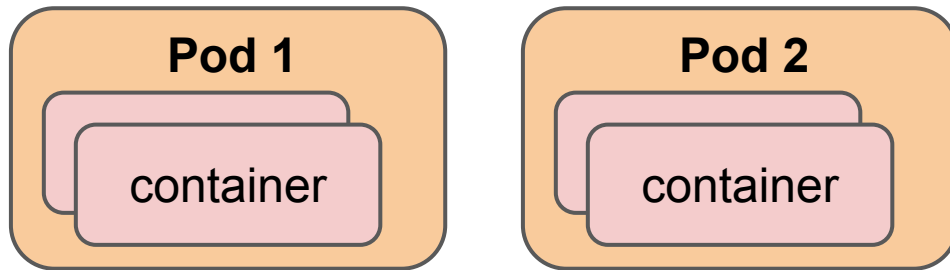


# The K8s Abstraction

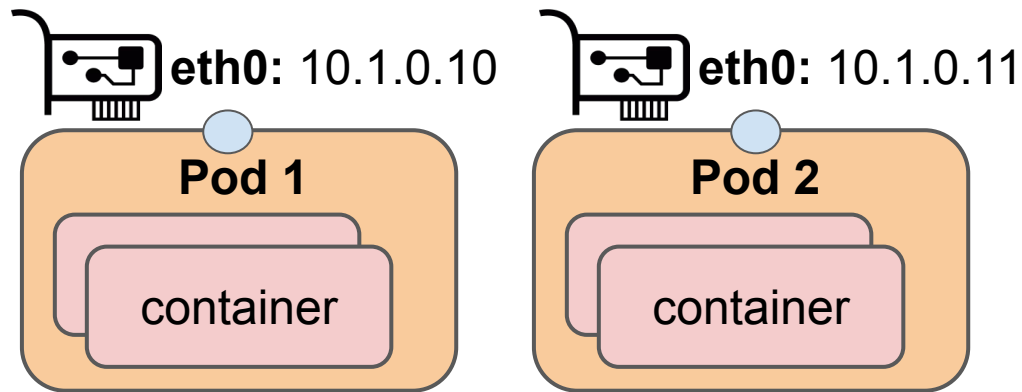
<https://github.com/paulopez78/k8s-abstraction>

