

03-deployment

Deployment What it is Deployment manages stateless application rollout and scaling using ReplicaSets. When to use Web/API applications Rolling updates and rollbacks Declarative scaling of stateless Pods Key fields spec.replicas: desired Pod count spec.selector.matchLabels: must match Pod template labels spec.template: Pod template spec.strategy.rollingUpdate: update surge/unavailable settings Common commands bash kubectl get deploy kubectl create deployment web --image=nginx:1.27 kubectl scale deployment web --replicas=4 kubectl set image deployment/web nginx=nginx:1.28 kubectl rollout status deployment/web kubectl rollout history deployment/web kubectl rollout undo deployment/web kubectl delete deployment web YAML example yaml apiVersion: apps/v1 kind: Deployment metadata: name: web spec: replicas: 3 selector: matchLabels: app: web strategy: type: RollingUpdate rollingUpdate: maxSurge: 1 maxUnavailable: 0 template: metadata: labels: app: web spec: containers: - name: web image: nginx:1.27 ports: - containerPort: 80 readinessProbe: httpGet: path: / port: 80 initialDelaySeconds: 5 periodSeconds: 10 Practical notes Keep health probes correct to avoid broken rollouts. Rollbacks work best when image tags are immutable.