

GOTO **AARHUS 2021**

#GOTOaar

TRIFORK[®]

SAILING THE CLOUD NATIVE SEA WITH K3S

Lowering the barrier to entry for Kubernetes



Agenda

- Prolog(ue)
- Act one: docker-compose up && echo done
- Act two: trouble in paradise
- Act three: is there something better and can we afford it?
- Act four: `curl -sfL https://get.k3s.io | sh -`
- Epilogue



Prolog(ue)

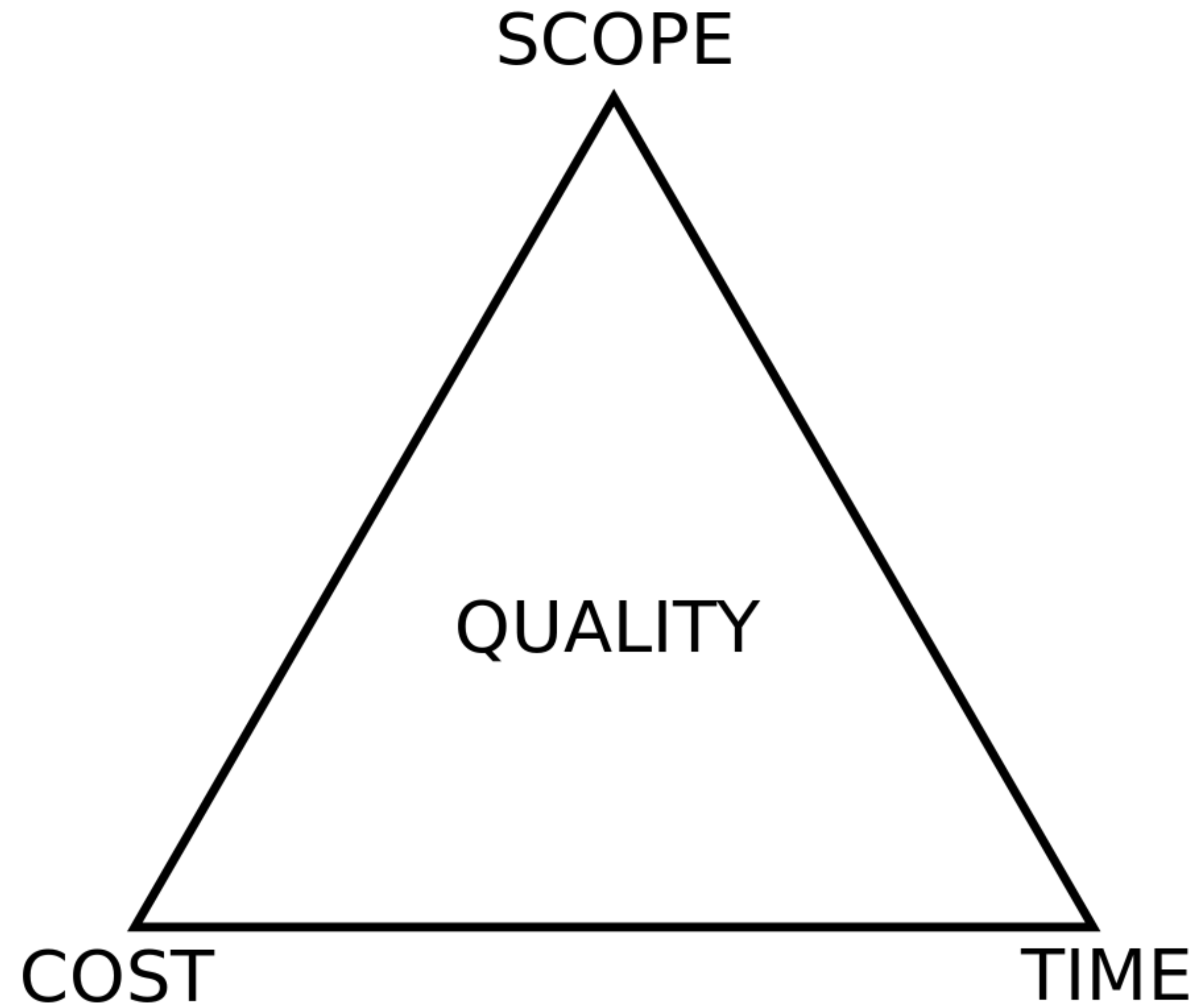


Act one: docker-compose

- Small team
- Small budget
- Short deadline
- Proof of concept
- On premises (in DK)
- Limited Kubernetes Knowledge

