Git push to deploy to your k8s cluster

GITKUBE

https://github.com/hasura/gitkube

Hi!

Tanmai Gopal

Co-founder, hasura.io

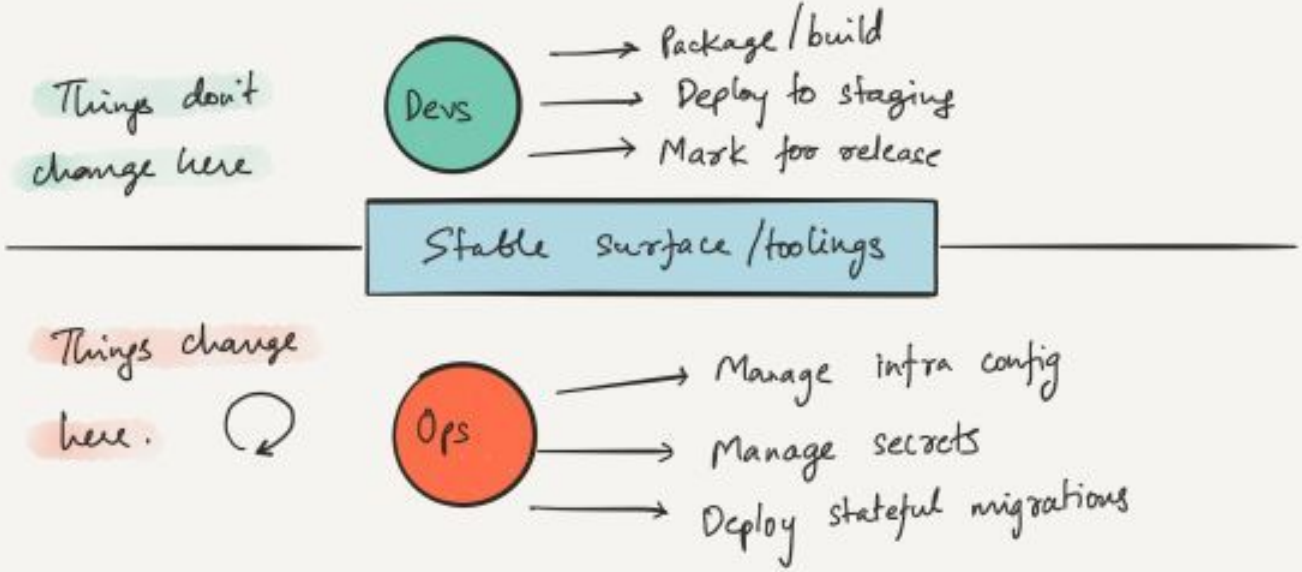
@tanmaigo

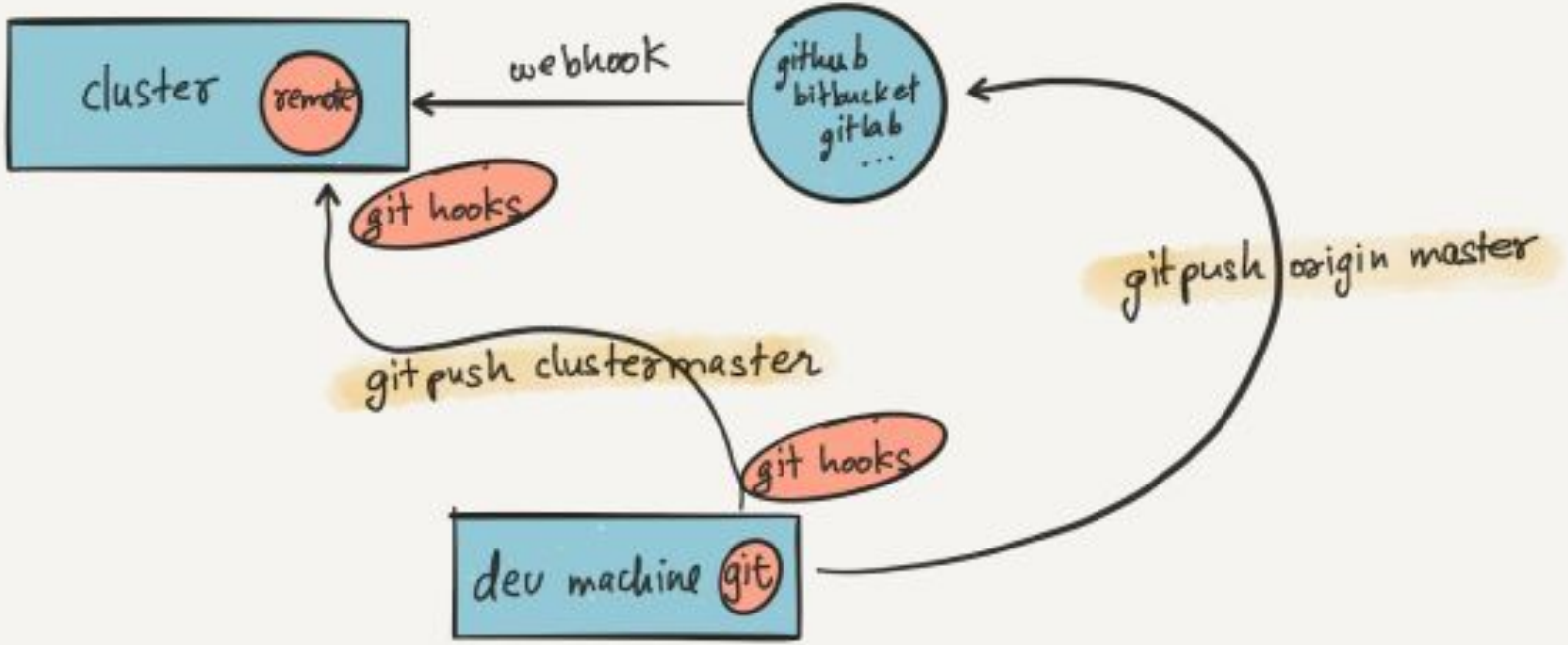
What is a git-push workflow

- Write code
- `git push heroku master` to deploy

Changed the world for developers, because it used dev only tools (git). Reduced *unnecessary* abstractions.

kubectl + git. Everything can be built around these 2 systems. Build your own git-push workflows. The main goal is to simplify devops and “pipelines”.


Things don't change here
Devs
Package / build
Deploy to staging
Mark for release
Stable surface / toolings
Things change here.
Ops
Manage infra config
Manage secrets
Deploy stateful migrations


cluster
remote
webhook
github
bitbucket
gitlab
...
git hooks
git push origin master
git push cluster master
git hooks
dev machine
git

The simplest DevOps task: Build & deploy

I have source code on my machine. I can run it locally.

I want to deploy my source code at current commit.

When git push:

- Build: Dockerfile tagged with commit
- Deploy: Apply changes to kubernetes deployment with new image tag

Before <> after

`$ docker build -t registry.com/my-image:my-tag`	
`$ docker push registry.com/my-image:my-tag`	`$ git push dev master`
`$ kubectl set image deployment/my-deployment container=registry.com/my-image:my-tag`	

Git hooks for “git push”

