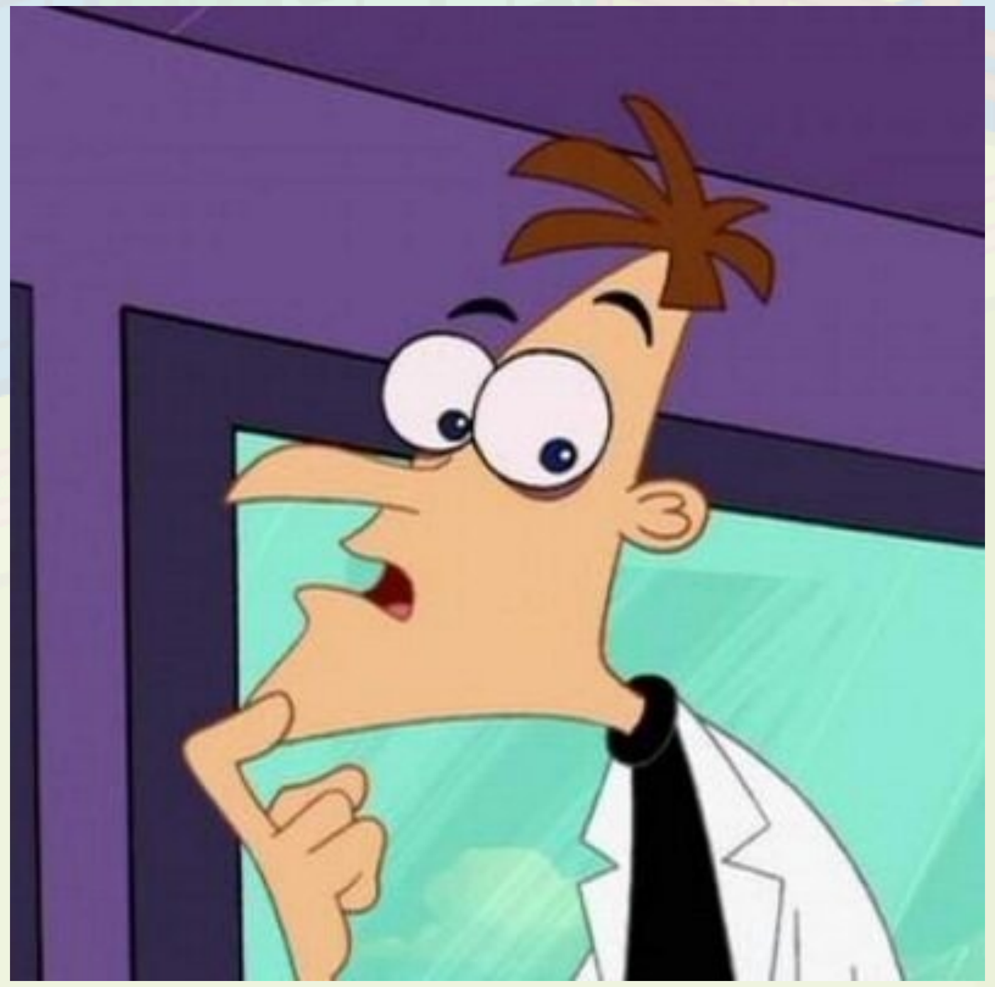
Introducing k3s - a Lightweight Kubernetes Distribution Built for the Edge

RANCHER®

#k3s • February 25, 2019

Shannon Williams

Co-Founder/VP Sales & Marketing
Rancher Labs
@smw355



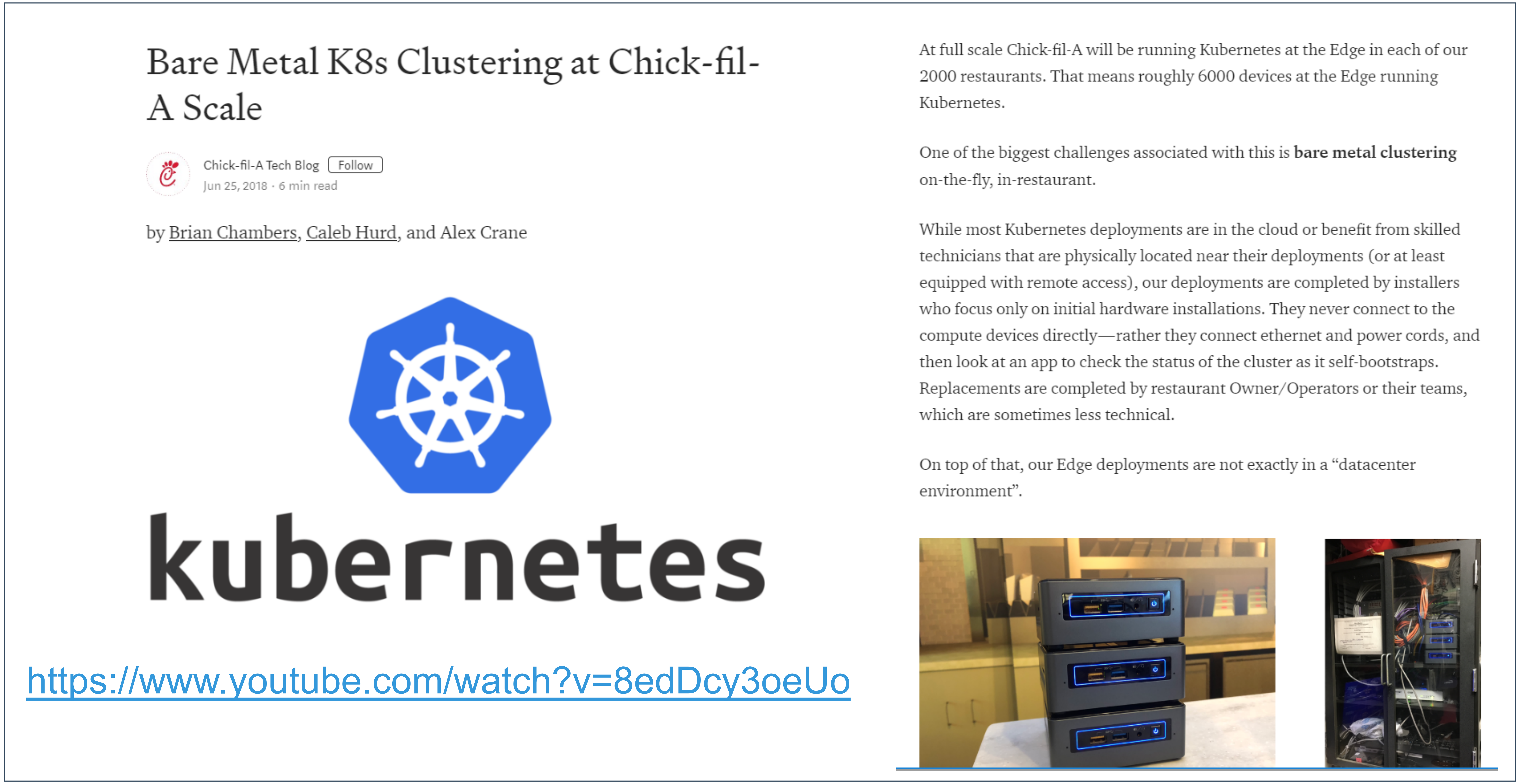
Darren Shepherd

Co-Founder/Chief Architect
Rancher Labs
@ibuildthecloud

Agenda

1. Increasing demand for an edge-optimized Kubernetes
2. Introducing k3s – a micro-distribution of Kubernetes
3. How we cut down the size and runtime-foot print of Kubernetes
4. Adding SQLite as an optional data store
5. Demo
6. Managing clusters on the edge