**Abstractions = Resources**

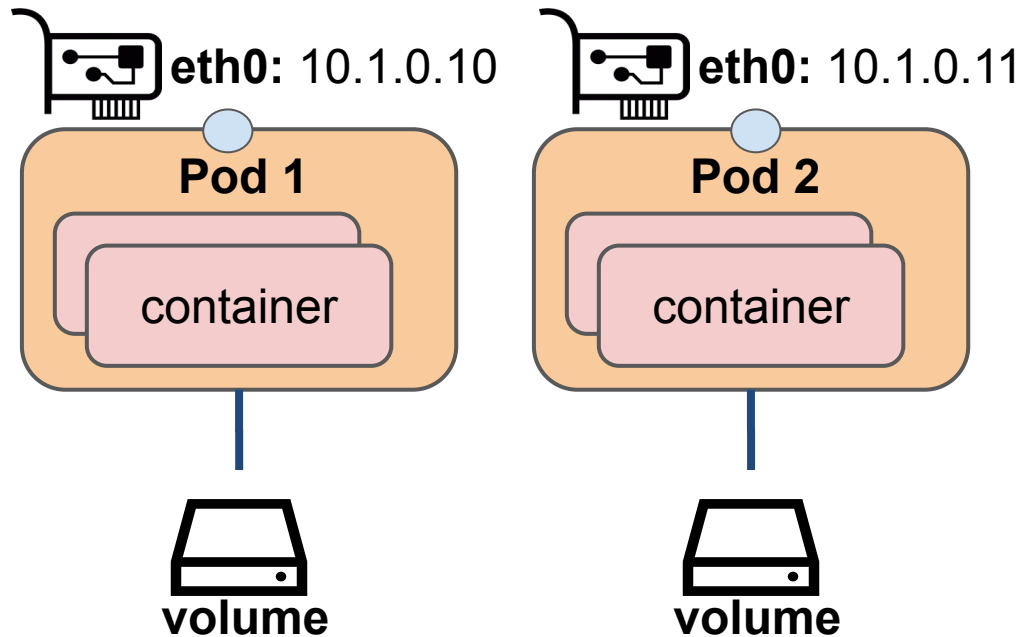
# Resources = Abstractions



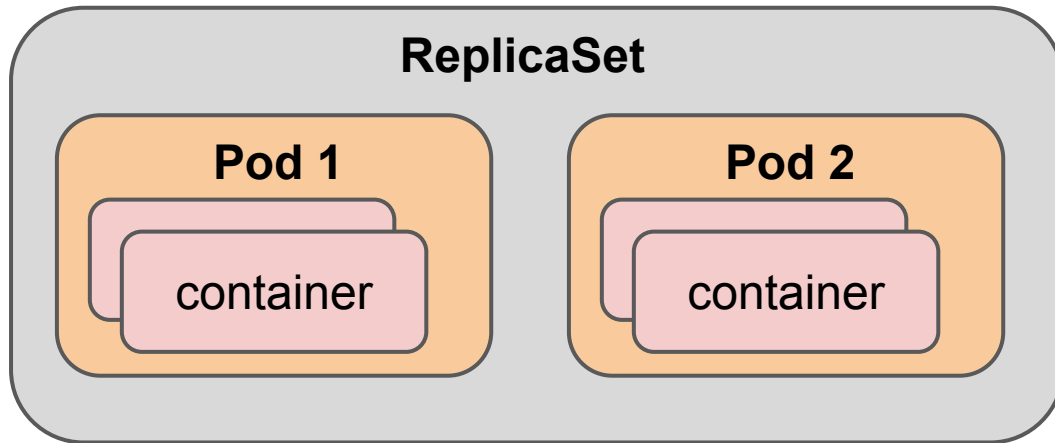
# Resources = Abstractions



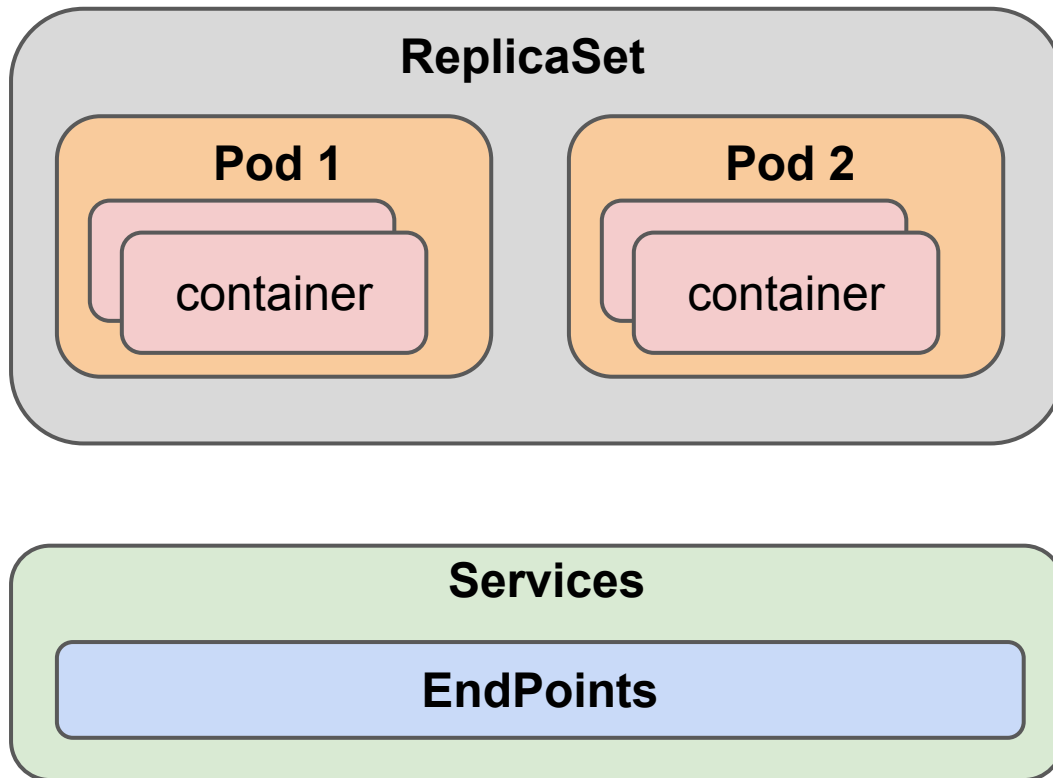
