# Deployment Strategy

Tuesday, January 5, 2021 2:49 PM

## Deployment Strategy

### ► Rolling update

```
[root@master ~]# cat deployments_rollingupdate.yml
apiVersion: apps/v1
kind: Deployment
metadata:
 name: myapp-deployment
 labels:
  app: nginx
spec:
 template:
   metadata:
     name: myap-pod
     labels:
       app: myapp
       type: front-end
   spec:
    containers:
    - name: nginx-container
      image: nginx:1.7.1
 replicas: 3
 strategy:
  type: RollingUpdate
 selector:
  matchLabels:
    type: front-end
```

Rolling update is by default but we explicitly defined here

**Create Deployment**

# kubectl apply -f deployments_rollingupdate.yml

```
[root@master ~]# kubectl apply -f deployments_rollingupdate.yml
deployment.apps/myapp-deployment created
```

**Deployment status**

# kubectl describe deployment myapp-deployment

```
[root@master ~]# kubectl describe deployment myapp-deployment
Name:                   myapp-deployment
Namespace:              default
CreationTimestamp:      Tue, 05 Jan 2021 00:47:13 +0530
Labels:                 app=nginx
Annotations:            deployment.kubernetes.io/revision: 1
Selector:               type=front-end
Replicas:               3 desired | 3 updated | 3 total | 0 available | 3 unavailable
StrategyType:           RollingUpdate
MinReadySeconds:        0
RollingUpdateStrategy:  25% max unavailable, 25% max surge
Pod Template:
  Labels:  app=myapp
           type=front-end
  Containers:
   nginx-container:
    Image:        nginx:1.7.1
    Port:         <none>
    Host Port:    <none>
    Environment:  <none>
    Mounts:       <none>
  Volumes:        <none>
Conditions:
  Type           Status  Reason
  ----           ------  ------
```

**Two ways to update image:** Either edit file with new version and save or use below command to update image

# kubectl set image deployment/myapp-deployment nginx-container=nginx:1.9.1

```
[root@master ~]# kubectl set image deployment/myapp-deployment nginx-container=nginx:1.9.1
deployment.apps/myapp-deployment image updated
```

**Deployment status after version upgrade**

# kubectl describe deployment myapp-deployment

```
 Containers:
  nginx-container:
   Image:        nginx:1.9.1
   Port:         <none>
   Host Port:    <none>
   Environment:  <none>
   Mounts:       <none>
 Volumes:        <none>
Conditions:
 Type           Status  Reason
 ----           ------  ------
 Available      True    MinimumReplicasAvailable
 Progressing    True    NewReplicaSetAvailable
OldReplicaSets:  <none>
NewReplicaSet:   myapp-deployment-dbd96b5f8 (3/3 replicas created)
Events:
 Type    Reason             Age    From                   Message
 ----    ------             ----   ----                   -------
 Normal  ScalingReplicaSet  12m    deployment-controller  Scaled up replica set myapp-deployment-5c7c75f4d8 to 3
 Normal  ScalingReplicaSet  3m48s  deployment-controller  Scaled up replica set myapp-deployment-dbd96b5f8 to 1
 Normal  ScalingReplicaSet  3m14s  deployment-controller  Scaled down replica set myapp-deployment-5c7c75f4d8 to 2
 Normal  ScalingReplicaSet  3m14s  deployment-controller  Scaled up replica set myapp-deployment-dbd96b5f8 to 2
 Normal  ScalingReplicaSet  3m11s  deployment-controller  Scaled down replica set myapp-deployment-5c7c75f4d8 to 1
 Normal  ScalingReplicaSet  3m11s  deployment-controller  Scaled up replica set myapp-deployment-dbd96b5f8 to 3
 Normal  ScalingReplicaSet  3m9s   deployment-controller  Scaled down replica set myapp-deployment-5c7c75f4d8 to 0
[root@master ~]#
```

## ► Recreate

```
[root@master ~]# cat deployments_recreate.yml
apiVersion: apps/v1
kind: Deployment
metadata:
 name: myapp-deployment
 labels:
  app: nginx
spec:
 template:
   metadata:
     name: myap-pod
     labels:
       app: myapp
       type: front-end
   spec:
    containers:
    - name: nginx-container
      image: nginx:1.7.1
 replicas: 3
 strategy:
  type: Recreate
 selector:
  matchLabels:
    type: front-end
```