In the last year we’ve seen a major increase in the demand for Kubernetes outside the datacenter



Bare Metal K8s Clustering at Chick-fil-A Scale

Chick-fil-A Tech Blog Follow

Jun 25, 2018 · 6 min read

by Brian Chambers, Caleb Hurd, and Alex Crane

At full scale Chick-fil-A will be running Kubernetes at the Edge in each of our 2000 restaurants. That means roughly 6000 devices at the Edge running Kubernetes.

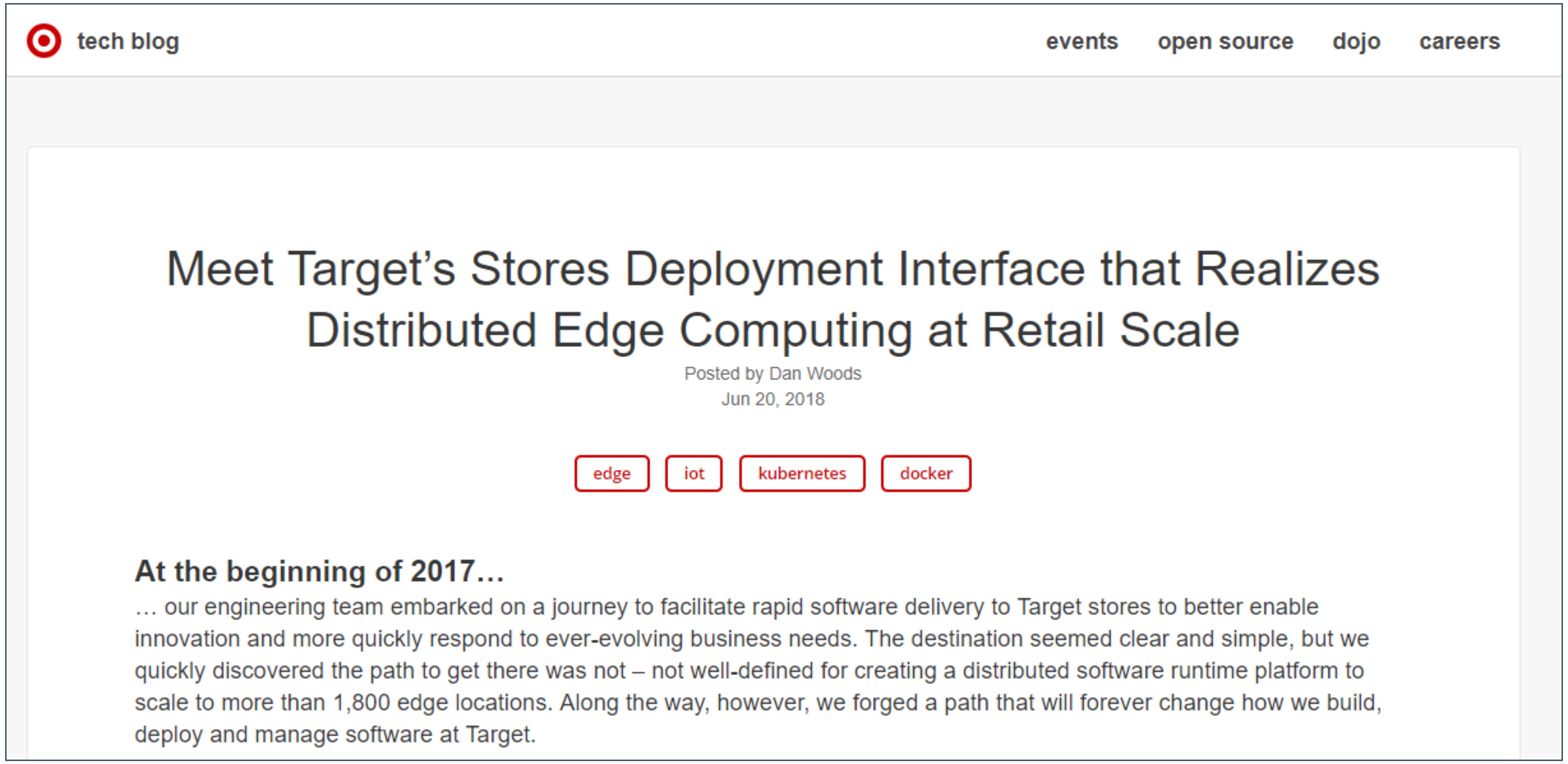
One of the biggest challenges associated with this is **bare metal clustering** on-the-fly, in-restaurant.

While most Kubernetes deployments are in the cloud or benefit from skilled technicians that are physically located near their deployments (or at least equipped with remote access), our deployments are completed by installers who focus only on initial hardware installations. They never connect to the compute devices directly—rather they connect ethernet and power cords, and then look at an app to check the status of the cluster as it self-bootstraps. Replacements are completed by restaurant Owner/Operators or their teams, which are sometimes less technical.

On top of that, our Edge deployments are not exactly in a “datacenter environment”.

https://www.youtube.com/watch?v=8edDcy3oeUo

In the last year we've seen a major increase in the demand for Kubernetes outside the datacenter



Meet Target's Stores Deployment Interface that Realizes Distributed Edge Computing at Retail Scale

