed API, while a controller writes those changes back into Git as clean declarative intent for audit, promotion, and distribution.*

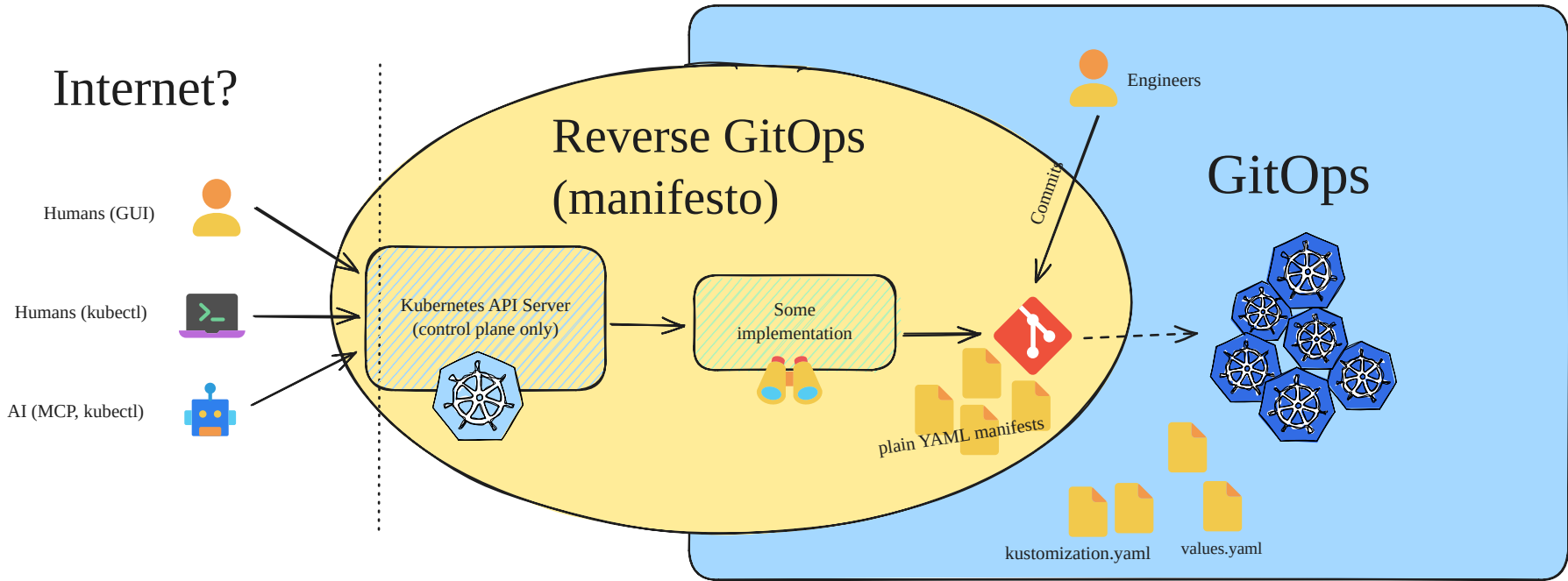
GitOps gave us reproducibility, auditability, and a shared language for intent. One of the best ideas the infrastructure community has produced.

But it carries an assumption that is starting to fail: that every user making a change should work directly through Git. Platform engineers are building self-service platforms. Developers and agents want a UI or a CLI backed by an API.

### THE PROBLEM WITH GIT AS A FRONT DOOR

Git was built for source code collaboration: line-oriented, text-based, and review-driven. These properties are virtues for code. They are friction for infrastructure operations. When Git is the front door, validation often happens after the fact: in CI, in controllers, or during deployment. Errors are discovered later, farther from the point of change.

Meanwhile, the Kubernetes API has spent ten years solving this problem with typed resources, validation at write time, and controllers built around structured state. Immediate feedback. A natural interface for humans and AI agents alike.