Act one: docker-compose

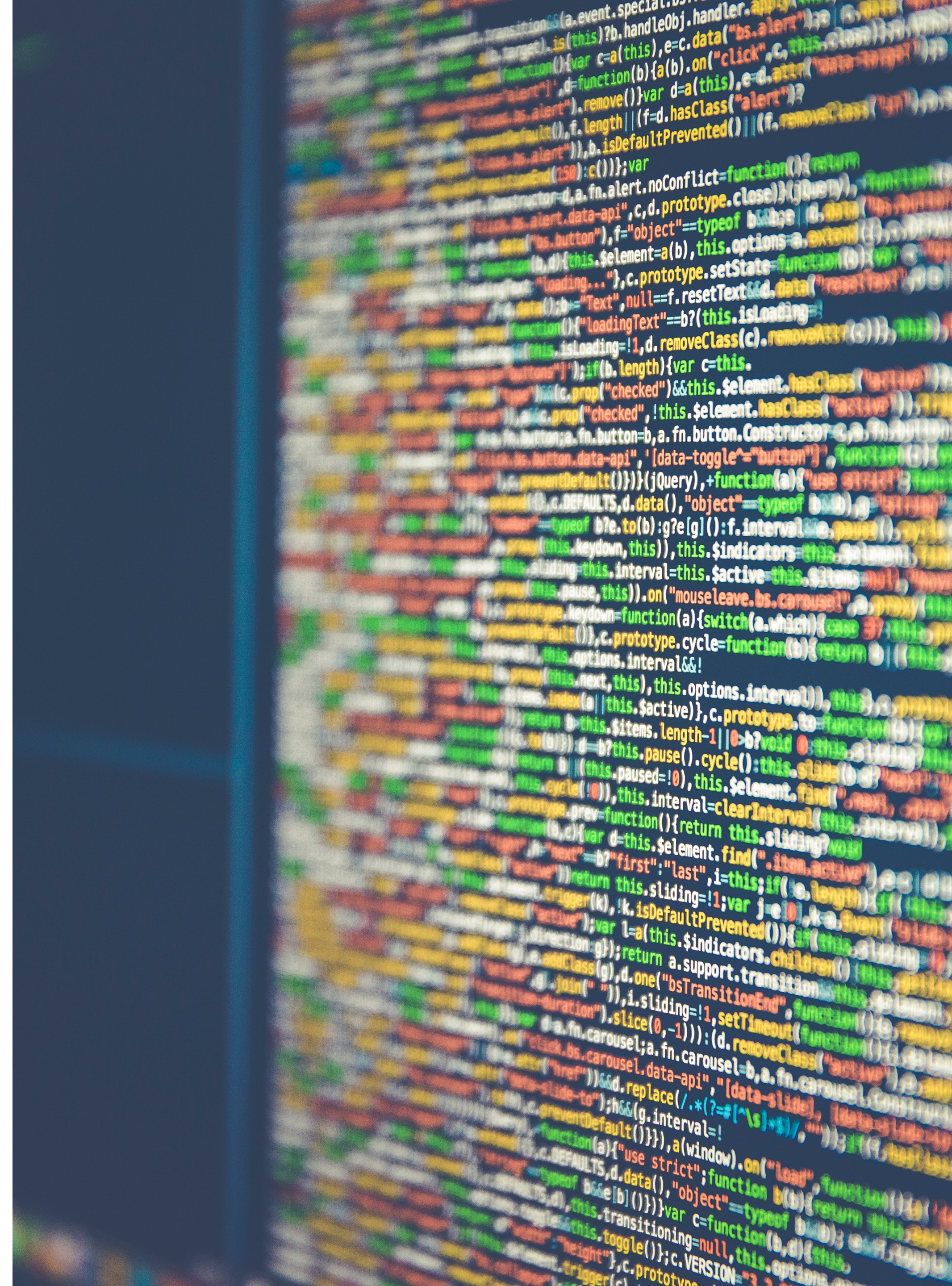
Goals	Constraints
Scalability	Budget
Simplicity	Limited experience with Kubernetes
Time to market	On-premises