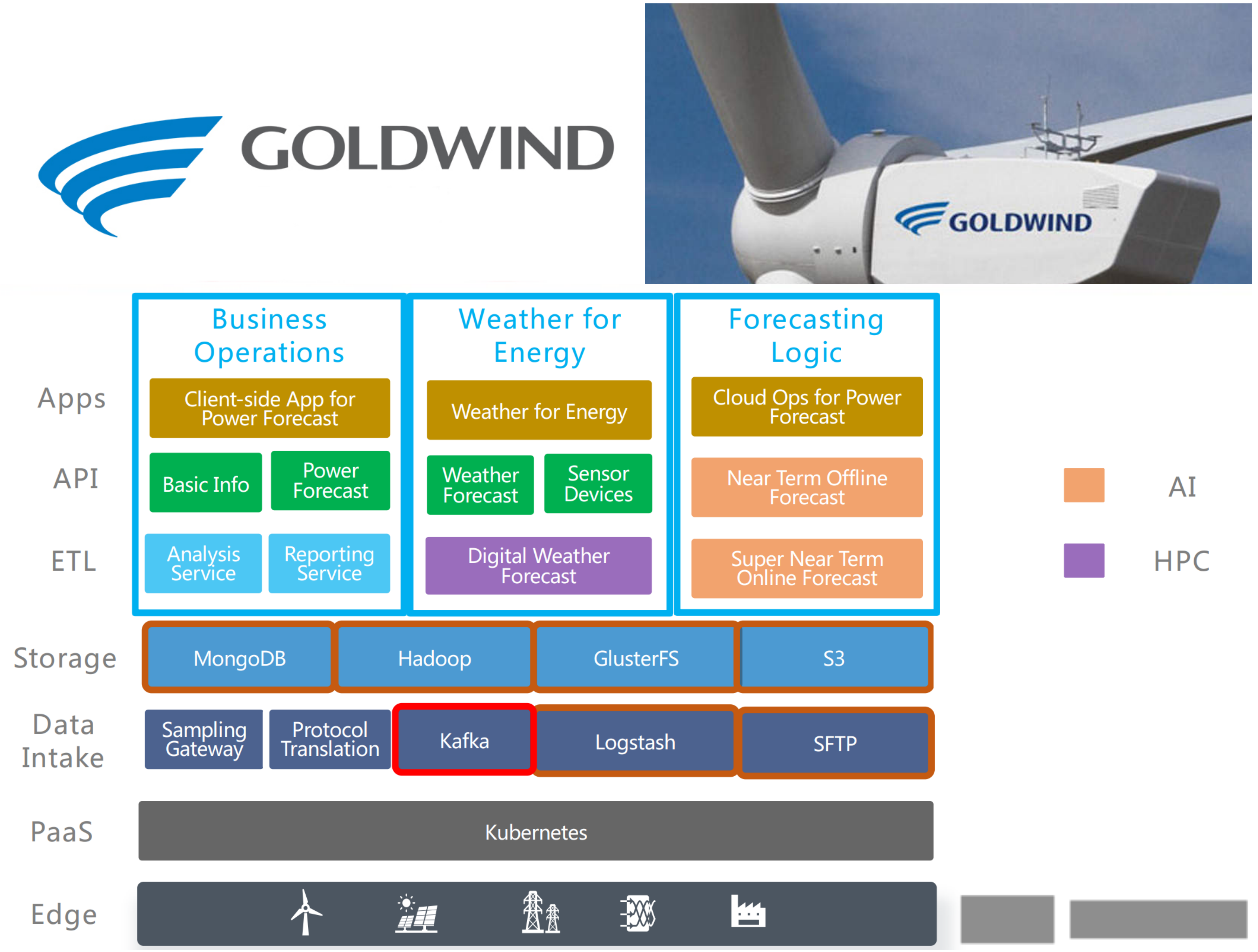
Posted by Dan Woods
Jun 20, 2018

At the beginning of 2017…

… our engineering team embarked on a journey to facilitate rapid software delivery to Target stores to better enable innovation and more quickly respond to ever-evolving business needs. The destination seemed clear and simple, but we quickly discovered the path to get there was not – not well-defined for creating a distributed software runtime platform to scale to more than 1,800 edge locations. Along the way, however, we forged a path that will forever change how we build, deploy and manage software at Target.

https://tech.target.com/infrastructure/2018/06/20/enter-unimatrix.html

In the last year we’ve seen a major increase in the demand for Kubernetes outside the datacenter



GOLDWIND

Business Operations

Weather for Energy

Forecasting Logic

Apps
Client-side App for Power Forecast
Weather for Energy
Cloud Ops for Power Forecast

API
Basic Info
Power Forecast
Weather Forecast
Sensor Devices
Near Term Offline Forecast

ETL
Analysis Service
Reporting Service
Digital Weather Forecast
Super Near Term Online Forecast

🟧 AI

🟪 HPC

Storage
MongoDB
Hadoop
GlusterFS
S3

Data Intake
Sampling Gateway
Protocol Translation
Kafka
Logstash
SFTP

PaaS
Kubernetes

Edge

https://kccncchina2018english.sched.com/event/FzEF?iframe=no

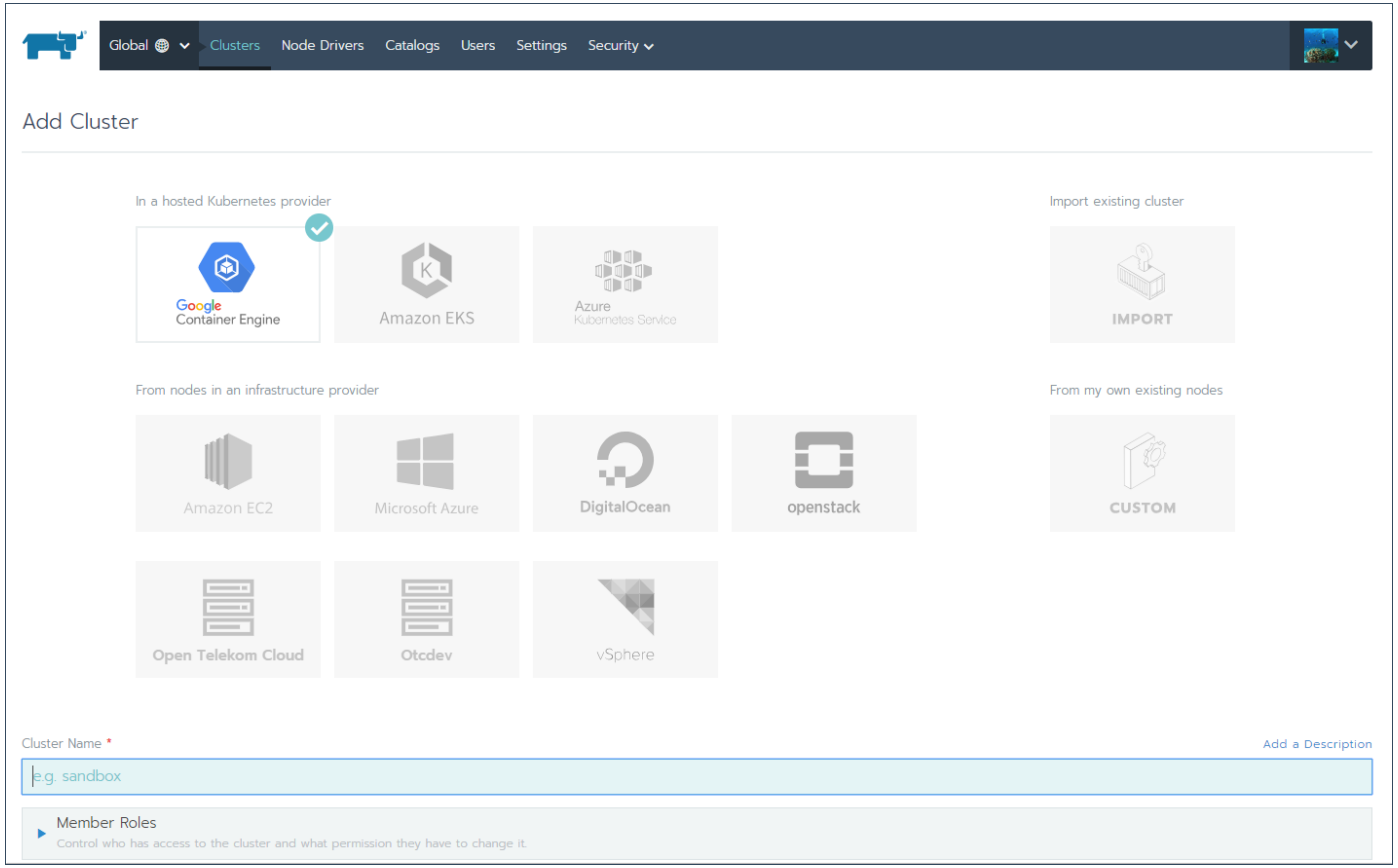
Containers and Kubernetes are an excellent choice for deploying complex software to the edge