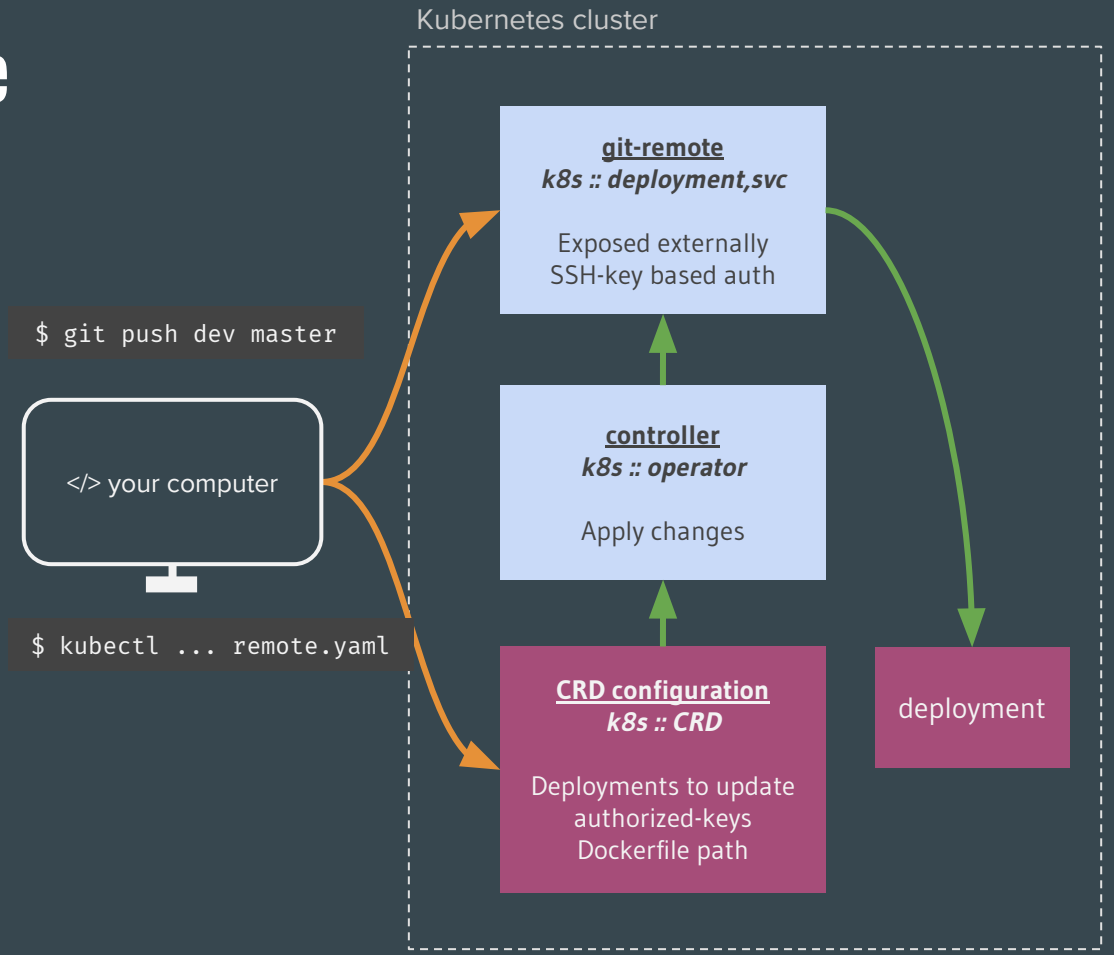
client-side	
> pre-push	Exit can abort push
server-side	
> pre-receive	Exit can abort push
> post-receive	Cannot abort push

Executable script in `.git/hooks/` named <hook>

`.git/hooks/pre-receive`

The obvious architecture

- Git remote agent on your cluster. Pre-receive hook:
  - Build docker image
  - Apply to k8s deployment
- Configuration:
  - SSH keys
  - Dockerfile path
- Sync configuration changes with the git-remote agent



Demos

But all based on just one idea

git push to apply changes to kubernetes objects.

