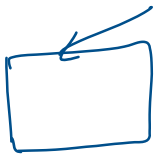
Pod - Kubernetes workloads

09:25 AM

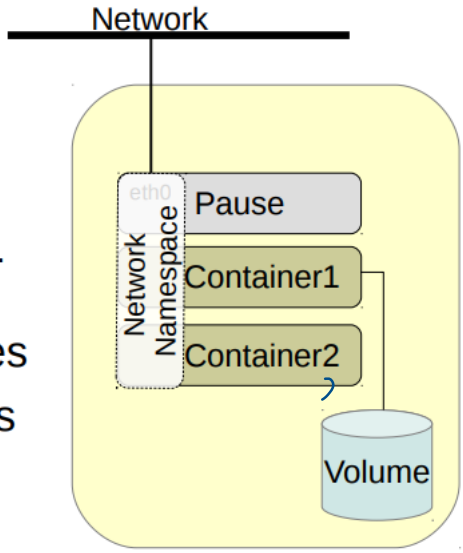
Kubernetes workloads

Pod
– Replication controllers
– Deployments, Replica sets
– Jobs and CronJobs
– DaemonSets
https://kubernetes.io/docs/concepts/workloads/pods/



The pod

- Pod - the smallest deployable object in the Kubernetes object model.
- It runs a single instance of an application
- Contains
  - One or more application containers
  - Storage resources
  - A unique IP address
  - Options about how the container(s) should run.
- Containers in one pod are sharing the network namespace and storage resources
- A pod is scheduled on a node and remains there until terminated or evicted
- Pods do not self-heal by themselves → controller.



The pod (cont)

- Pod lifecycle:
  - Pending – pod has been accepted by the Kubernetes system, but one or more of the Container images has not been created.
  - Running – has been bound to a node, all of the containers have been created. At least one container is still running (or starting / restarting).
  - Succeeded – all containers have terminated in success, and will not be restarted
  - Failed - All Containers have terminated; at least one has terminated in failure.
  - Unknown – the state of the pod could not be obtained
- Probes – performed by the kubelet on a Container using a handler
  - Probe types – what is testing: readinessProbe, livenessProbe
  - Handler Types – how is testing: ExecAction, TCPSocketAction, HTTPGetActior

- Probe result: Success, Failure, Unknown

- Restart policy – restarts a pod based on the liveness test result
  - restartPolicy: Always, OnFailure, Never
- Pods are restarted on the same node, only controllers can schedule a new pod on a different node.

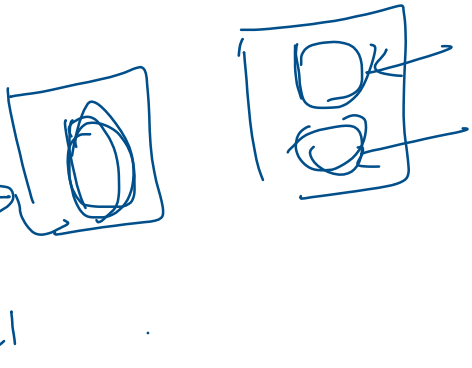
Our first Pod

Describe the Pod using a YAML file:

```
apiVersion: v1
kind: Pod
metadata:
  name: busybox
spec:
  restartPolicy: OnFailure
  containers:
    - name: busybox
      image: busybox
      command:
        - sleep
      args:
        - "100"
```

Operations on pods

- Create the pod using the kubectl command:
  - kubectl create -f pod1.yaml
- Check the pod status
  - kubectl get pod busybox [-o wide]
  - kubectl get pod --watch
- Get information about the pod
  - kubectl describe pod busybox
  - kubectl get pod busybox -o yaml
- Check the logs of a pod
  - kubectl logs busybox
- Execute a command inside the pod
  - kubectl exec -ti busybox sh



- Delete the pod
  - kubectl delete pod busybox