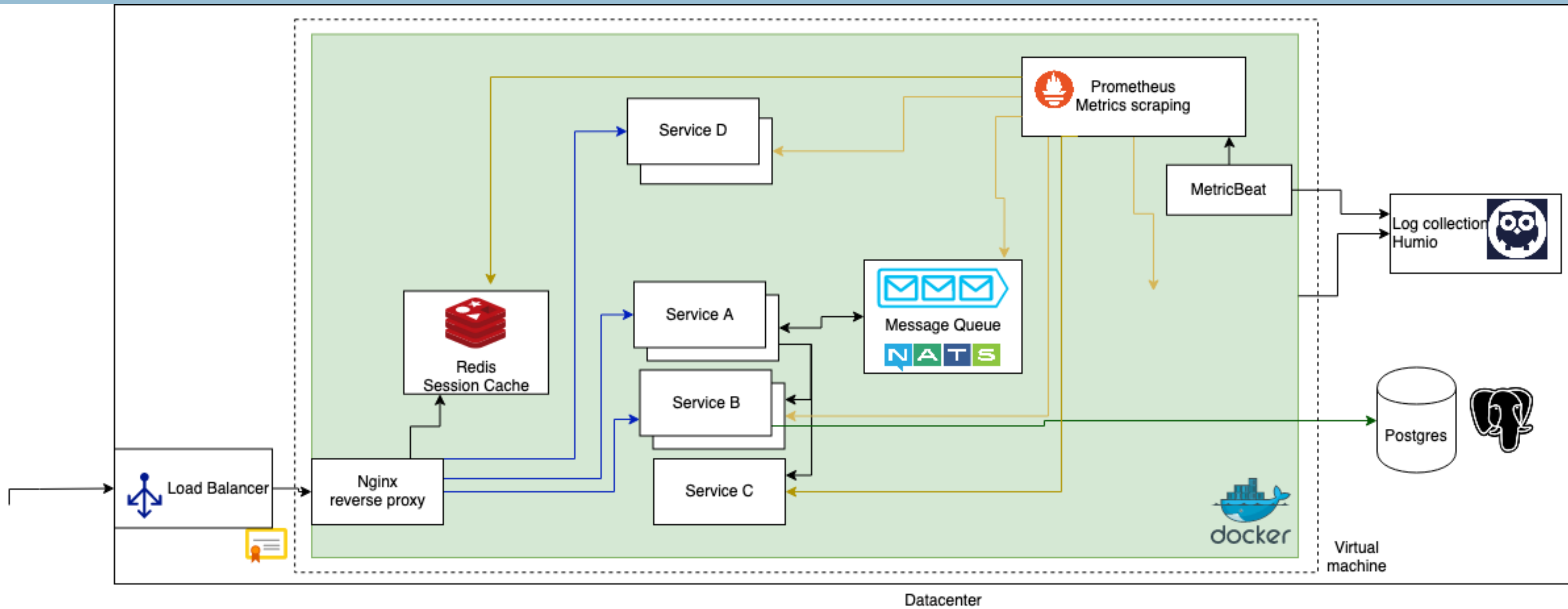




kubernetes

- Doesn't always make sense
 - Cost
 - Size of project
 - Organisational complexity
 - Difficulty in getting buy-in





docker-compose up

- Staging / Production split on two VMs
- 2 cores and 4 Gb RAM
- **Desired state:** kept in git
- **Secrets:** encrypted partition mounted as volumes
- **Deployment:** ssh + “docker-compose up”
- Good UX for developers

```
version: "3.9" # optional since v1.27.0
services:
  web:
    build: .
    ports:
      - "5000:5000"
    volumes:
      - .:/code
      - logvolume01:/var/log
    links:
      - redis
  redis:
    image: redis
volumes:
  logvolume01: {}
```


Act two: trouble in paradise

Act two:

- More covid => more users => need better observability
 - Configuration of off-the-shelf tools was a pain
- Deployment took a lot of manual work
- No rolling upgrades
 - Couldn't deploy during business hours
- Secrets managed manually and separate from services


```
<source>
  @type forward
  port 24224
  bind 0.0.0.0
</source>

<match **>
  @type          elasticsearch
  host           "#{ENV['HUMIO_HOST']}"
  port           "#{ENV['HUMIO_PORT']}"
  scheme         "#{ENV['HUMIO_PROTOCOL']}"
  user           "#{ENV['HUMIO_REPO']}" # Replace with your Humio repo
  password       "#{ENV['HUMIO_TOKEN']}" # Replace with your actual ingest token
  logstash_format true
  verify_es_version_at_startup false
  <buffer>
    flush_mode interval
    flush_interval 10s
    retry_type periodic
    retry_wait 10s
    retry_timeout 10d
    total_limit_size 500MB
    chunk_limit_size 8MB
    overflow_action drop_oldest_chunk
  </buffer>
</match>
```


🔗 master ▼


🔗 1 branch

🏷 114 tags

Go to file

Add file ▼

📄 Code ▼

	nightah Update grafana to v8.0.0	87f6747 2 days ago	🕒 296 commits
📁 alertmanager	add Slack user to config, fix repo rename	5 years ago	
📁 caddy	Update Caddy to v2 (#216)	3 months ago	
📁 grafana/provisioning	Update grafana to v6.5.0-beta1 and utilise provisioning system	2 years ago	
📁 helpers/aws	simple EC2/ECS helpers for dockprom	5 years ago	
📁 prometheus	Update dashboards, alerts and README to reflect changes in node-e...	3 years ago	
📁 screens	Update screenshots	2 years ago	
📄 .gitattributes	Stop git handling binary images files as text	2 years ago	
📄 .gitignore	ignore idea	4 years ago	
📄 LICENSE	Initial commit	5 years ago	
📄 README.md	Update code format in README.md (#221)	3 days ago	
📄 config	added config file for credentials and updated README.md how to use...	3 years ago	
📄 docker-compose.exporters.yml	Update cadvisor to v0.39.2	3 days ago	
📄 docker-compose.yml	Update grafana to v8.0.0	2 days ago	

☰ README.md

dockprom

A monitoring solution for Docker hosts and containers with [Prometheus](#), [Grafana](#), [cAdvisor](#), [NodeExporter](#) and alerting with [AlertManager](#).