# Resources = Abstractions



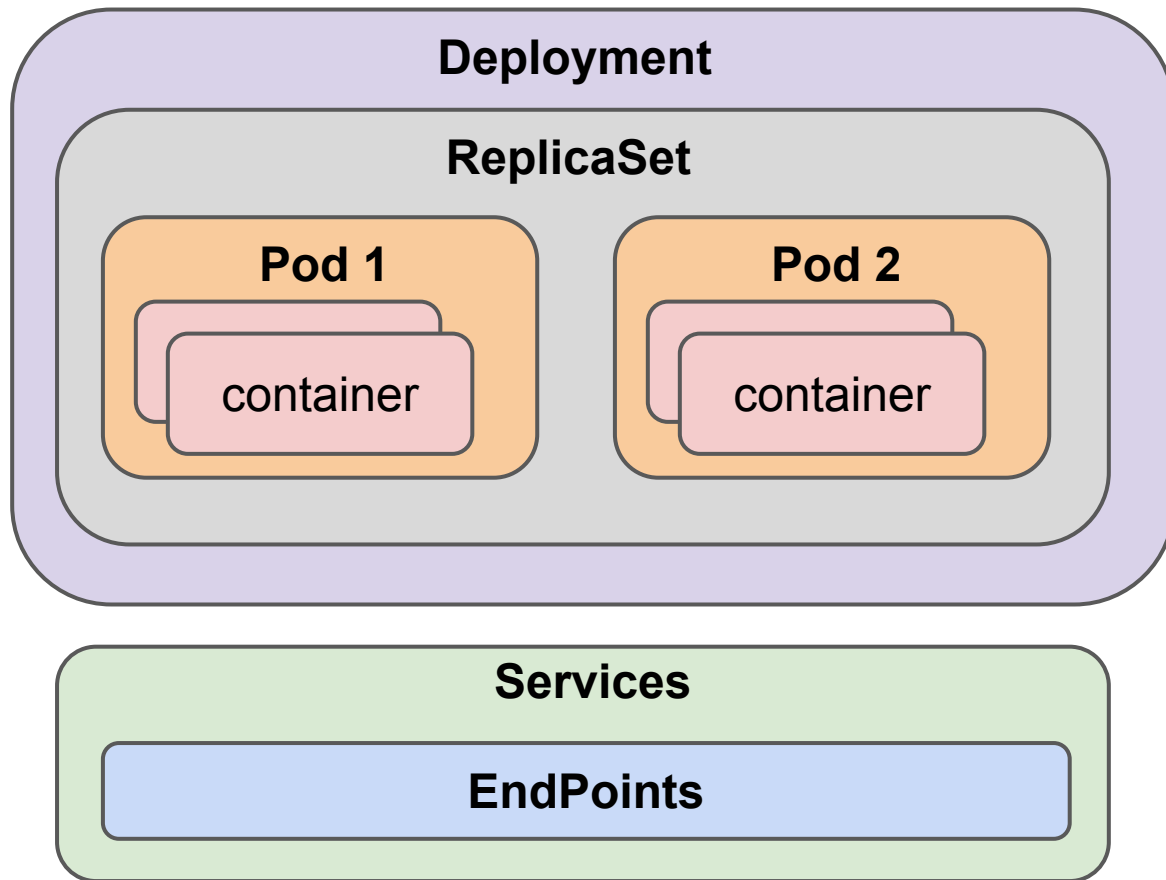
# Resources = Abstractions



# Resources = Abstractions



# Resources = Abstractions

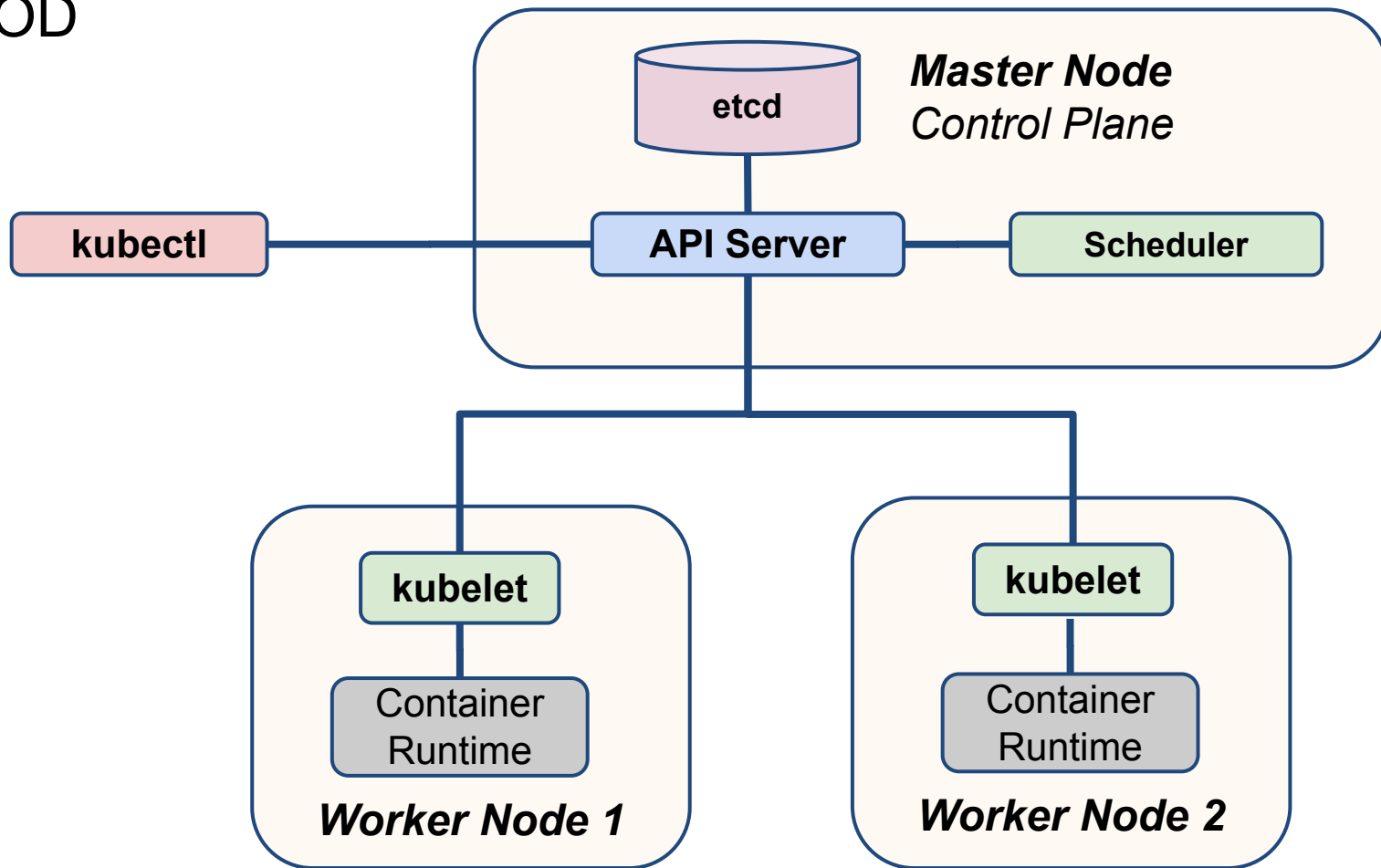




