

Kubernetes

As Fast As Possible



James Quigley

~7 years exp in DevOps space



James Quigley

Staff Site Reliability Engineer. Tech Lead/
Engineering Manager



What does production even mean?

What happens to code after it gets written?



Development

You write code on your laptop



Review

PR created, your code gets reviewed and merged



???

Literally who knows



Profit

Code is running in production.

What happens to code after it gets written?

Breaking down step 3

1. Where does your application run?
2. How does your application receive traffic?
3. What happens if your application crashes?
4. What happens if your application can't keep up with traffic?
5. What happens if the machine your application runs on goes away?





A brief history

Physical Machines

- Still done by plenty of companies
- Cost efficient (if you can manage it well)

Virtual Machines (VMs)

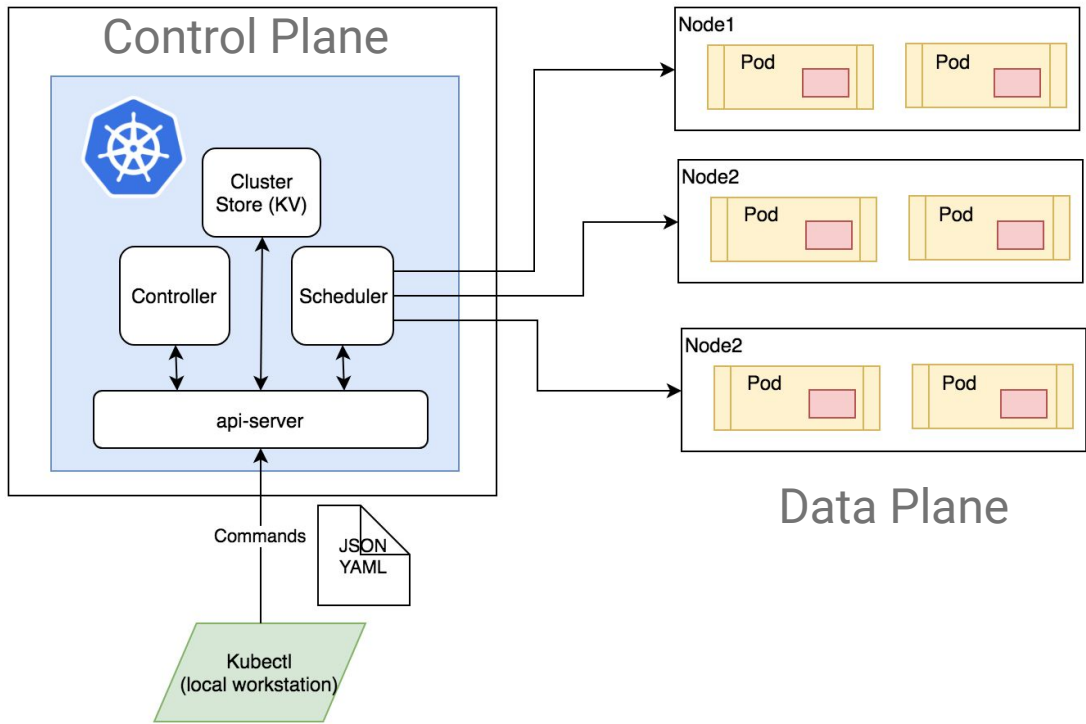
- Makes better use of the hardware
- One machine dedicated to multiple tasks
- Security benefits to isolation
- Benefit to “ephemeral” environments

Containers

- Similar benefits to VMs
- Lightweight (no OS)
- Include everything they need to run the application

Kubernetes

Kubernetes is an open source, [distributed container orchestration system](#) for automating software deployment, scaling, and management



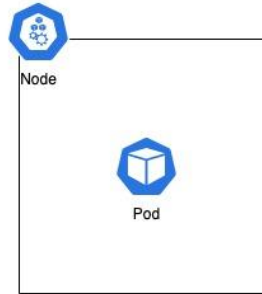
Pod

Pods are the smallest deployable units of computing that you can create and manage in Kubernetes.



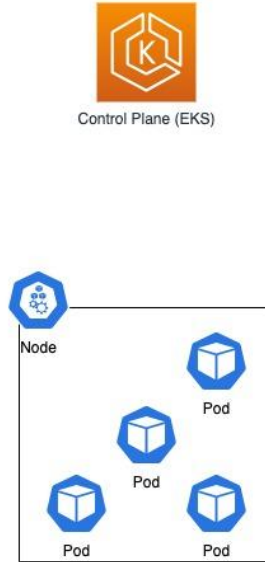
Node

Kubernetes runs your workload by placing containers into Pods to run on Nodes. A node may be a virtual or physical machine, depending on the cluster. Each node is managed by the control plane and contains the programs necessary to run Pods.