If you're looking for the Docker Swarm version please go to [stefanprodan/swarmprom](#)

Install

Clone this repository on your Docker host, cd into dockprom directory and run compose up:

```
git clone https://github.com/stefanprodan/dockprom
cd dockprom

ADMIN_USER=admin ADMIN_PASSWORD=admin ADMIN_PASSWORD_HASH=JDJhJDE0JE91S1FrN0Z0VEsyWmhrQVpON1VzdHVLSDI
```

Caddy v2 does not accept plaintext passwords. It MUST be provided as a hash value. The above password hash corresponds to ADMIN_PASSWORD 'admin'. To know how to generate hash password, refer [Updating Caddy to v2](#)

Prerequisites:

- Docker Engine >= 1.13
- Docker Compose >= 1.11

Updating Caddy to v2

Perform a `docker run --rm caddy caddy hash-password --plaintext 'ADMIN_PASSWORD'` in order to generate a hash for your new password. ENSURE that you replace `ADMIN_PASSWORD` with new plain text password and

About

Docker hosts and containers monitoring with Prometheus, Grafana, cAdvisor, NodeExporter and AlertManager

docker

monitoring

grafana

prometheus

cadvisor

alertmanager

📖 Readme

📄 MIT License

Releases 114

📦 v4.0.0

Latest

2 days ago

+ 113 releases

Packages

No packages published

Contributors 17





+ 6 contributors



Problems to solve

- Simplify deployment
- Secrets management
- Configuration of common tools

**Act three: is there
something better and can
we afford it?**

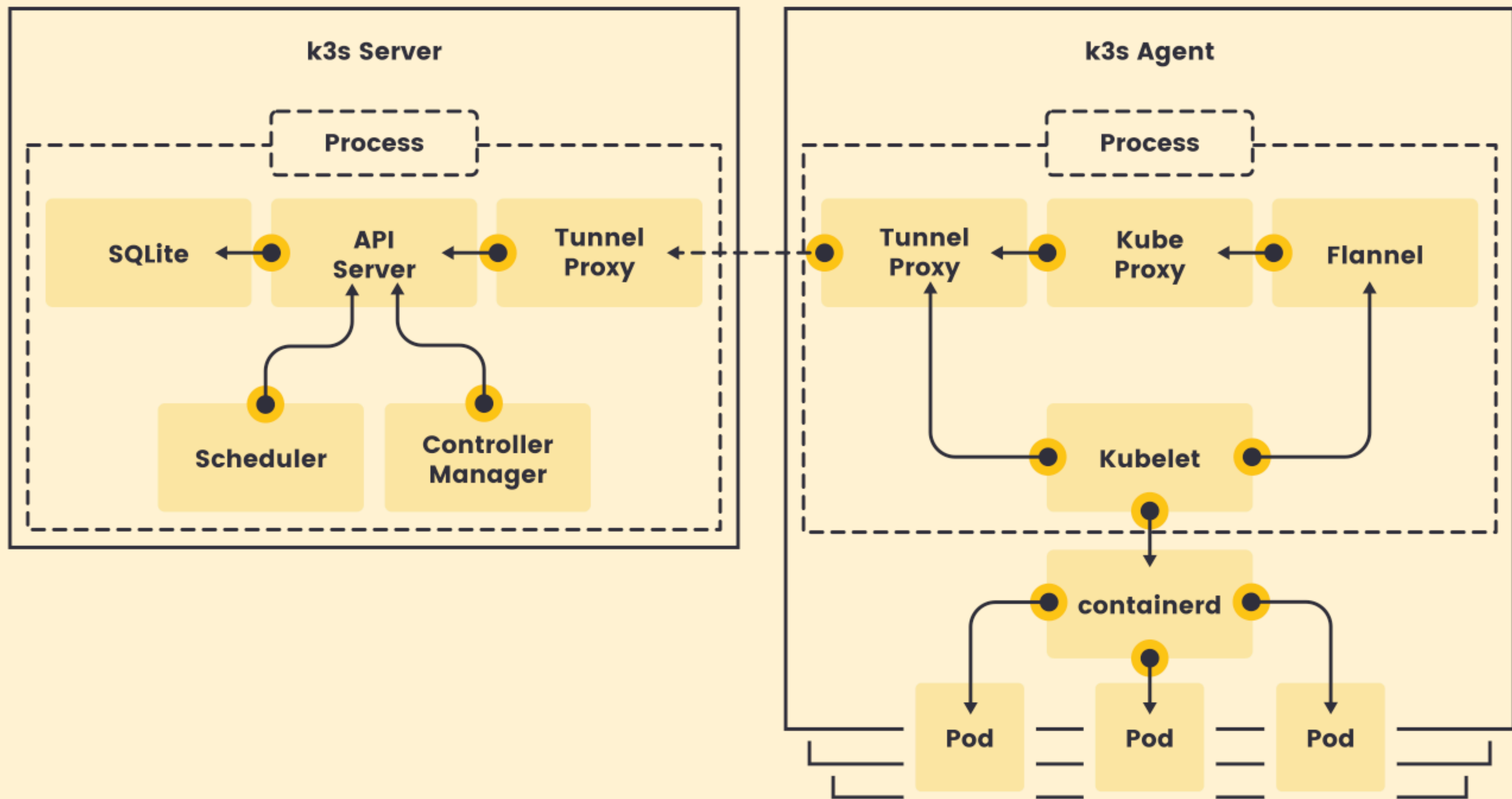


K3S

- Certified Kubernetes distribution
- Originally by Rancher
- Now CNCF Sandbox project
- Embedded SQLite instead of etcd
- Single binary
- So small that you can run it on a Raspberry PI



- Super simple setup
- Generates initrc / systemd scripts
- Runs containerd under the hood instead of Docker
- Comes with Traefik as ingress controller