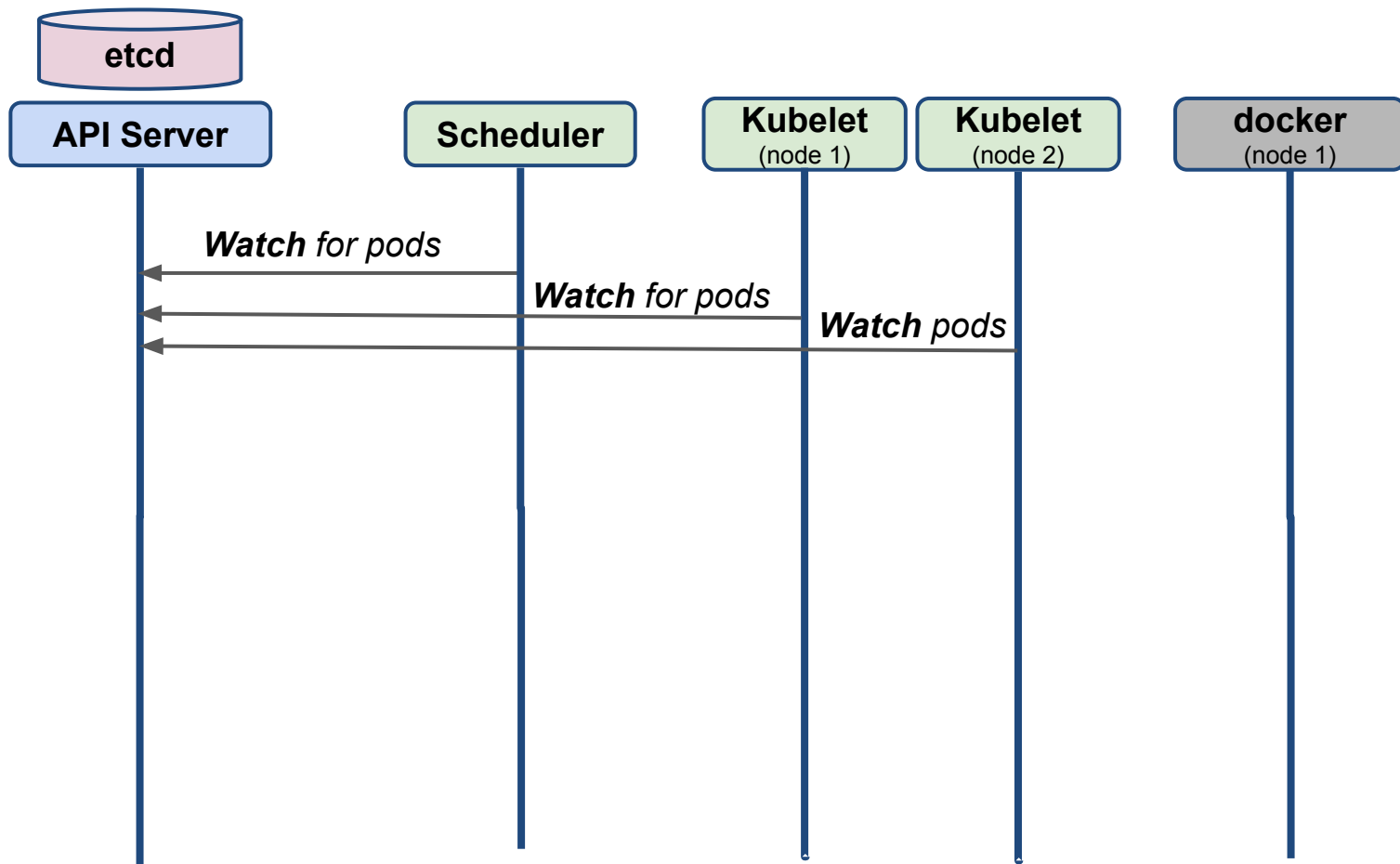
# POD

```
apiVersion: v1
kind: Pod
metadata:
  name: votingapp
  labels:
    app: votingapp
spec:
  containers:
    - image: paulopez/votingapp:0.1
      name: votingapp
```