**Create Deployment**

# kubectl apply -f deployments_recreate.yml

```
[root@master ~]# kubectl apply -f deployments_recreate.yml
deployment.apps/myapp-deployment created
```

**Status of Deployment**

# kubectl describe deployment myapp-deployment

```
[root@master ~]# kubectl describe deployment myapp-deployment
Name:               myapp-deployment
Namespace:          default
CreationTimestamp:  Tue, 05 Jan 2021 01:04:13 +0530
Labels:             app=nginx
Annotations:        deployment.kubernetes.io/revision: 1
Selector:           type=front-end
Replicas:           3 desired | 3 updated | 3 total | 3 available | 0 unavailable
StrategyType:       Recreate
MinReadySeconds:    0
Pod Template:
  Labels:  app=myapp
           type=front-end
  Containers:
   nginx-container:
    Image:        nginx:1.7.1
    Port:         <none>
    Host Port:    <none>
    Environment:  <none>
    Mounts:       <none>
  Volumes:        <none>
Conditions:
  Type           Status  Reason
  ----           ------  ------
  Available      True    MinimumReplicasAvailable
  Progressing    True    NewReplicaSetAvailable
```

**Two ways to update image:** Either edit file with new version and save or use below command to update image

**#** kubectl set image deployment/myapp-deployment nginx-container=nginx:1.9.1

```
[root@master ~]# kubectl set image deployment/myapp-deployment nginx-container=nginx:1.9.1
deployment.apps/myapp-deployment image updated
```

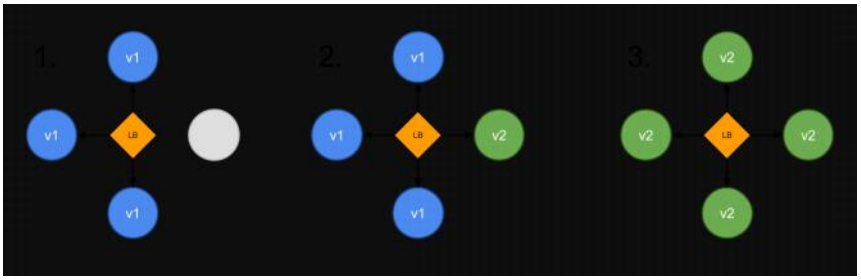
**Deployment status after version upgrade**

**#** kubectl describe deployment myapp-deployment

```
StrategyType:       Recreate
MinReadySeconds:    0
Pod Template:
  Labels:  app=myapp
           type=front-end
  Containers:
   nginx-container:
    Image:        nginx:1.9.1
    Port:         <none>
    Host Port:    <none>
    Environment:  <none>
    Mounts:       <none>
  Volumes:        <none>
Conditions:
  Type           Status  Reason
  ----           ------  ------
  Available      True    MinimumReplicasAvailable
  Progressing    True    NewReplicaSetAvailable
OldReplicaSets:  <none>
NewReplicaSet:   myapp-deployment-dbd96b5f8 (3/3 replicas created)
Events:
  Type    Reason             Age    From                   Message
  ----    ------             ----   ----                   -------
  Normal  ScalingReplicaSet  2m57s  deployment-controller  Scaled up replica set myapp-deployment-5c7c75f4d8 to 3
  Normal  ScalingReplicaSet  42s    deployment-controller  Scaled down replica set myapp-deployment-5c7c75f4d8 to 0
  Normal  ScalingReplicaSet  37s    deployment-controller  Scaled up replica set myapp-deployment-dbd96b5f8 to 3
```

## ► Canary Strategy

One replica of new version is rolled out along with old version.
Scale up if no error is detected.




**Create Load Balancer Service**

```
[root@master ~]# cat service-canary.yml
apiVersion: v1
kind: Service
metadata:
  name: service-canary
spec:
  type: LoadBalancer
  ports:
  - port: 80
  selector:
    app: nginx
```

```
[root@master ~]# kubectl create -f service-canary.yml
service/service-canary created
```

**Create Deployment for Version 1**

```
[root@master ~]# cat deployments_canary_v1.yml
apiVersion: apps/v1
kind: Deployment
metadata:
 name: myapp-deployment-canaryv1
spec:
 template:
   metadata:
     labels:
       app: nginx
   spec:
    containers:
    - name: nginx-container
      image: nginx:1.7.1
 replicas: 2
 selector:
    matchLabels:
      app: nginx
```

```
[root@master ~]# kubectl apply -f deployments_canary_v1.yml
deployment.apps/myapp-deployment-canaryv1 configured
```