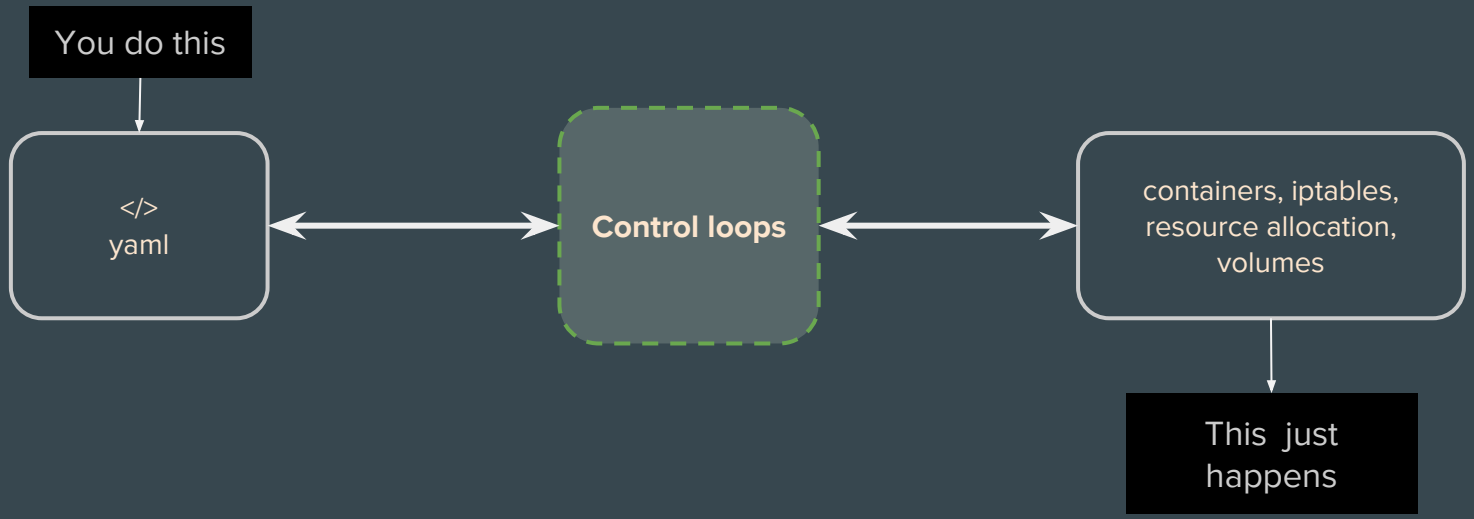
This way, git just works for AllTheThings™

```
git checkout <commit>

git push dev my-branch:master
```

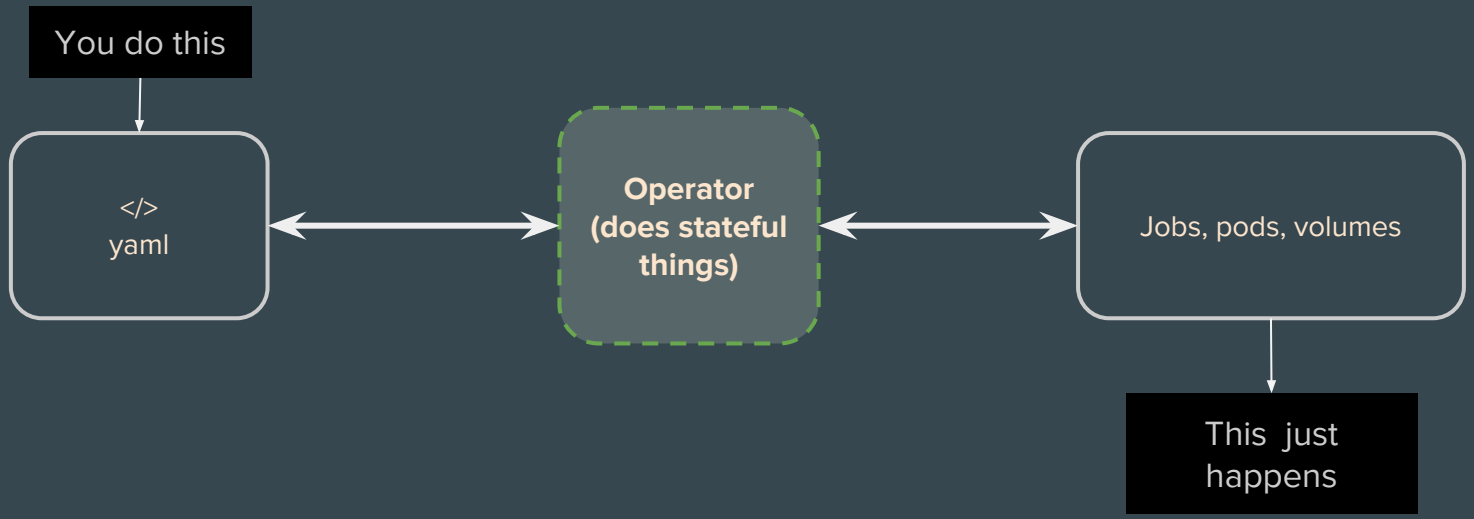
Kubernetes controller

The most awesome thing about how kubernetes works:

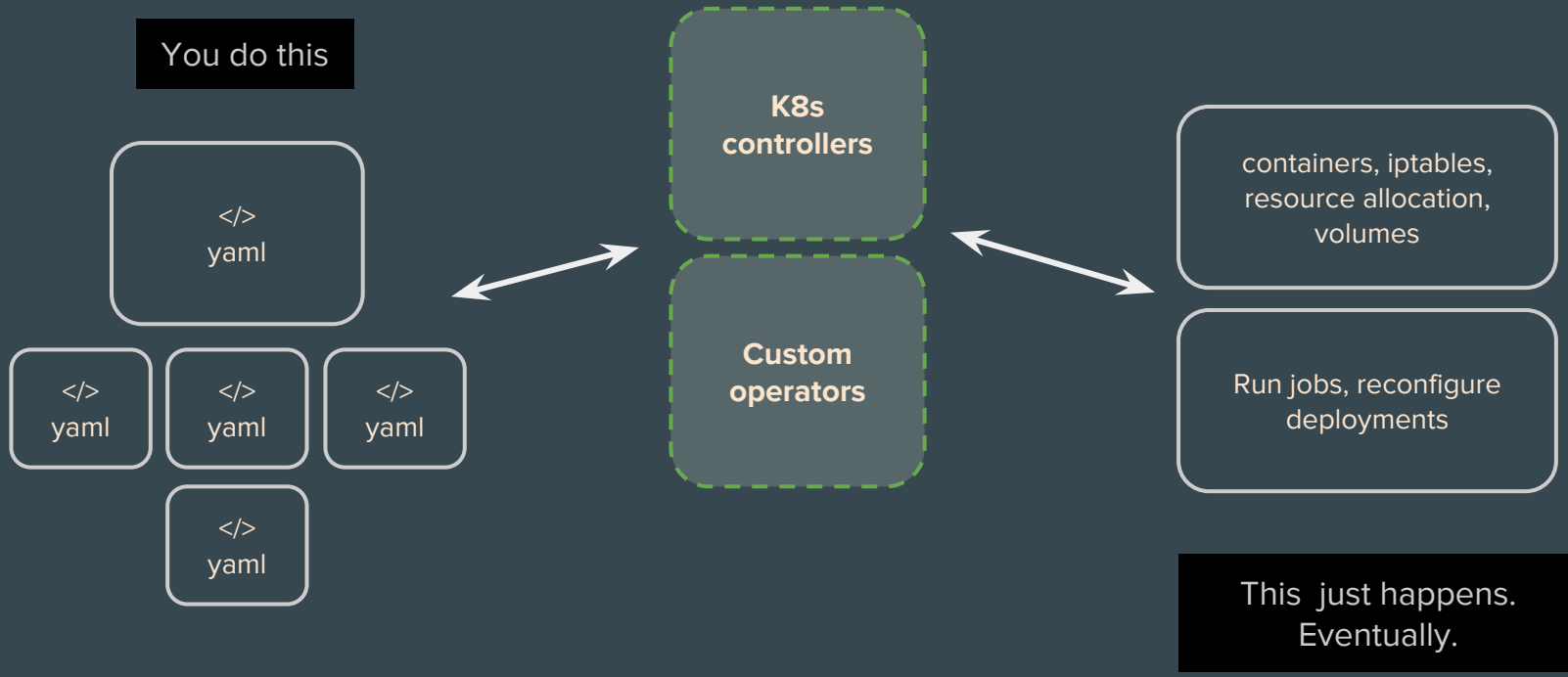


And everything is moving in this direction. Eg: The CRD + operator pattern

CRD + operator pattern



Build your own operators



Things that are hard with GitOps

- Secrets
  - Must be applied without committing to git
    - *pre-push hook?*
- Templating
  - Helm
  - Kubernetes native templating
- Releases, canary deployments?
  - GitOps with istio :)

@tanmaigo

@HasuraHQ