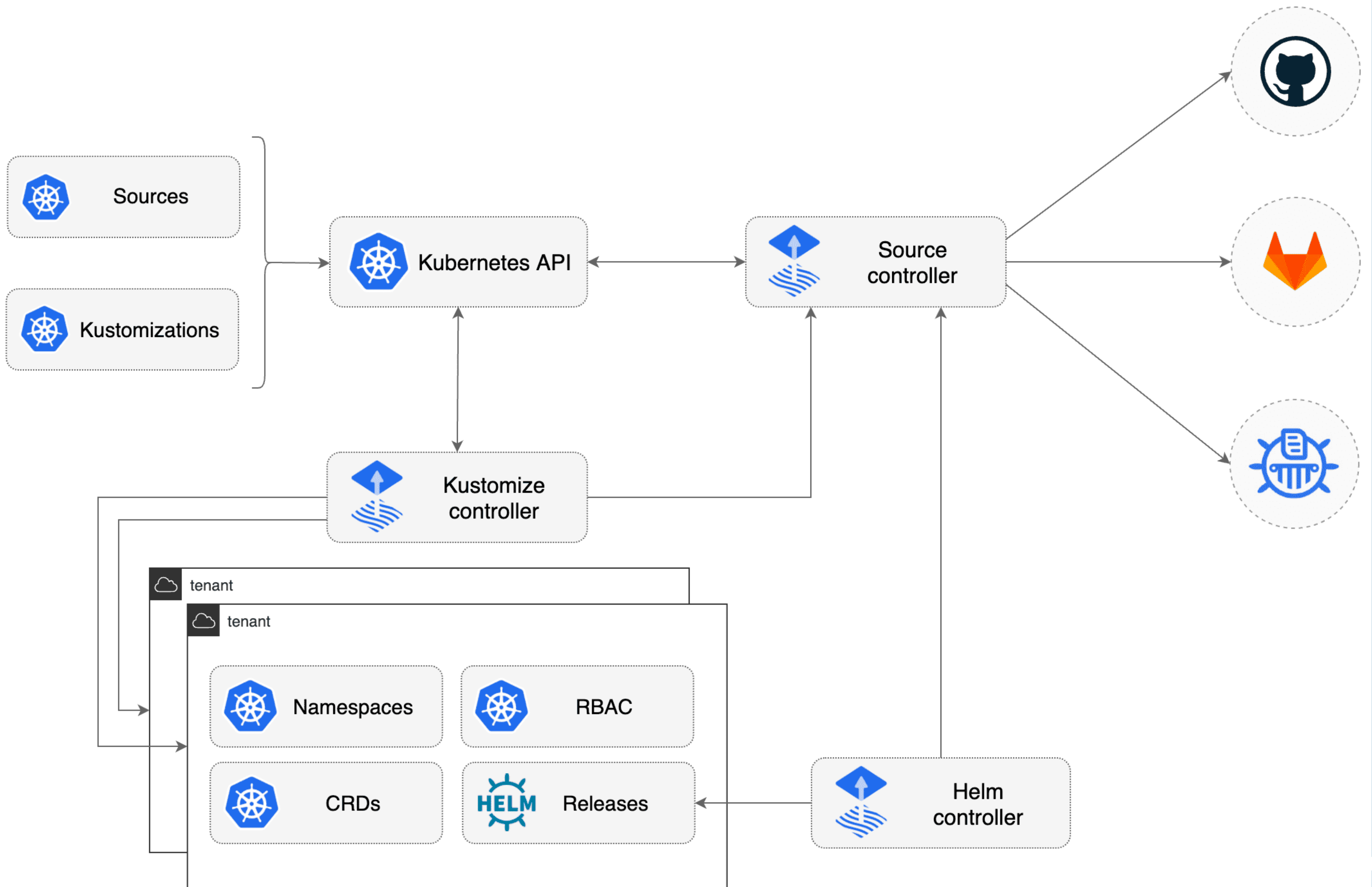


Trade-offs of single-node clusters

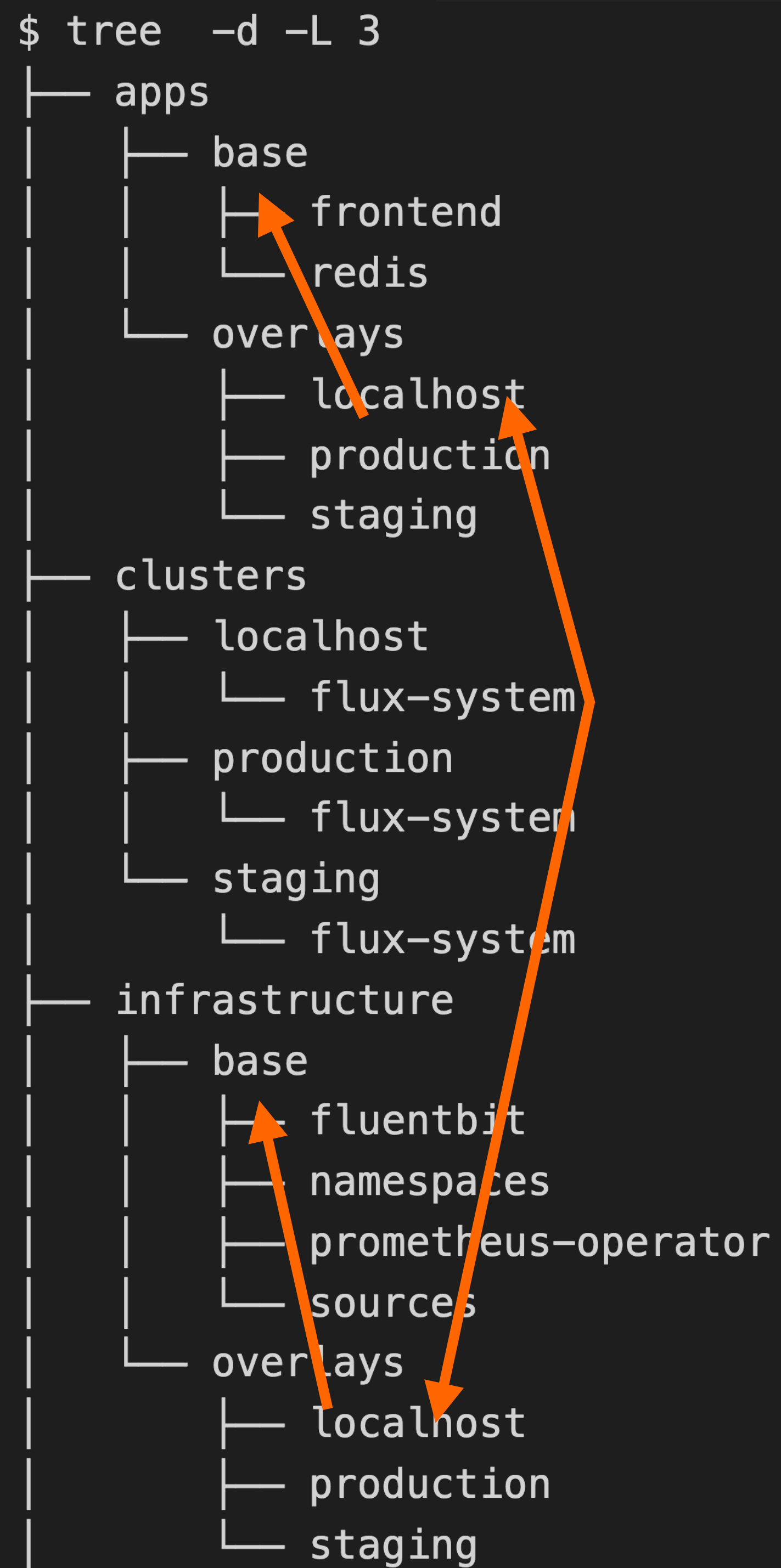
- ✗ Highly available
- ✗ Zero-down time cluster upgrades
- ✗ Cluster auto-scaling