- Containers are awesome
- Consistent across a wide variety of infrastructure
- Capable of standalone or clustered operations
- Easy to upgrade and/or replace containers
- Support for different infrastructure configs (storage, cpu, etc)
- Strong ecosystem (monitoring, logging, CI, management, etc)

Our journey to building an edge-optimized k8s distro started with Rancher

RANCHER

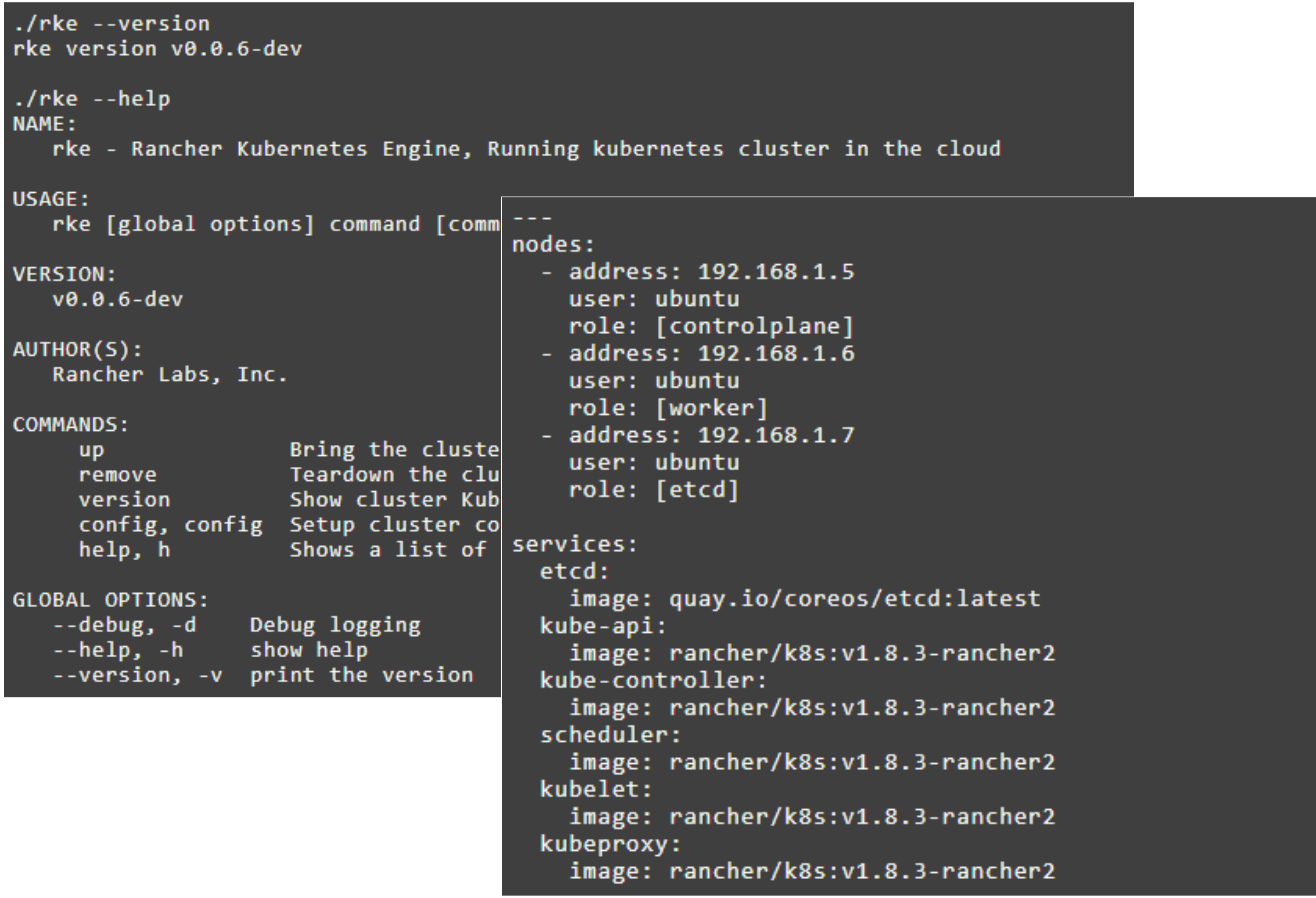
- Open source
- Multi-cluster management
- Deploy k8s anywhere
- Manage any k8s cluster



Control who has access to the cluster and what permission they have to change it.

In late 2017, we pulled out RKE our Kubernetes installer

- Open-source k8s installer and distribution
- Externally define cluster config
- Launch services and add-ons as part of deployment