# 1. API First

- Use Kubernetes as the front door
- Give humans, GUIs, and AI one typed write path
- Keep it separate from your workload clusters
- Reuse auth, authz, CRDs, status, and controllers



## 2. Capture Intent, Not Implementation

- This is not API-only
- Keep resources declarative and reversible
- Store clean intent, not generated implementation details
- Think CRDs, Crossplane, KRO, and similar patterns



# KRO as example

```
apiVersion: kro.run/v1alpha1
kind: PodInfoApp
metadata:
  name: fiets
  namespace: podinfos-intent
spec:
  replicas: 1
  message: The Dutch word for bicycle is "fiets". We have a lot of them. 🚲
  color: "#115711"
  logoURL: https://demo.configbutler.ai/podinfo-assets/podinfos-fiets.jpg
```

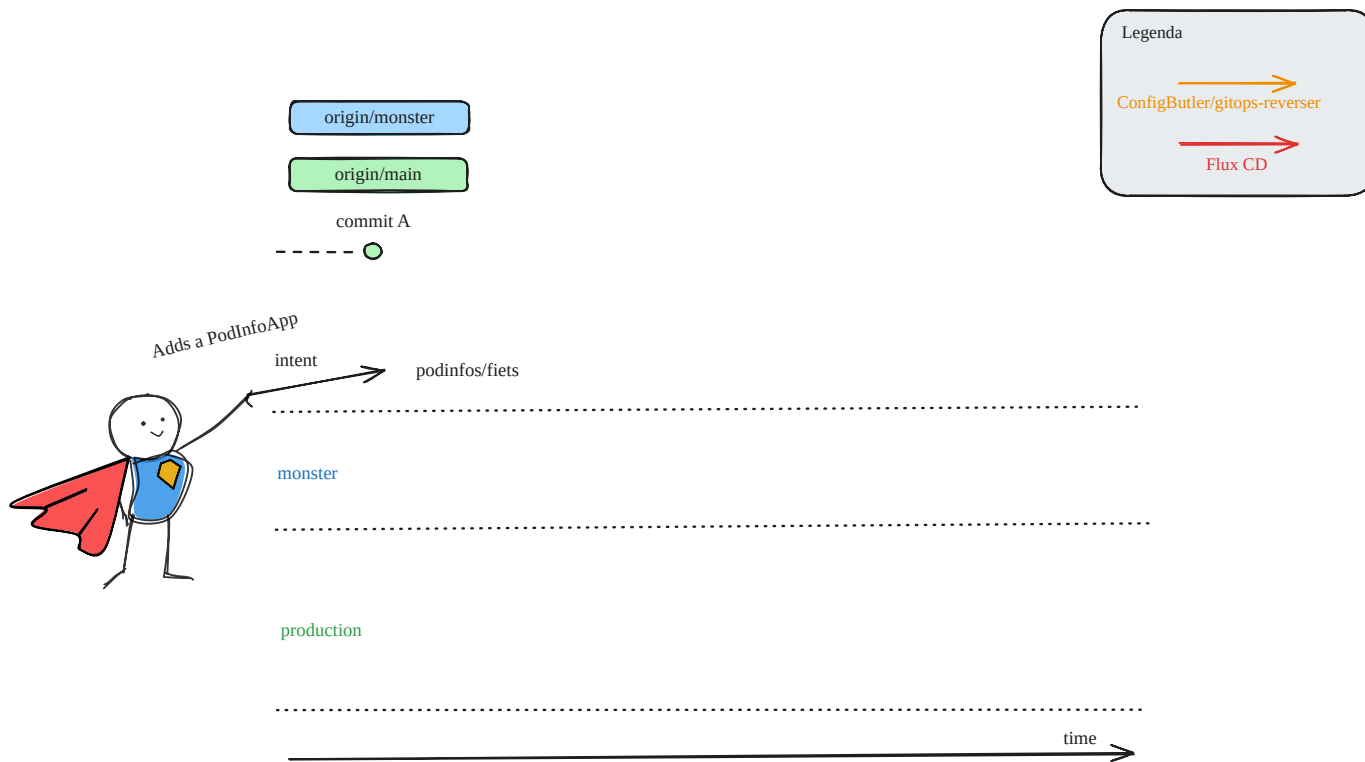
# 3. GitOps Applies

- The write path changes, not the deployment model
- Git still gives you review, history, rollback, and promotion
- Flux still does the delivery
- Branches become preview, `main` becomes production