- GitOps tool originally by WeaveWorks
- Now a CNCF incubation project
- Built-in support for Helm, Kustomize
- Good monitoring + alerting options
- Highly recommend:
<https://github.com/fluxcd/flux2-kustomize-helm-example>
<https://github.com/fluxcd/flux2-multi-tenancy>




```
$ tree -d -L 3
├── apps
│   ├── base
│   │   ├── frontend
│   │   └── redis
│   └── overlays
│       ├── localhost
│       ├── production
│       └── staging
├── clusters
│   ├── localhost
│   │   └── flux-system
│   ├── production
│   │   └── flux-system
│   └── staging
│       └── flux-system
├── infrastructure
│   ├── base
│   │   ├── fluentbit
│   │   ├── namespaces
│   │   ├── prometheus-operator
│   │   └── sources
│   └── overlays
│       ├── localhost
│       ├── production
│       └── staging
```



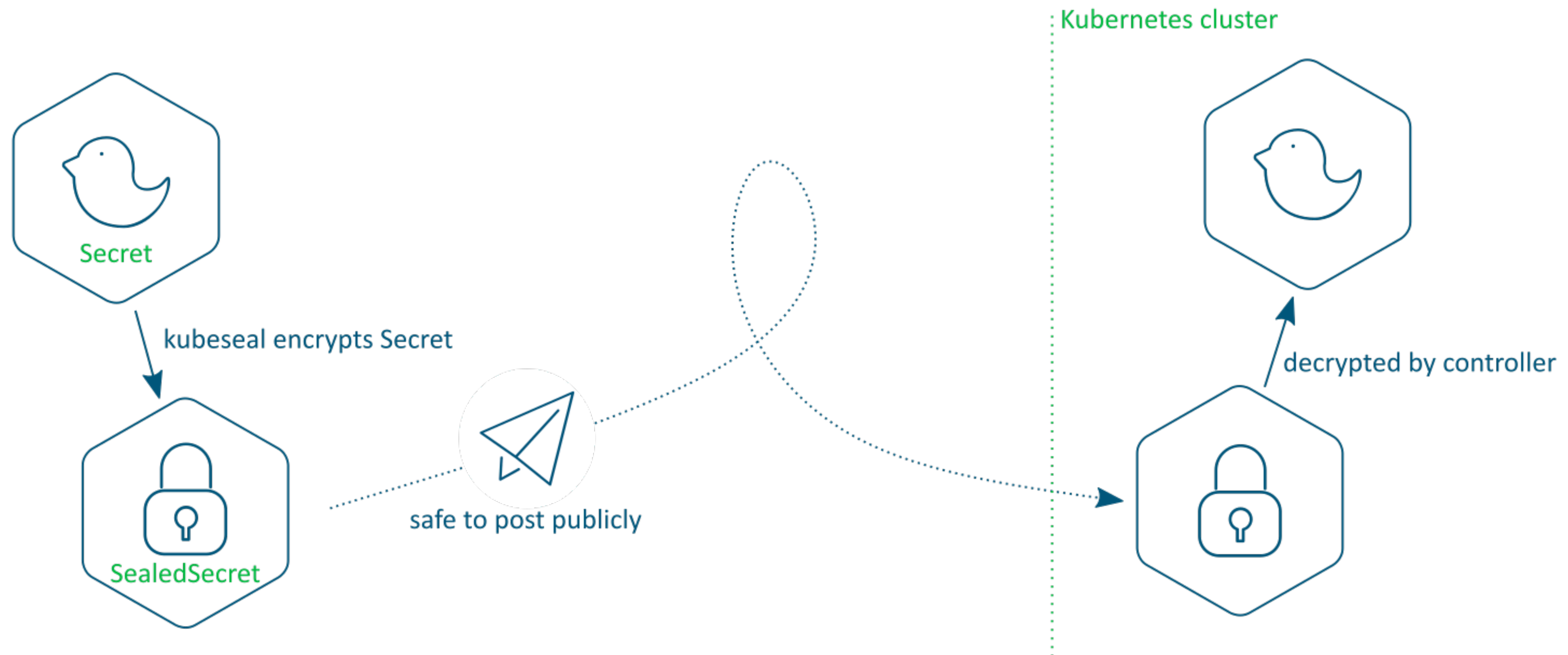
The diagram illustrates a directory tree structure with three main branches: `apps`, `clusters`, and `infrastructure`. Each branch contains subdirectories for different environments: `base`, `overlays`, and `localhost`, `production`, and `staging`. Orange arrows highlight specific paths: one from `apps` to `frontend`, another from `apps` to `localhost`, and a third from `infrastructure` to `localhost`.