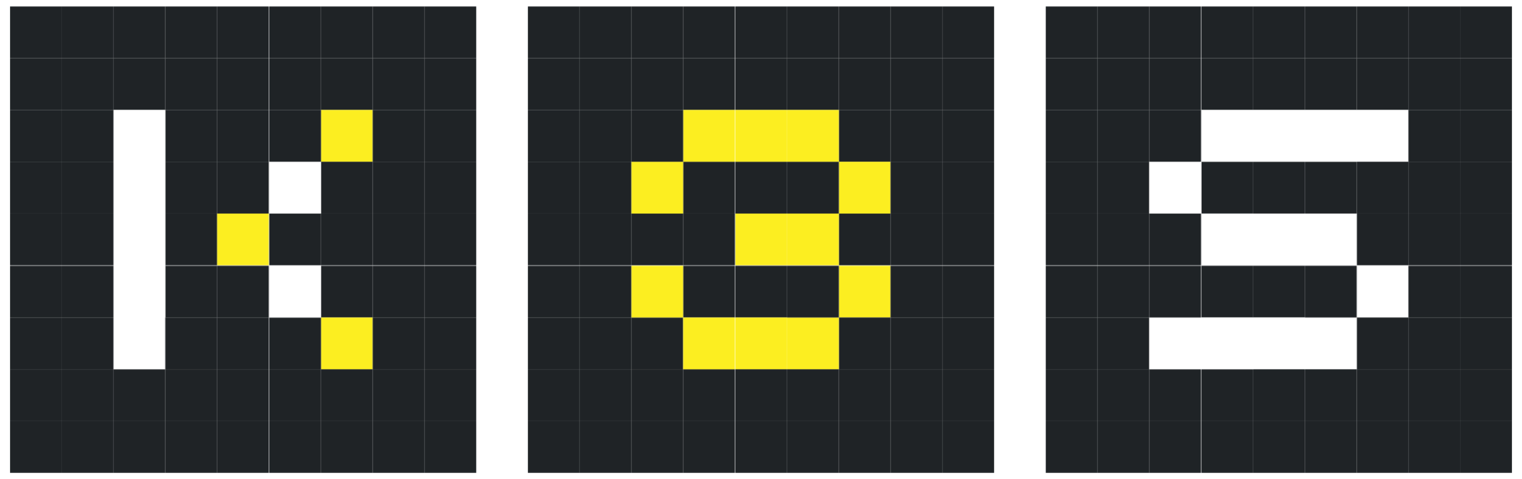
```
./rke --version
rke version v0.0.6-dev

./rke --help
NAME:
   rke - Rancher Kubernetes Engine, Running kubernetes cluster in the cloud

USAGE:
   rke [global options] command [comm

VERSION:
   v0.0.6-dev

AUTHOR(S):
   Rancher Labs, Inc.

COMMANDS:
     up                Bring the cluste
     remove            Teardown the clu
     version           Show cluster Kub
     config, config    Setup cluster co
     help, h           Shows a list of

GLOBAL OPTIONS:
   --debug, -d      Debug logging
   --help, -h       show help
   --version, -v    print the version
```

```
---
nodes:
  - address: 192.168.1.5
    user: ubuntu
    role: [controlplane]
  - address: 192.168.1.6
    user: ubuntu
    role: [worker]
  - address: 192.168.1.7
    user: ubuntu
    role: [etcd]

services:
  etcd:
    image: quay.io/coreos/etcd:latest
  kube-api:
    image: rancher/k8s:v1.8.3-rancher2
  kube-controller:
    image: rancher/k8s:v1.8.3-rancher2
  scheduler:
    image: rancher/k8s:v1.8.3-rancher2
  kubelet:
    image: rancher/k8s:v1.8.3-rancher2
  kubeproxy:
    image: rancher/k8s:v1.8.3-rancher2
```

However, running Kubernetes on the edge left us with some challenges

- Most Kubernetes distributions don't support ARM
- Kubernetes could easily consume up to 4gb of RAM
- Kubernetes wasn't built for embedded or offline management

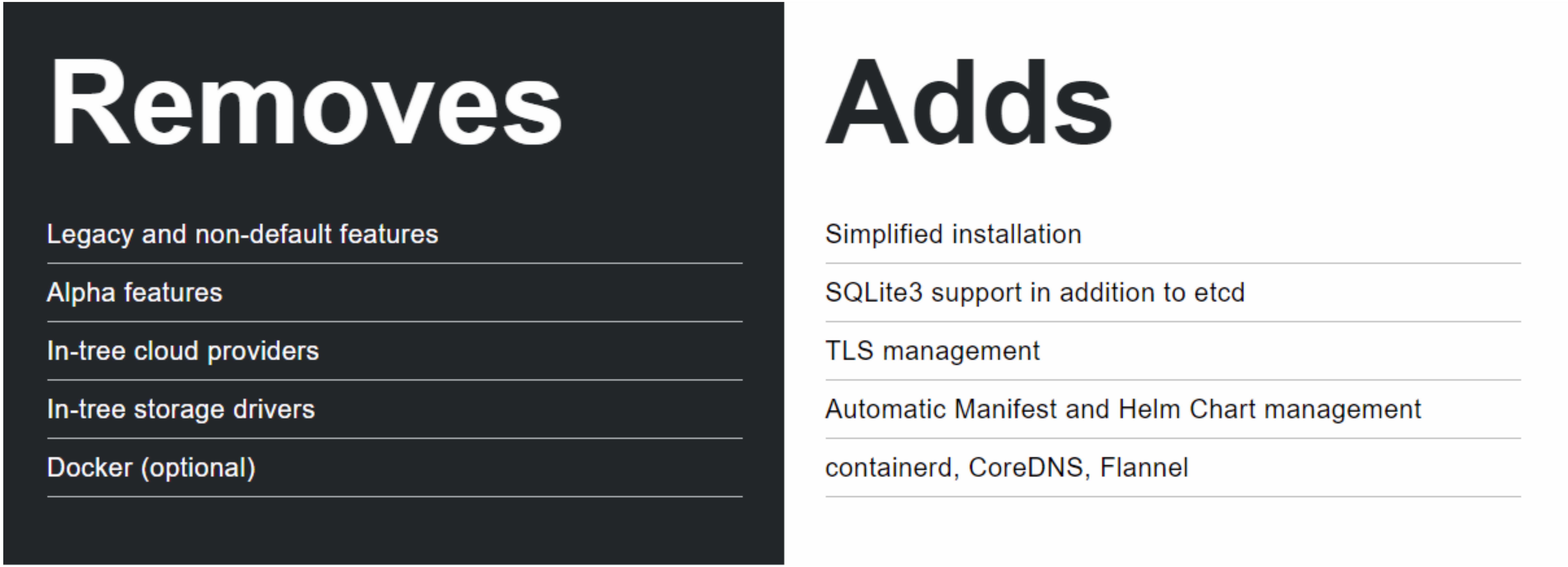
Introducing k3s



certified
kubernetes

- Lightweight certified Kubernetes distro
- Built for production operations
- 40MB binary, 512MB memory consumption
- Single process w/ integrated Kubernetes master, Kubelet, and containerd
- SQLite in addition to etcd
- Simultaneously released for x86_64, ARM64, and ARMv7
- Open source project, not yet a Rancher product