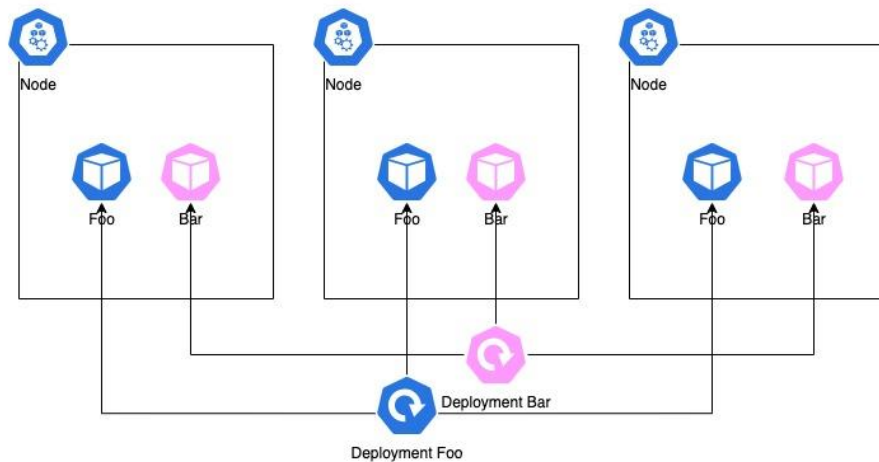
Control Plane

The “brain” of Kubernetes.
You mostly don’t need to
worry about it, unless you
are a cluster administrator.



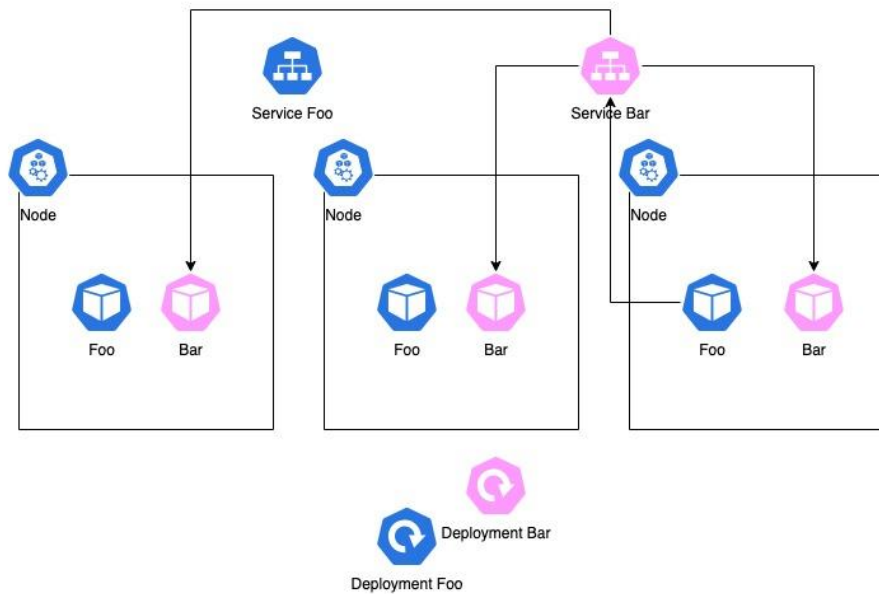
Deployment

TL;DR - a grouping of Pods.
When people talk about their “app” running in Kubernetes, they often mean a deployment. A “pod” is like a single instance, or replica of an application.



Service

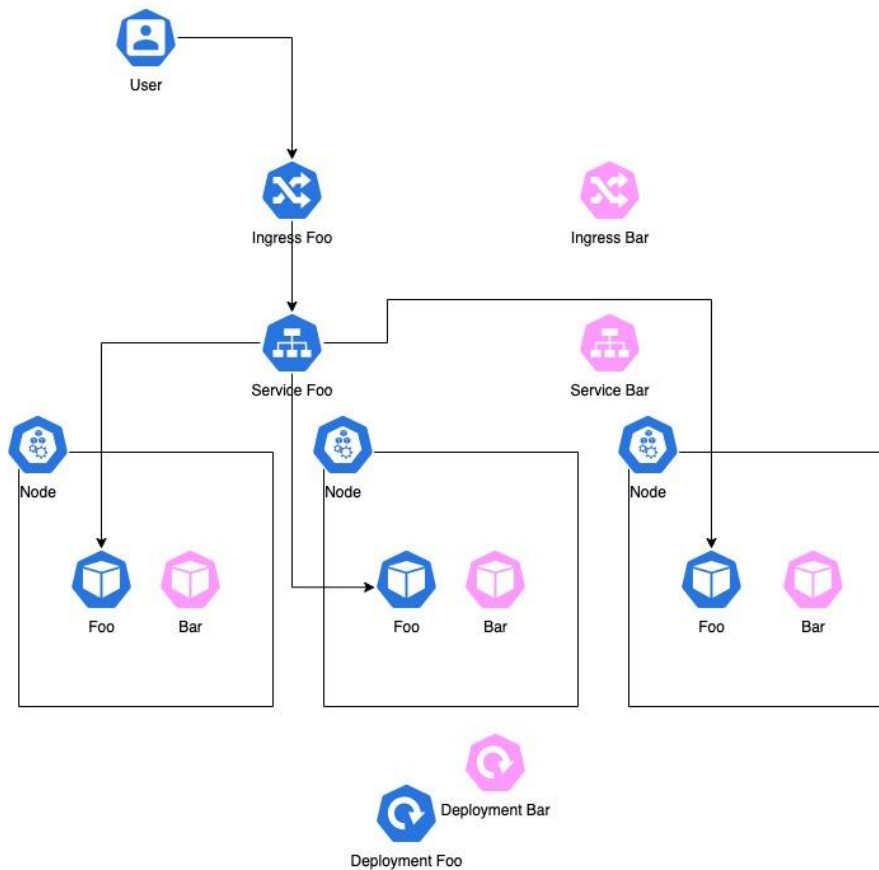
A service is responsible for routing traffic to Pods in a round-robin-ish fashion



