

Stateful Set

DCN(Distributed Cloud and Networking) Lab

StatefulSets

Manages the deployment and scaling of a set of Pods, and provides **guarantees about the ordering and uniqueness of these Pods.**

These pods are created from the same spec, but are not interchangeable: each has a **persistent identifier that it maintains across any rescheduling.**

Like a Deployment, a StatefulSet manages Pods that are based on an identical container spec.

Unlike a Deployment, a StatefulSet maintains a sticky identity for each of their Pods.

If you want to use storage volumes to provide persistence for your workload, you can use a StatefulSet as part of the solution. Although individual Pods in a StatefulSet are susceptible to failure, the persistent Pod identifiers make it easier to match existing volumes to the new Pods that replace any that have failed.

Using StatefulSets

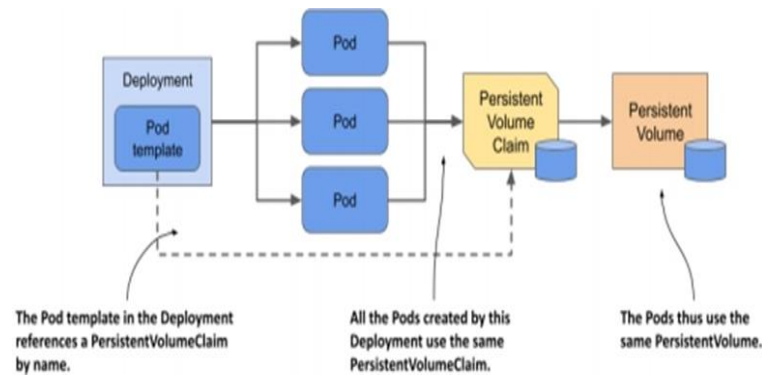
Stable, unique network identifiers.

Stable, persistent storage.

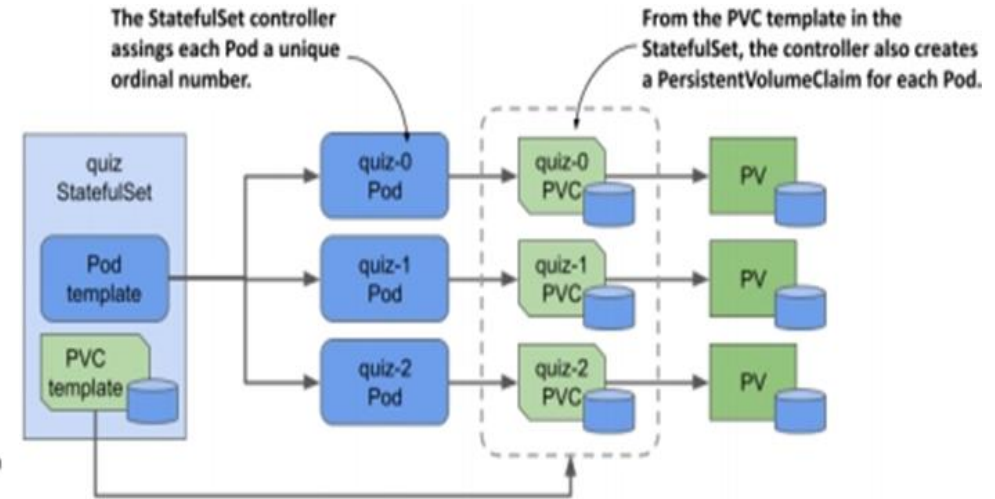
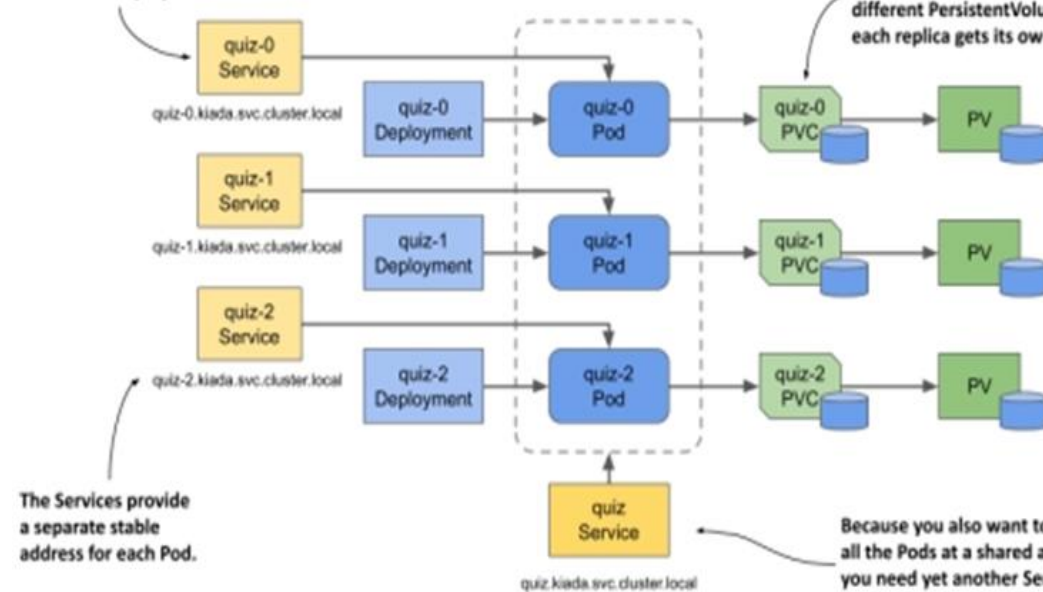
Ordered, graceful deployment and scaling.