To build k3s we removed unnecessary code and made a few enhancements



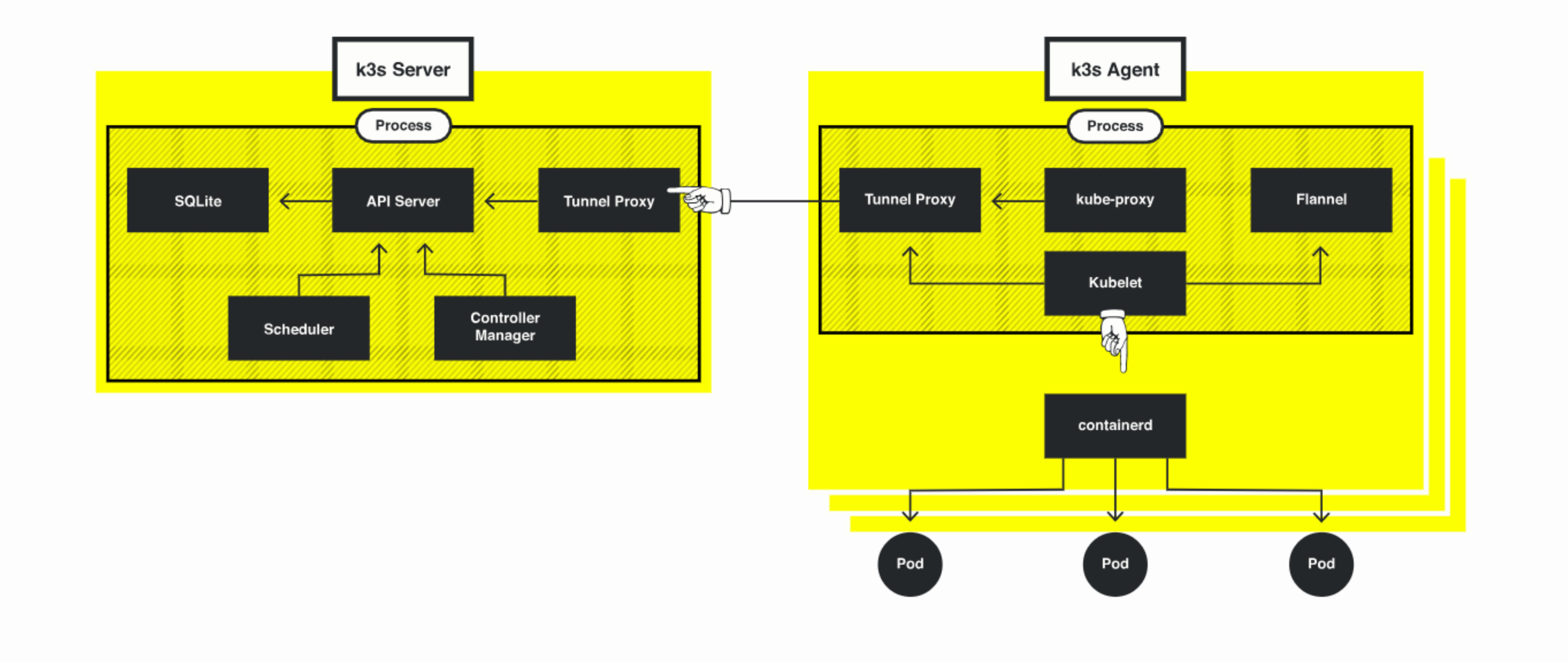
Removes

- Legacy and non-default features
- Alpha features
- In-tree cloud providers
- In-tree storage drivers
- Docker (optional)

Adds

- Simplified installation
- SQLite3 support in addition to etcd
- TLS management
- Automatic Manifest and Helm Chart management
- containerd, CoreDNS, Flannel