Ingress

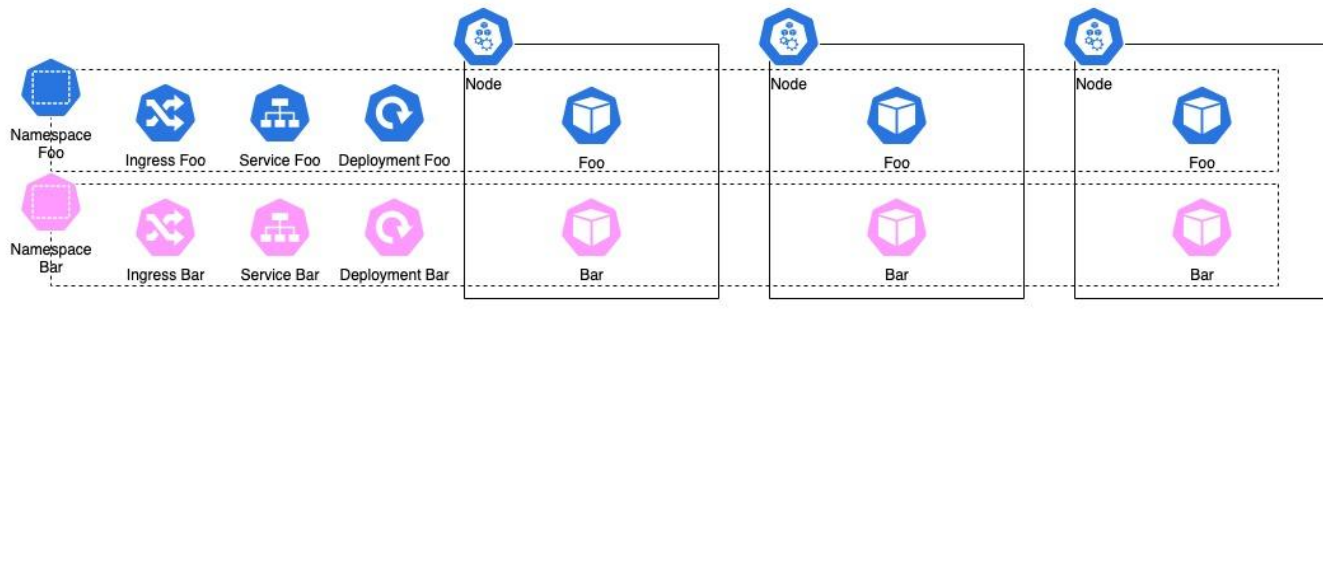
Ingress helps expose your application to traffic from outside the cluster. Can define routing rules for more complex use cases.

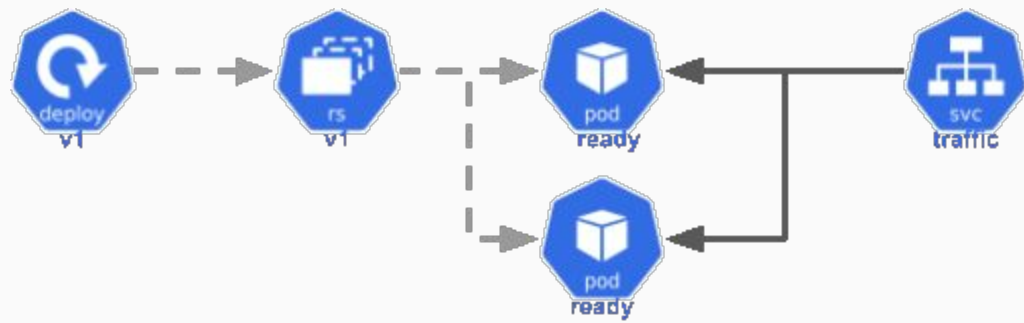
Requires you to install an “Ingress Controller” in your cluster, such as ingress-nginx



Namespace

Most resources in a Kubernetes cluster live in a namespace. A namespace groups together all different types of resources. You can control permissions at a namespace level, and you can put limits on how many resources can be used in a namespace.





Questions?

Want to work with me on
Kubernetes? Scan there ->

