Kubernetes Services
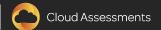
- Underlying architecture

- Pod -- Simplest kubernetes object, represents one or more containers running on a single node.

- Ephemeral, disposable & replaceable -- Stateless

- "Cattle vs. Pets"
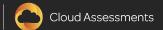
- Usually Managed via Deployments

Kubernetes Services

- Deployment specifications

- Image

- Number of replicas

  - Services -> Deployments

  - Particular port or IP address

Kubernetes Services

- Running the application pods.

- How you set up a service depends on your networking configuration and how you will handle load balancing and port forwarding.

- If you use the pod network IP address method, then a deployment gets assigned a single IP address -- even if there are multiple replicas of that pod.

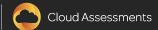- The kubernetes service (using kube-proxy on the node) redirects traffic.

Kubernetes Services

- ## Imperative
  - kubectl run nginx --image=nginx

- ## Declarative
  - apiVersion: apps/v1
    kind: Deployment
    metadata:
     name: nginx
     labels:
      app: nginx
    spec:
     replicas: 3
     selector:
      matchLabels:
       app: nginx
    ...

Kubernetes Services

Points to remember:

- Containers are run in Pods, the simplest Kubernetes object representing an application.

- Pods are (usually) managed by deployments.

- Services expose deployments.

- Third parties handle load balancing or port forwarding to those services, though Ingress objects (along with an appropriate ingress controller) are needed to do that work.