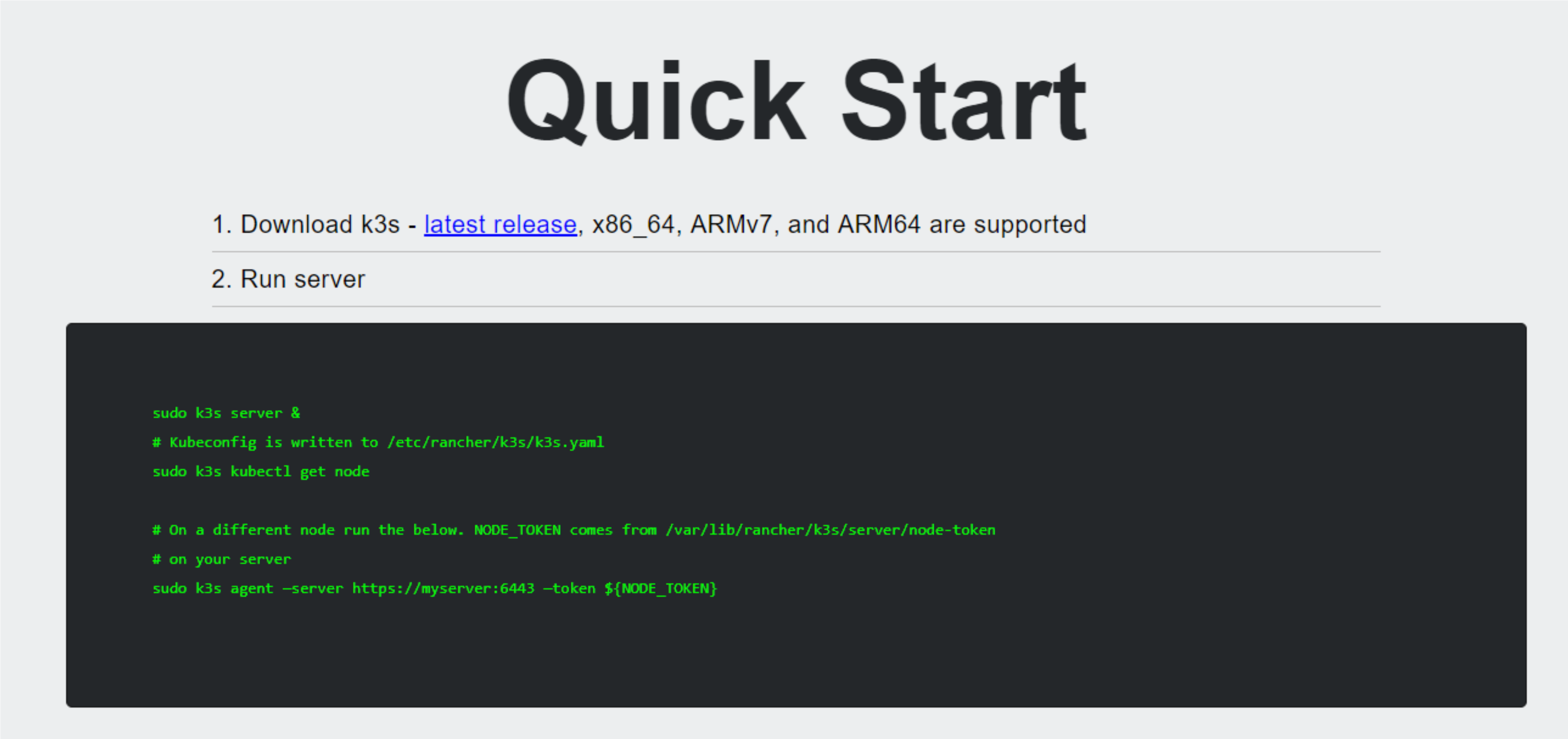
How k3s works



K3s use cases

1. Edge computing
2. Bundled into applications
3. Embedded appliances
4. CI environments

Get started with k3s in two easy steps



Quick Start

1. Download k3s - latest release, x86_64, ARMv7, and ARM64 are supported
2. Run server

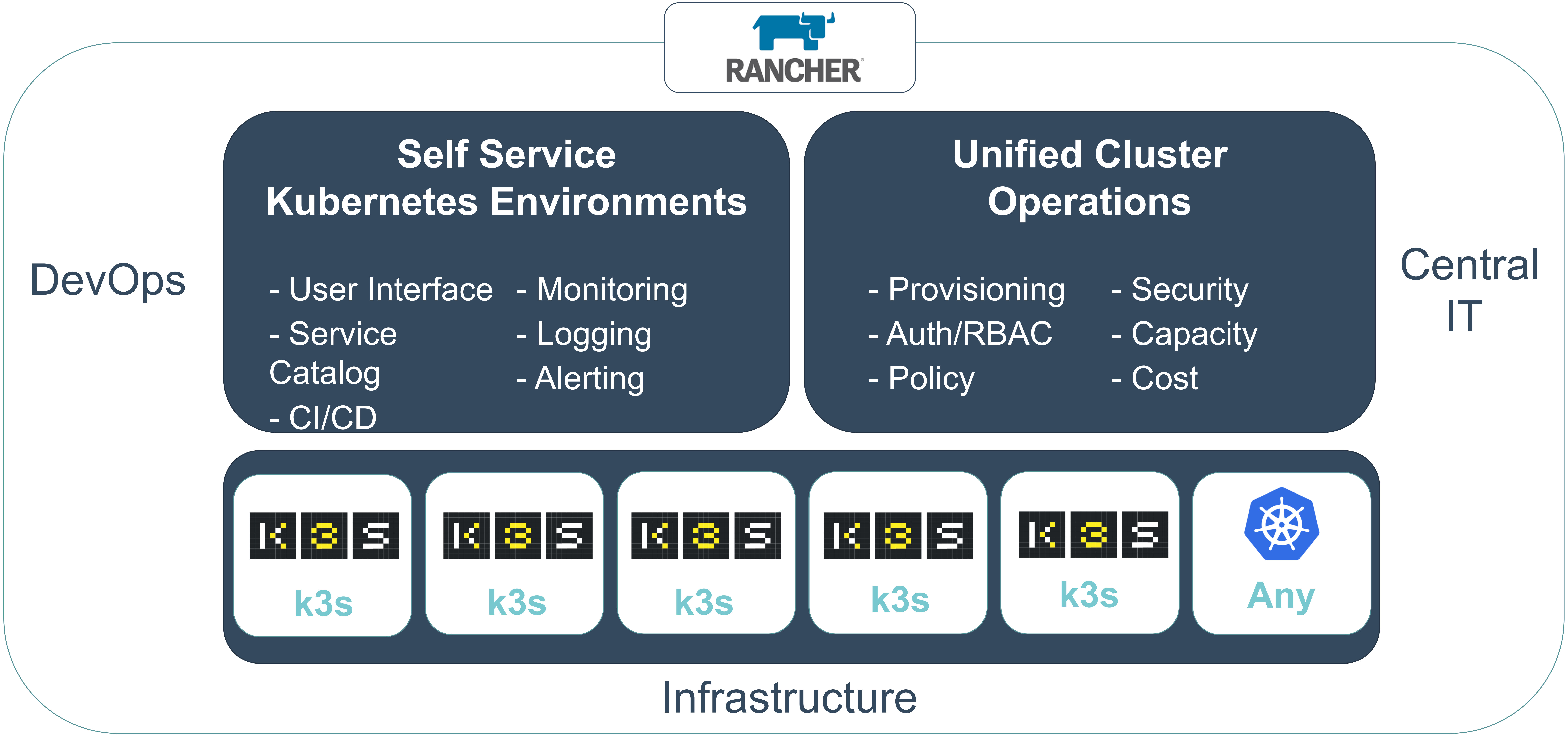
```
sudo k3s server &
# Kubeconfig is written to /etc/rancher/k3s/k3s.yaml
sudo k3s kubectl get node

# On a different node run the below. NODE_TOKEN comes from /var/lib/rancher/k3s/server/node-token
# on your server
sudo k3s agent –server https://myserver:6443 –token ${NODE_TOKEN}
```

Demo

Fleet Management for Edge Clusters

1. Multi-cluster operations (monitoring, logging, etc.)
2. Centralized access control and user management
3. Centralized security policies
4. Multi-cluster application templates
5. CI environments