**Create Deployment of Version 2**

```
[root@master ~]# cat deployments_canary_v2.yml
apiVersion: apps/v1
kind: Deployment
metadata:
 name: myapp-deployment-canaryv2
spec:
 template:
   metadata:
     labels:
       app: nginx
   spec:
    containers:
    - name: nginx-container
      image: nginx:1.9.1
 replicas: 2
 selector:
    matchLabels:
      app: nginx
```

```
[root@master ~]# kubectl apply -f deployments_canary_v2.yml
deployment.apps/myapp-deployment-canaryv2 created
```

**Status of pod with both version**

```
[root@master ~]# kubectl get pods -o wide
NAME                                      READY   STATUS    RESTARTS   AGE   IP           NODE      NOMINATED NODE   READINESS GA
ES
myapp-deployment-canaryv1-6cbff78895-fjbn6   1/1     Running   0          42s   10.44.0.10   worker1   <none>           <none>
myapp-deployment-canaryv1-6cbff78895-gstcb   1/1     Running   0          94m   10.44.0.9    worker1   <none>           <none>
myapp-deployment-canaryv2-7877cc4f6f-sdnkc   1/1     Running   0          6s    10.44.0.11   worker1   <none>           <none>
myapp-deployment-canaryv2-7877cc4f6f-xq257   1/1     Running   0          6s    10.44.0.8    worker1   <none>           <none>
```

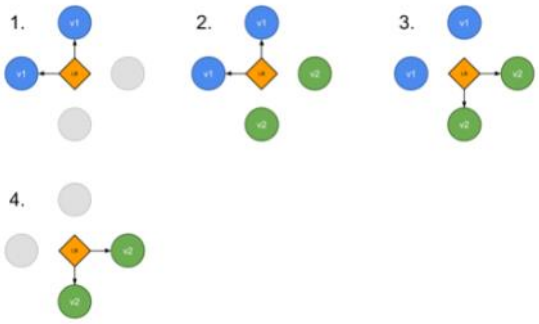
with old version

with new version

**Service Status shows endpoint for both versions**

```
[root@master ~]# kubectl describe service service-canary
Name:                     service-canary
Namespace:                default
Labels:                   <none>
Annotations:              <none>
Selector:                 app=nginx
Type:                     LoadBalancer
IP Families:              <none>
IP:                       10.109.165.156
IPs:                      10.109.165.156
Port:                     <unset>  80/TCP
TargetPort:               80/TCP
NodePort:                 <unset>  31603/TCP
Endpoints:                10.44.0.10:80,10.44.0.11:80,10.44.0.8:80 + 1 more...
Session Affinity:         None
External Traffic Policy:  Cluster
Events:
  Type     Reason                      Age                 From                     Message
  ----     ------                      ----                ----                     -------
  Warning  FailedToUpdateEndpointSlices  38m (x5 over 103m)  endpoint-slice-controller  Error updating Endpoint Slices for Service de
fault/service-canary: failed to update service-canary-j4md5 EndpointSlice for Service default/service-canary: Operation cannot be ful
filled on endpointslices.discovery.k8s.io "service-canary-j4md5": the object has been modified; please apply your changes to the late
st version and try again
```

- Blue Green Strategy :




1. version 1 is serving traffic
2. deploy version 2
3. wait until version 2 is ready
4. switch incoming traffic from version 1 to version 2

**Create Service ( Node Port) --> Deploy V1 ---> Deploy V2 ---> switch traffic**

**Create NodePort service**

```
[root@master ~]# cat service-bluegreen.yml
apiVersion: v1
kind: Service
metadata:
  name: service-bluegreen
  labels:
    app: my-app
spec:
  type: NodePort
  ports:
  - name: http
    port: 80
    targetPort: http

  # Note here that we match both the app and the version
  selector:
    app: my-app
    version: v1.0.0
```

```
[root@master ~]# kubectl apply -f service-bluegreen.yml
service/service-bluegreen created
```