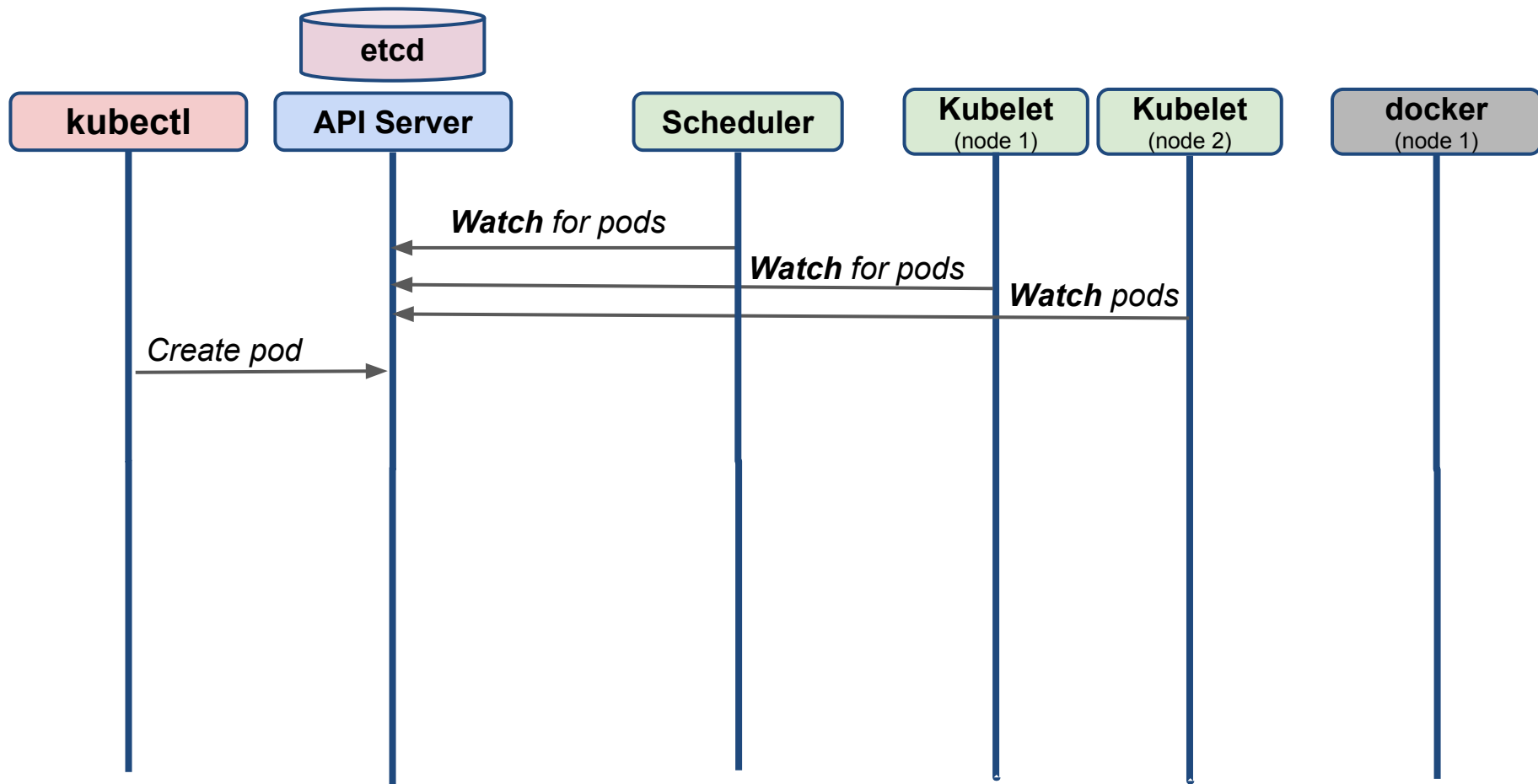
# POD



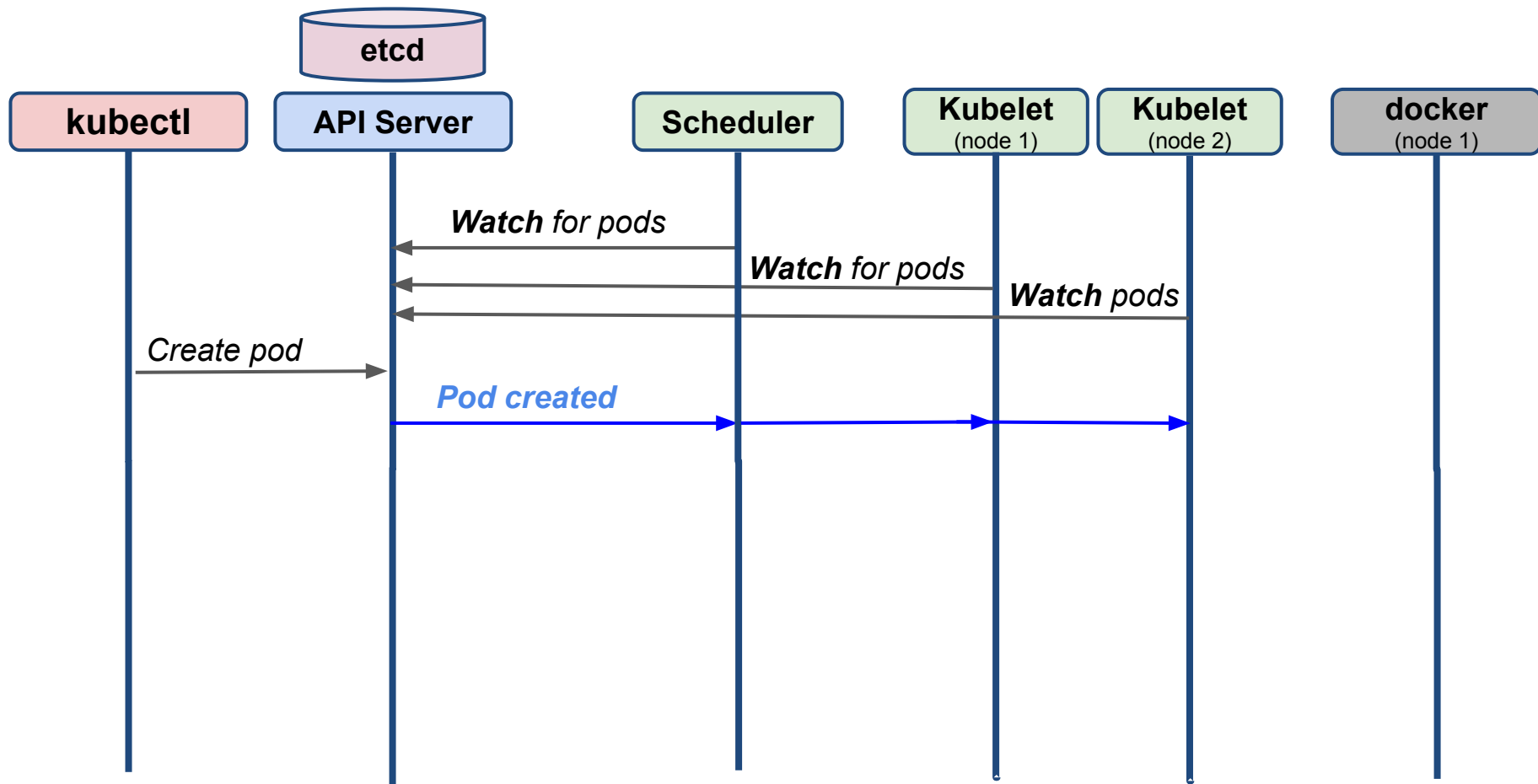