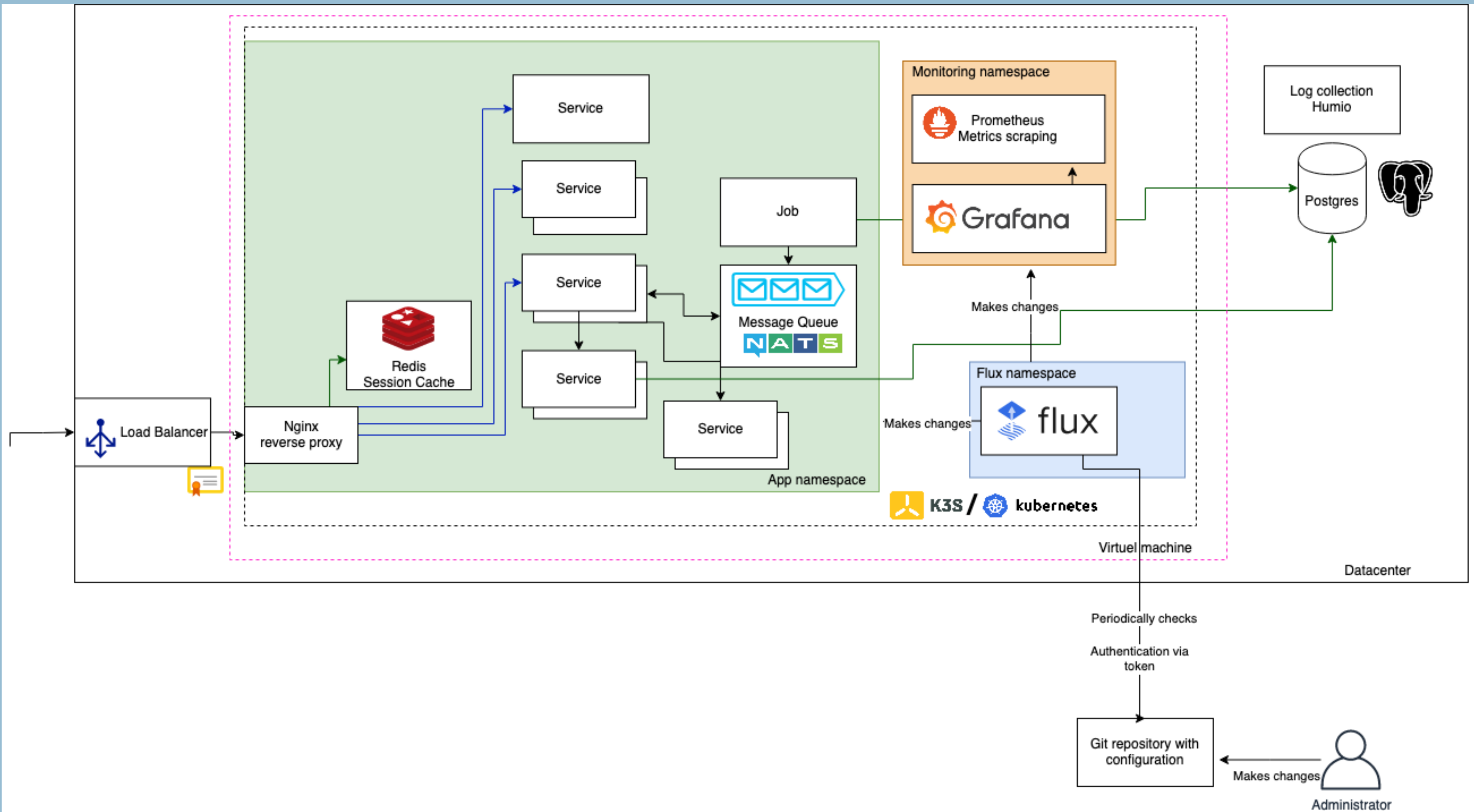
Sealed Secrets

- “One-way” encrypted secrets
 - public / private key
- Built by Bitnami
- Encrypted secrets committed to git
- Automatically unencrypted inside Kubernetes

Sealed Secrets

Life of a SealedSecret





Epilogue

Where to go from here?

- More clusters (dev/staging/prod)
- Upgrade the VM instance
- Multi-node K3s
- Managed Kubernetes
- Everything's in git - easy to move

Tools in the toolbox

- Single sign-on for monitoring tools:
 - Dex + OAuth2 Proxy + GitHub as IdP
- Automatic Docker image upgrades based on semantic versioning
- Developer experience:
 - K3d (K3s in Docker)
 - Live-reloading using Tilt
- AlertManager
- Notifications in Slack
- Validate manifests using GitHub actions on Pull Requests
- Webhooks to show deployment status of commits in GitHub

Don't forget to
vote for this session
in the **GOTO Guide app**