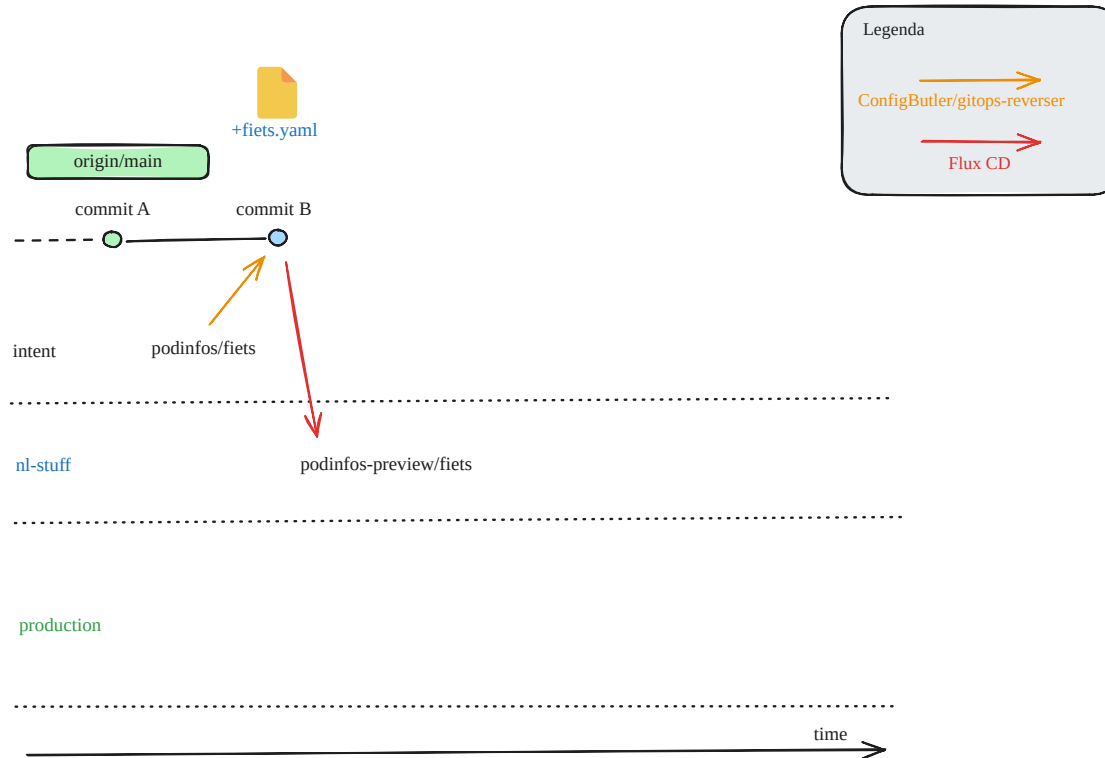
# A. User writes intent as YAML

Humans, GUIs, and AI can all use the same typed API.



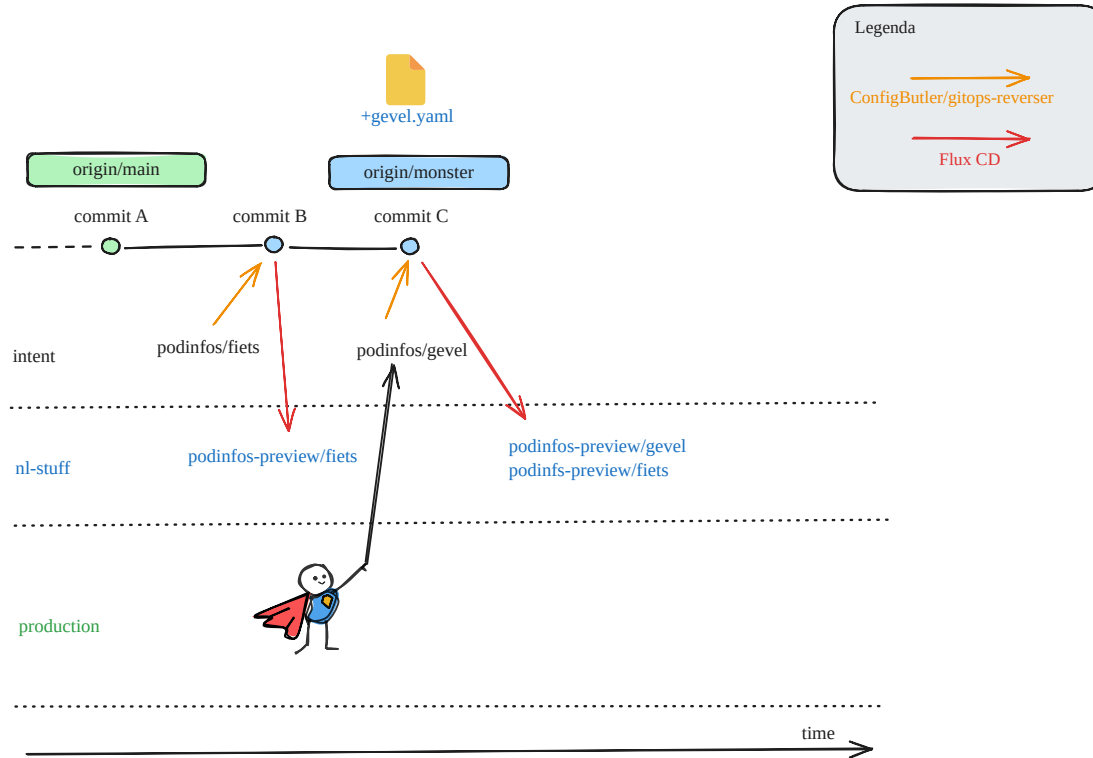
# B. gitops-reverser commits, Flux deploys preview