# POD: creation workflow



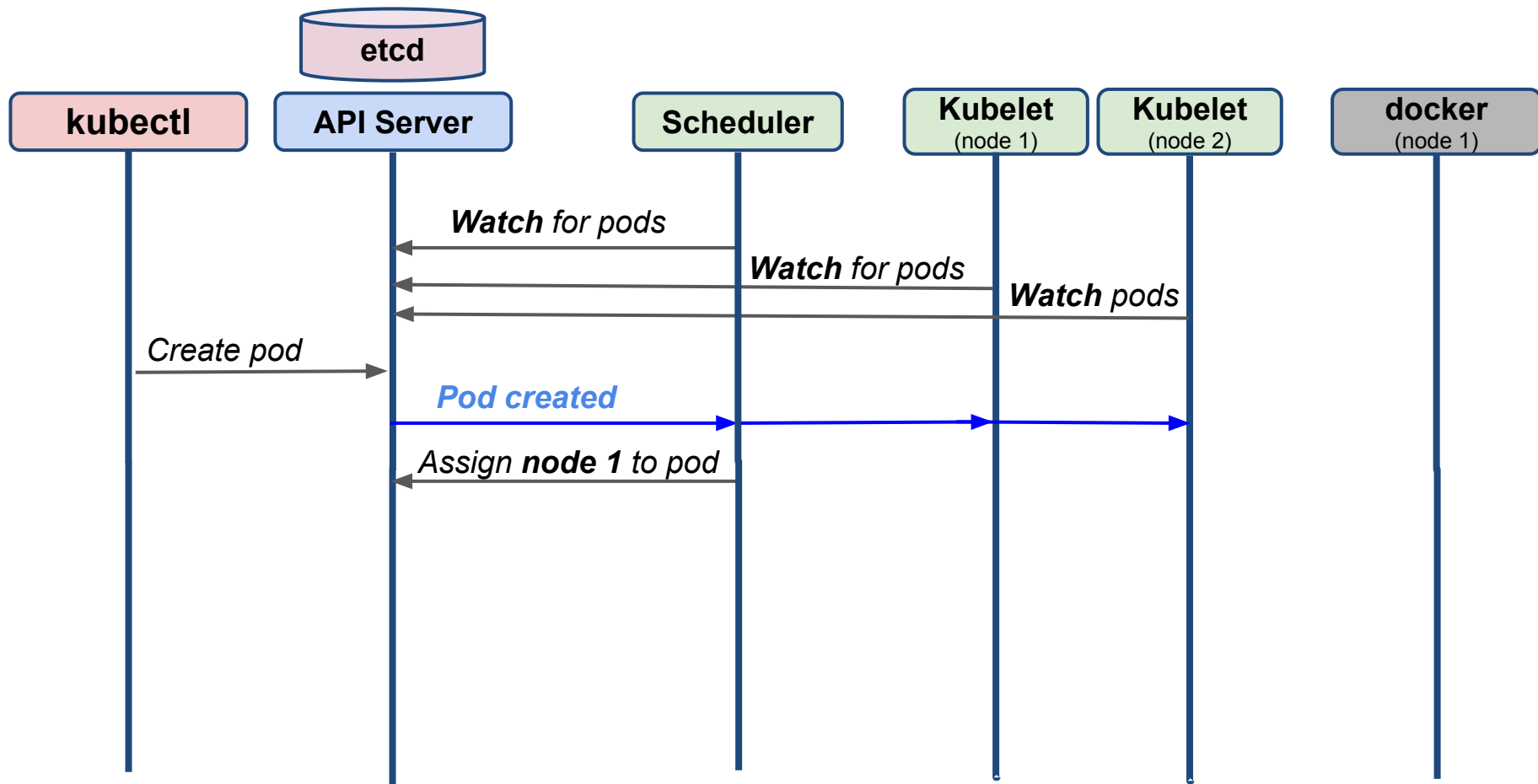
# POD: creation