**Status of service**

```
[root@master ~]# kubectl get services
NAME                TYPE           CLUSTER-IP       EXTERNAL-IP   PORT(S)        AGE
kubernetes          ClusterIP      10.96.0.1        <none>        443/TCP        2d12h
service-bluegreen   NodePort       10.100.174.55    <none>        80:30281/TCP   32s
service-canary      LoadBalancer   10.109.165.156   <pending>     80:31603/TCP   8h
```

**Create deployment of version 1**

```
[root@master ~]# cat deployment_bluegreenv1.yml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: bluegreenv1
  labels:
    app: my-app
spec:
  replicas: 3
  selector:
    matchLabels:
      app: my-app
      version: v1.0.0
  template:
    metadata:
      labels:
        app: my-app
        version: v1.0.0
    spec:
      containers:
      - name: my-app
        image: containersol/k8s-deployment-strategies
        ports:
        - name: http
          containerPort: 8080
```

**Create deployment of Version 2**

```
[root@master ~]# cat deployment_bluegreenv2.yml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: bluegreenv2
  labels:
    app: my-app
spec:
  replicas: 3
  selector:
    matchLabels:
      app: my-app
      version: v2.0.0
  template:
    metadata:
      labels:
        app: my-app
        version: v2.0.0
    spec:
      containers:
      - name: my-app
        image: containersol/k8s-deployment-strategies
        ports:
        - name: http
          containerPort: 8080
```

```
[root@master ~]# kubectl create -f deployment_bluegreenv1.yml
deployment.apps/bluegreenv1 created
```

```
[root@master ~]# kubectl apply -f deployment_bluegreenv2.yml
deployment.apps/bluegreenv2 created
```

**Pod status of blue green deployment of both version**

```
[root@master ~]# kubectl get po |grep bluegreen
bluegreenv1-9984f47bd-87xnt                1/1     Running            0          19m
bluegreenv1-9984f47bd-8g8z7                1/1     Running            0          19m
bluegreenv1-9984f47bd-fttzk                1/1     Running            0          19m
bluegreenv2-b88967595-9gczn                1/1     Running            0          8m22s
bluegreenv2-b88967595-pmpsx                1/1     Running            0          8m22s
bluegreenv2-b88967595-q9ggm                1/1     Running            0          8m22s
```

**Verify from curl command with NodePort ip of node**

# while true; do curl http://192.168.85.123:30281; sleep 2s; done

```
[root@master ~]# while true; do curl http://192.168.85.123:30281; sleep 2s; done
Host: bluegreenv1-9984f47bd-8g8z7, Version:
Host: bluegreenv1-9984f47bd-8g8z7, Version:
Host: bluegreenv1-9984f47bd-87xnt, Version:
Host: bluegreenv1-9984f47bd-87xnt, Version:
Host: bluegreenv1-9984f47bd-fttzk, Version:
Host: bluegreenv1-9984f47bd-87xnt, Version:
Host: bluegreenv1-9984f47bd-87xnt, Version:
Host: bluegreenv1-9984f47bd-fttzk, Version:
Host: bluegreenv1-9984f47bd-fttzk, Version:
Host: bluegreenv1-9984f47bd-8g8z7, Version:
Host: bluegreenv1-9984f47bd-87xnt, Version:
Host: bluegreenv1-9984f47bd-fttzk, Version:
Host: bluegreenv1-9984f47bd-fttzk, Version:
```

Traffic is going to version 1

**Now change selector section to V2 in service-bluegreen.yml file and apply changes**

```
selector:
  app: my-app
  version: v2.0.0
```

```
[root@master ~]# kubectl apply -f service-bluegreen.yml
service/service-bluegreen configured
```

**Traffic switched to Version 2**

```
[root@master ~]# while true; do curl http://192.168.85.123:30281; sleep 2s; done
Host: bluegreenv2-b88967595-9gczn, Version:
Host: bluegreenv2-b88967595-q9ggm, Version:
Host: bluegreenv2-b88967595-pmpsx, Version:
Host: bluegreenv2-b88967595-q9ggm, Version:
Host: bluegreenv2-b88967595-q9ggm, Version:
Host: bluegreenv2-b88967595-q9ggm, Version:
Host: bluegreenv2-b88967595-q9ggm, Version:
Host: bluegreenv2-b88967595-pmpsx, Version:
Host: bluegreenv2-b88967595-9gczn, Version:
Host: bluegreenv2-b88967595-9gczn, Version:
Host: bluegreenv2-b88967595-q9ggm, Version:
```

ow Traffic switched to Version 2