Using Rancher for unified cluster operations



DevOps

Self Service Kubernetes Environments

- User Interface
- Service Catalog
- CI/CD
- Monitoring
- Logging
- Alerting

Unified Cluster Operations

- Provisioning
- Auth/RBAC
- Policy
- Security
- Capacity
- Cost

Central IT

k3s

k3s

k3s

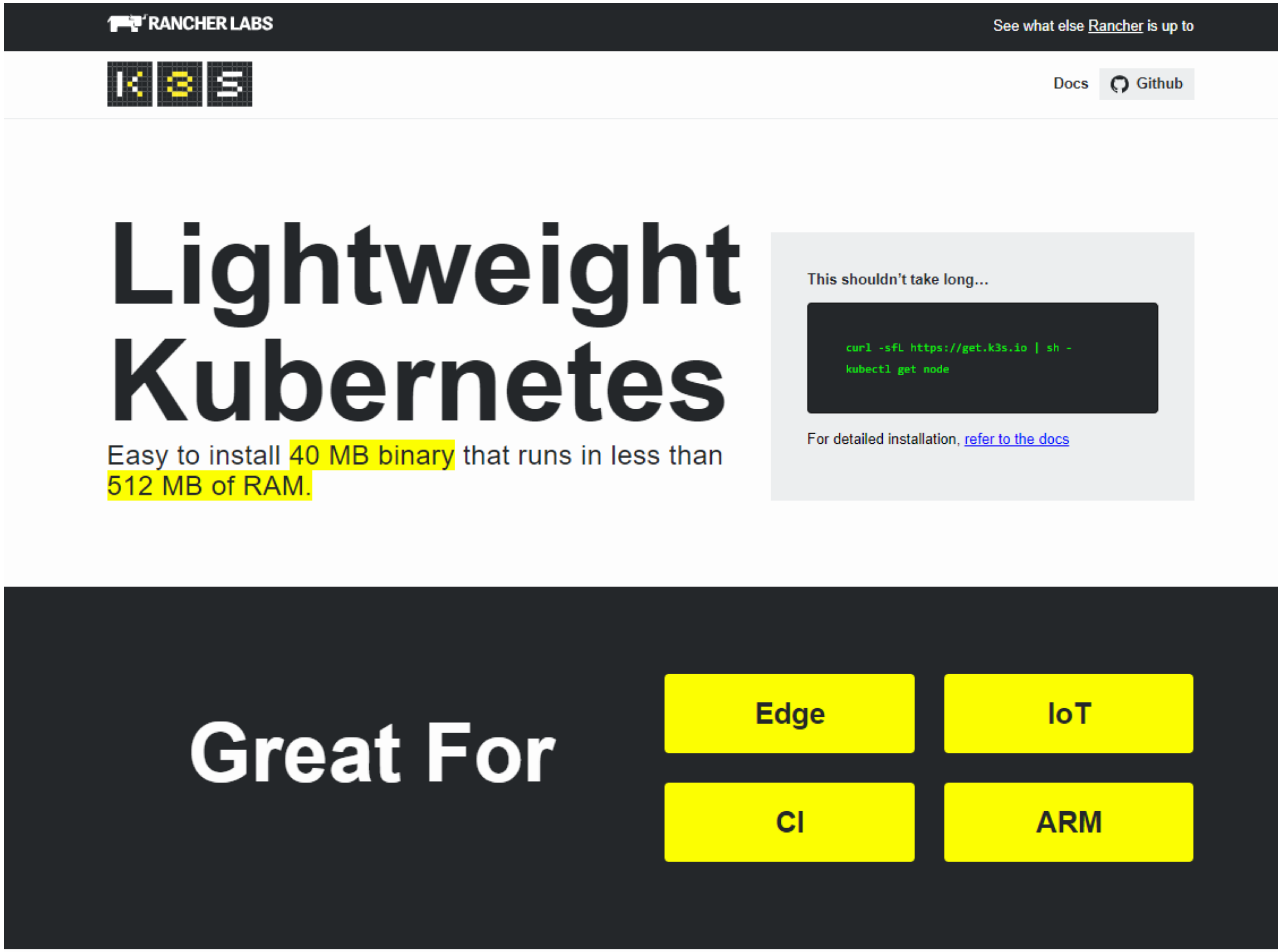
k3s

k3s

Any

Infrastructure

http://k3s.io



Lightweight Kubernetes

Easy to install 40 MB binary that runs in less than 512 MB of RAM.

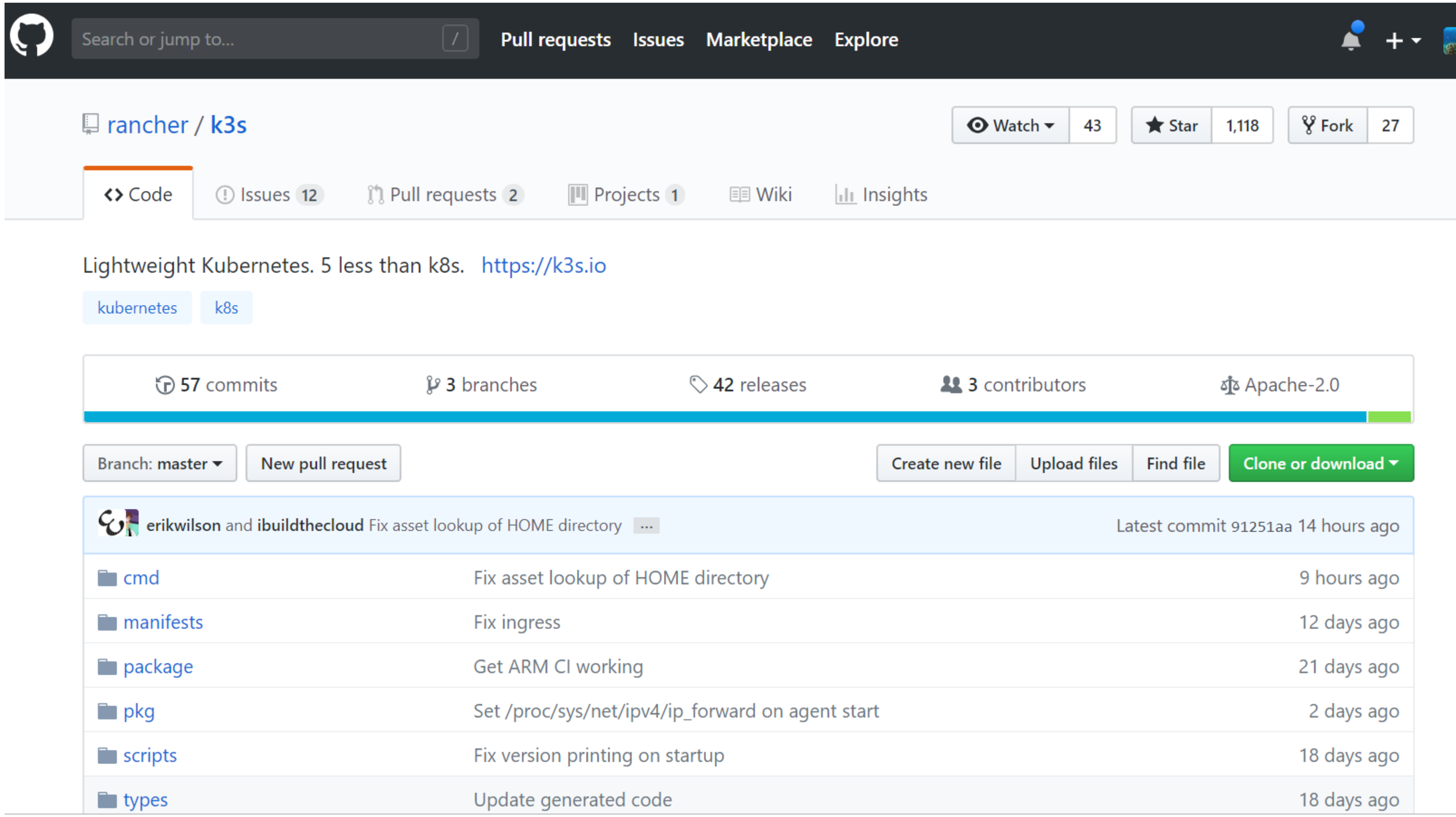
This shouldn't take long...

```
curl -sfL https://get.k3s.io | sh -
kubectl get node
```

For detailed installation, refer to the docs

Great For

https://github.com/rancher/k3s



rancher / k3s

Lightweight Kubernetes. 5 less than k8s. https://k3s.io

erikwilson and ibuildthecloud Fix asset lookup of HOME directory … Latest commit 91251aa 14 hours ago

cmd	Fix asset lookup of HOME directory	9 hours ago
manifests	Fix ingress	12 days ago
package	Get ARM CI working	21 days ago
pkg	Set /proc/sys/net/ipv4/ip_forward on agent start	2 days ago
scripts	Fix version printing on startup	18 days ago
types	Update generated code	18 days ago

Thank you

@Rancher_Labs · #RancherMeetup