workflow



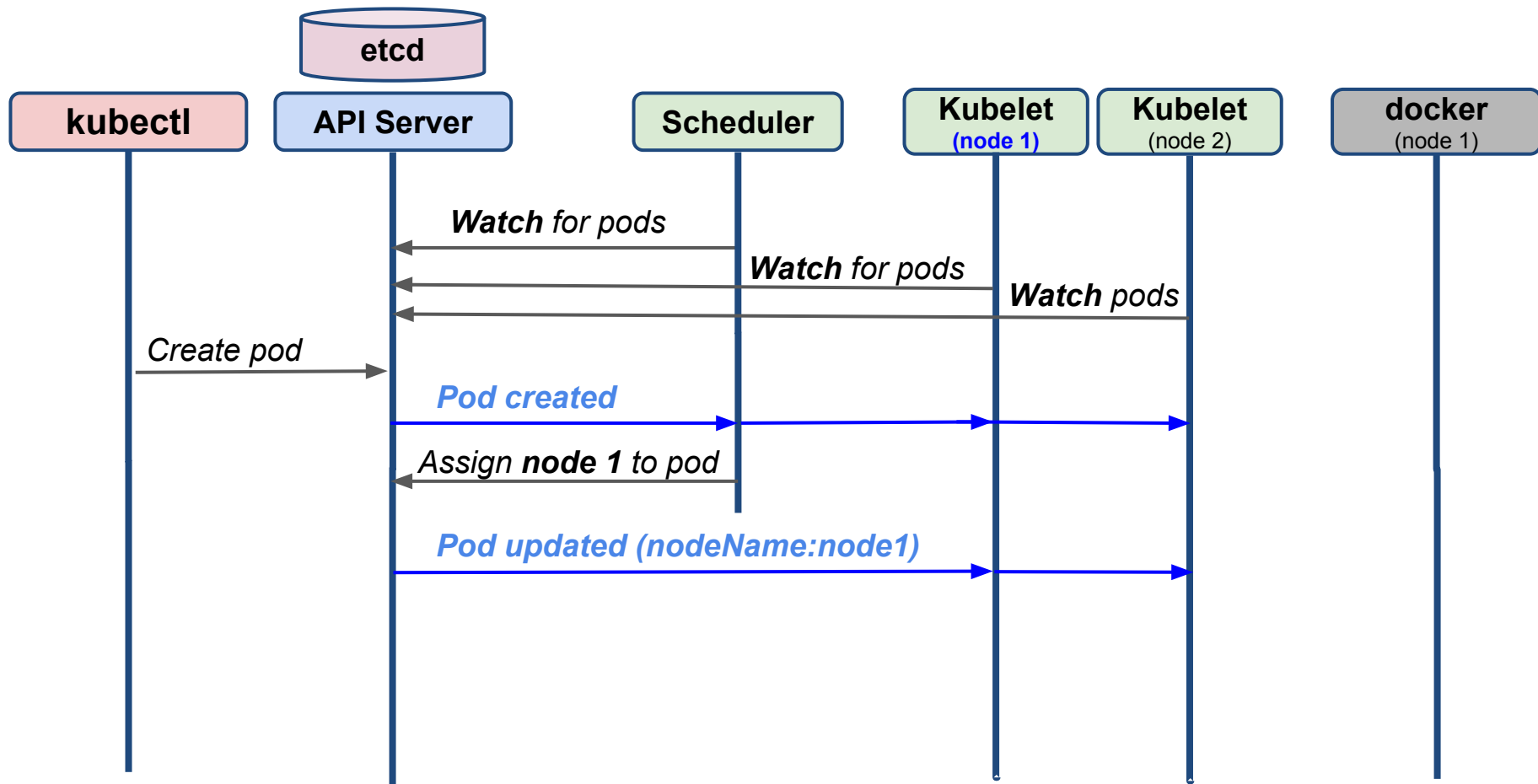
# POD: creation workflow