The branch exists for review, but the preview is already live.



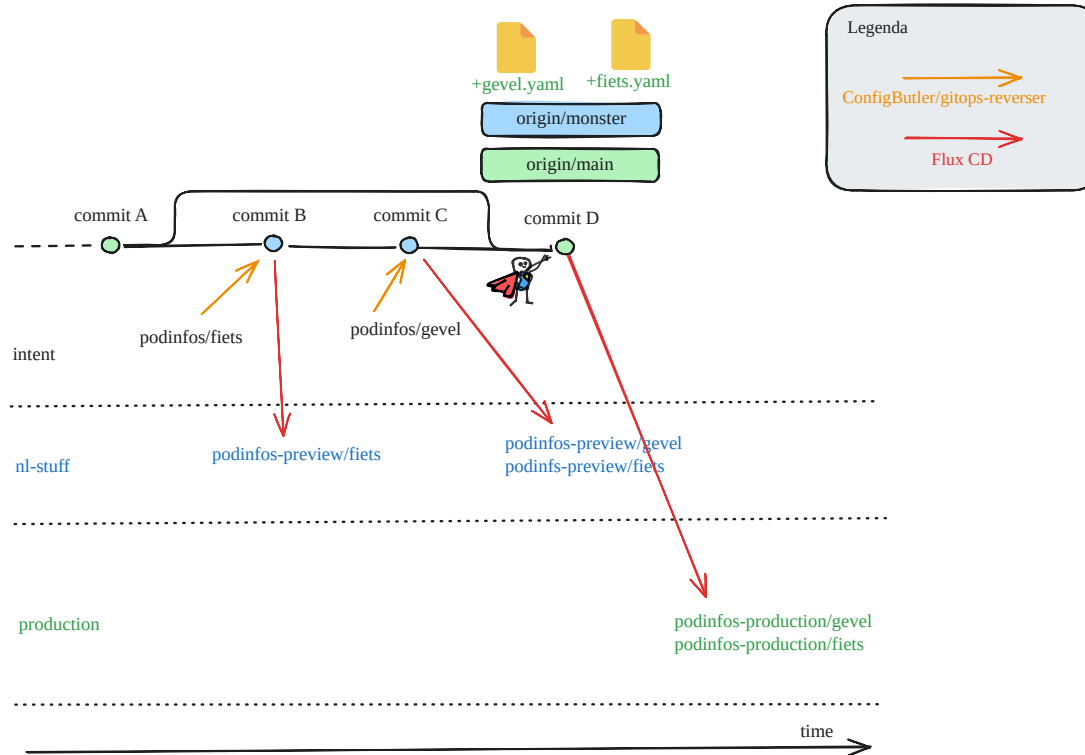
# C. Another change, another preview deploy

Each new YAML change becomes another commit and another preview update.



# D. Merge to `main`, Flux promotes to production

Promotion is still normal GitOps: review it, merge it, let Flux reconcile `main`.



# Key takeaways

- Reverse GitOps: API first, Git remembers
- Git and GitOps keep working as they do today
- Simple changes can happen via API, GUI, or `kubectl`

Questions or feedback: [koudijs.dev](https://koudijs.dev) or [LinkedIn](#)

More: [reversegitops.dev](https://reversegitops.dev) or the OSS prototype [ConfigButler/gitops-reverser](#)