



kubernetes



1. overview
2. getting started
3. tips & tricks

What is Kubernetes?

Commonly referred to as an **operating system for a datacenter**

A way to treat a selection of servers as if they were a single machine

What is it good for?

Resilience

Increase uptime, and resistance to errors

Performance

Allow more traffic / load

Deployment

A standard for running applications

Portability

If it runs on one Kubernetes, it will run on most of them

Ecosystem

Running in Kubernetes allows easy integration of infrastructure

Imagine the following scenario

You wrote a a web app in C#.

It's really pretty. You're starting to get users. It kind of works.

But... with great users comes great responsibility

They complain when your application is unavailable, or slow.

- It crashes
- It runs out of memory / cpu
- When you make updates
- It's getting more traffic than it can handle

Run multiple instances of the application on one machine

Dolor ex Machina

But...

Servers are not perfect either.

- Hardware failures
- Software failures
- Restarts
- Updates
- Can get overloaded

Run multiple instances of the application on several servers

This is YAML

```
apiVersion: v1
kind: namespace
metadata:
  name: my-first-namespace
  labels:
    important: "wrong indentation => BOOM!"
```



It's a bit like finding
a mutated kitten

You've seen better
It smells like fish
Potty training didn't take

But you get used to it



Common resources

Pod

Your application lives here. A docker image with some rules

Secret

Sensitive data - secrets - for your application

Configmap

Configuration for your application

Deployment

Describes the behaviour of your pods

Namespace

A way to separate resources

Service

A way to connect to an application across multiple instances

Ingress

Makes your services accessible outside of the cluster

...it's all YAML!

Stored inside Kubernetes (etcd)

Supports common actions

Create

Edit

Patch

List

Get

Delete

Kubernetes can be extended with
Custom Resource Definitions



What do you need?

Somewhere to run Kubernetes

Cloud / virtual machines / or local

Applications

You probably want something to run in Kubernetes. If you're just trying out, there are plenty of examples. Your application might need to be adapted. 12factor.net

A way to deploy applications

There are a lot of ways to deploy applications to Kubernetes. You probably want to automate this process. When in doubt....[GitOps!](#)

A bulletproof plan

Be clear on why you need Kubernetes - and be realistic with expectations

A Fabled YAML-ruler™

For getting all those indentations right.

So why do Fabled run it?

Standardised way of running a cluster of applications.

Unlikely to be abandoned, backed by strong foundations funded by megacorps

We make our own **kubernetes distribution** and **infrastructure bundles**.

We help other companies so **they won't have to**

No need for all companies to **reinvent the wheel**

More **cost efficient** and **flexible** than public clouds

Pure, unadulterated altruism?



fabled.se

Java Influx Ceph Odigos Agones Falco Dragonfly Arango Zitadel Prometheus ClickHouse Minio Redpanda Mimir Infisical Cilium ViteSS Postgres DGraph Alloy Go Grafana Tetragon OpenTelemetry Earthly Pomerium ArgoCD OpenObserve Dask Trivy Coroot Garage Istio TiDB Yugabyte Loki Redis Tempo Tekton Prefect Vaultwarden Rust Quickwit



The old dude is boring

Time for something

badass



DEMO!

<https://github.com/fabled-se/k8s-demo>