Ordered, automated rolling updates.

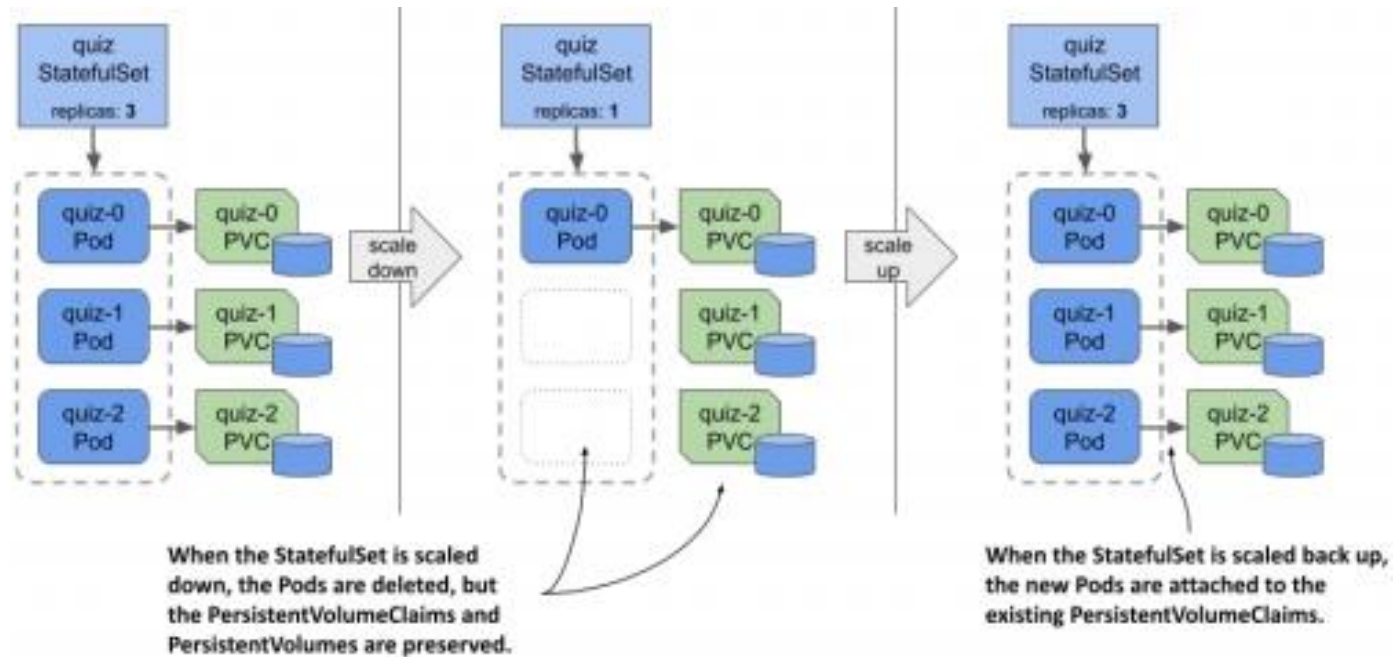
Deployment 와 StatefulSet 비교



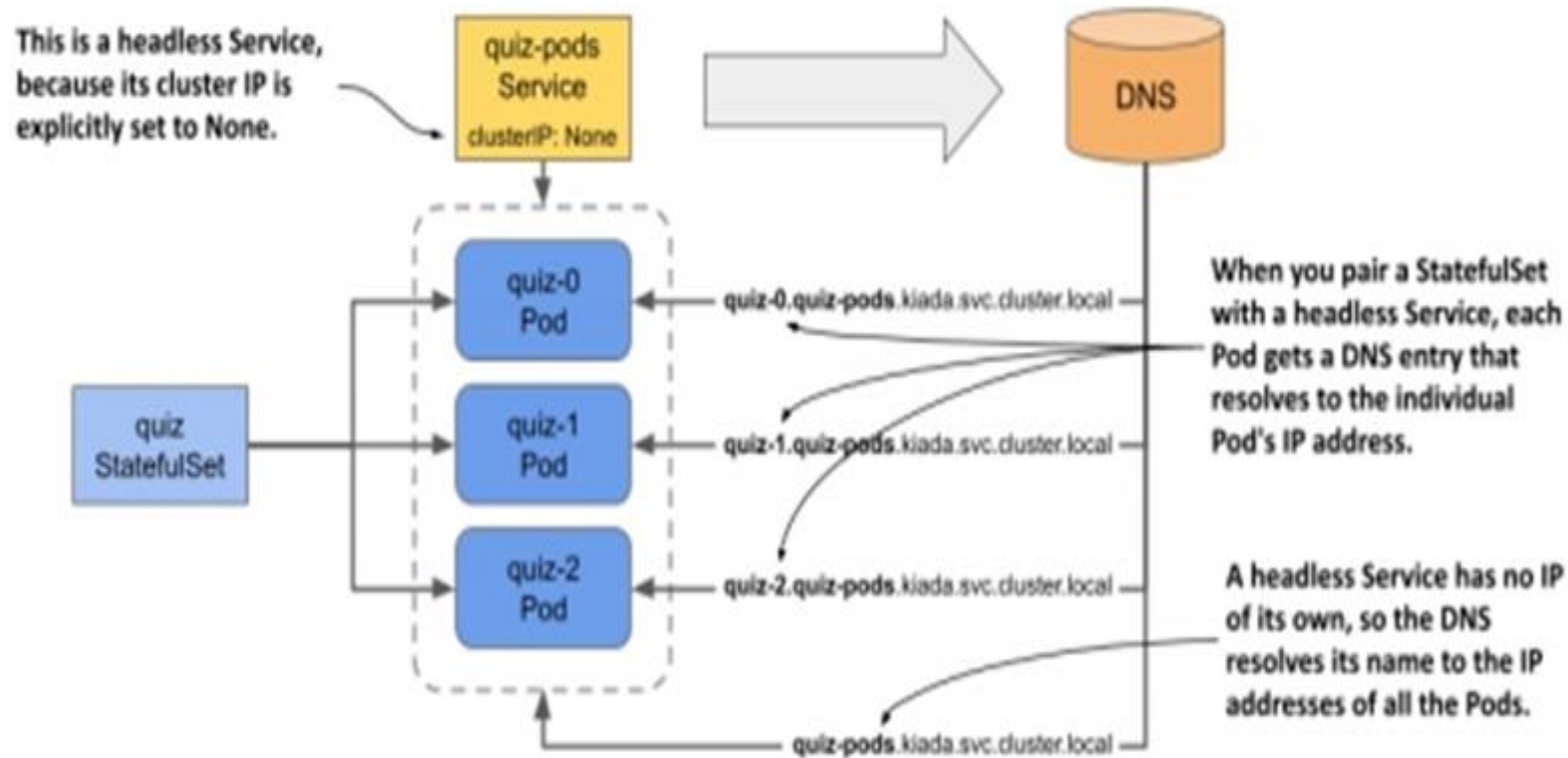
Each replica has its own Service and Deployment.



Scale Up/Down of stateful set



Headless Service for Stateful Set



apiVersion: v1
kind: Service
metadata:
 name: nginx
 labels:
 app: nginx
spec:
 ports:
 - port: 80
 name: web
clusterIP: None
 selector:
 app: nginx

apiVersion: apps/v1
kind: StatefulSet
metadata:
 name: web
spec:
 serviceName: "nginx"
 replicas: 2
 selector:
 matchLabels:
 app: nginx
 template:
 metadata:
 labels:
 app: nginx
 spec:
 containers:
 - name: nginx
 image: registry.k8s.io/nginx-slim:0.8
 ports:
 - containerPort: 80
 name: web
 volumeMounts:
 - name: www
 mountPath: /usr/share/nginx/html

volumeClaimTemplates:
 - metadata:
 name: www
 spec:
 accessModes: ["ReadWriteOnce"]
 resources:
 requests:
 storage: 1Gi

Limitations

The storage for a given Pod must either be provisioned by a PersistentVolume Provisioner based on the requested storage class, or pre-provisioned by an admin.

Deleting and/or scaling a StatefulSet down will not delete the volumes associated with the StatefulSet.

StatefulSets currently require a Headless Service to be responsible for the network identity of the Pods.

StatefulSets do not provide any guarantees on the termination of pods when a StatefulSet is deleted. To achieve ordered and graceful termination of the pods in the StatefulSet, it is possible to scale the StatefulSet down to 0 prior to deletion.