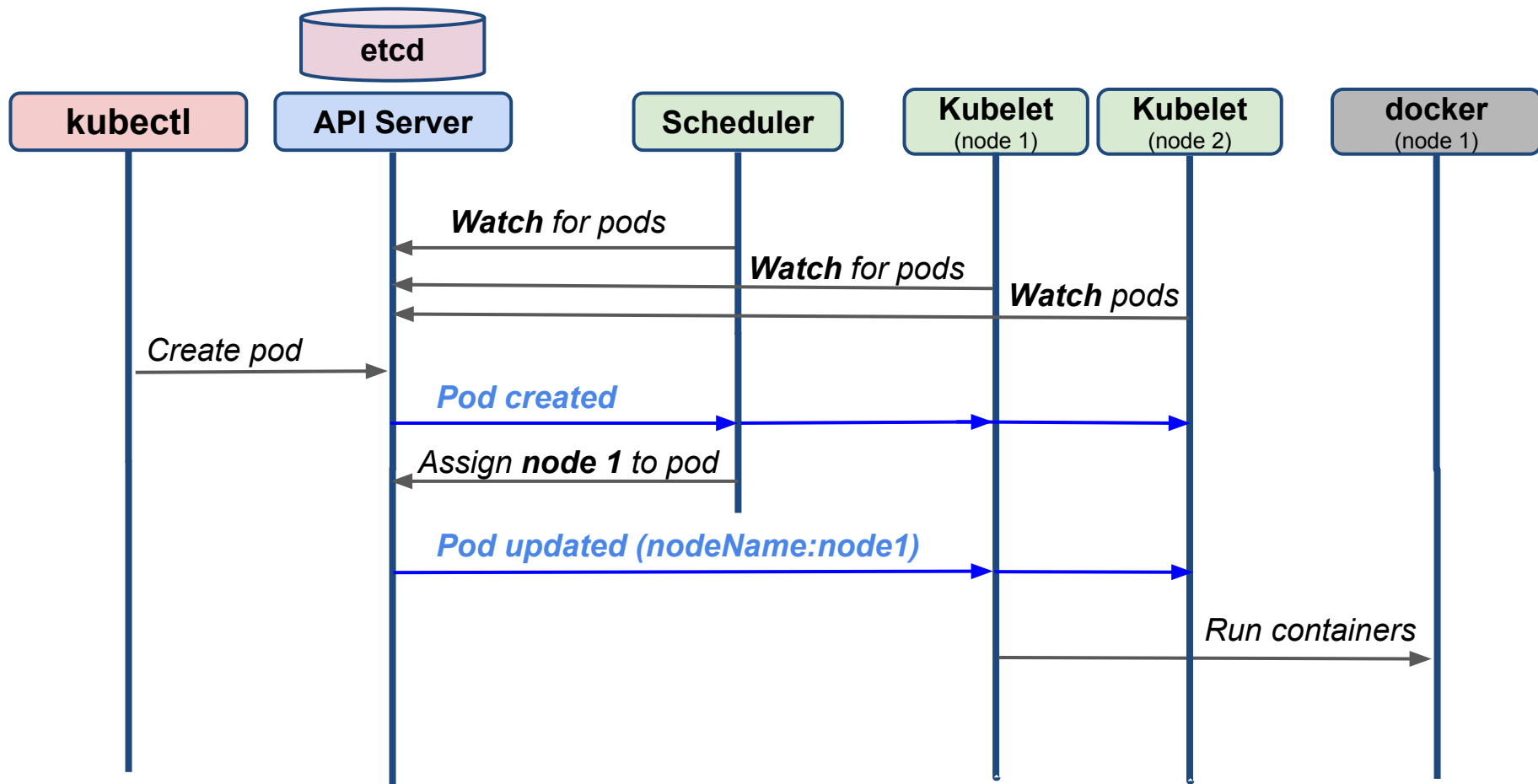
# POD: creation workflow



# POD: creation workflow

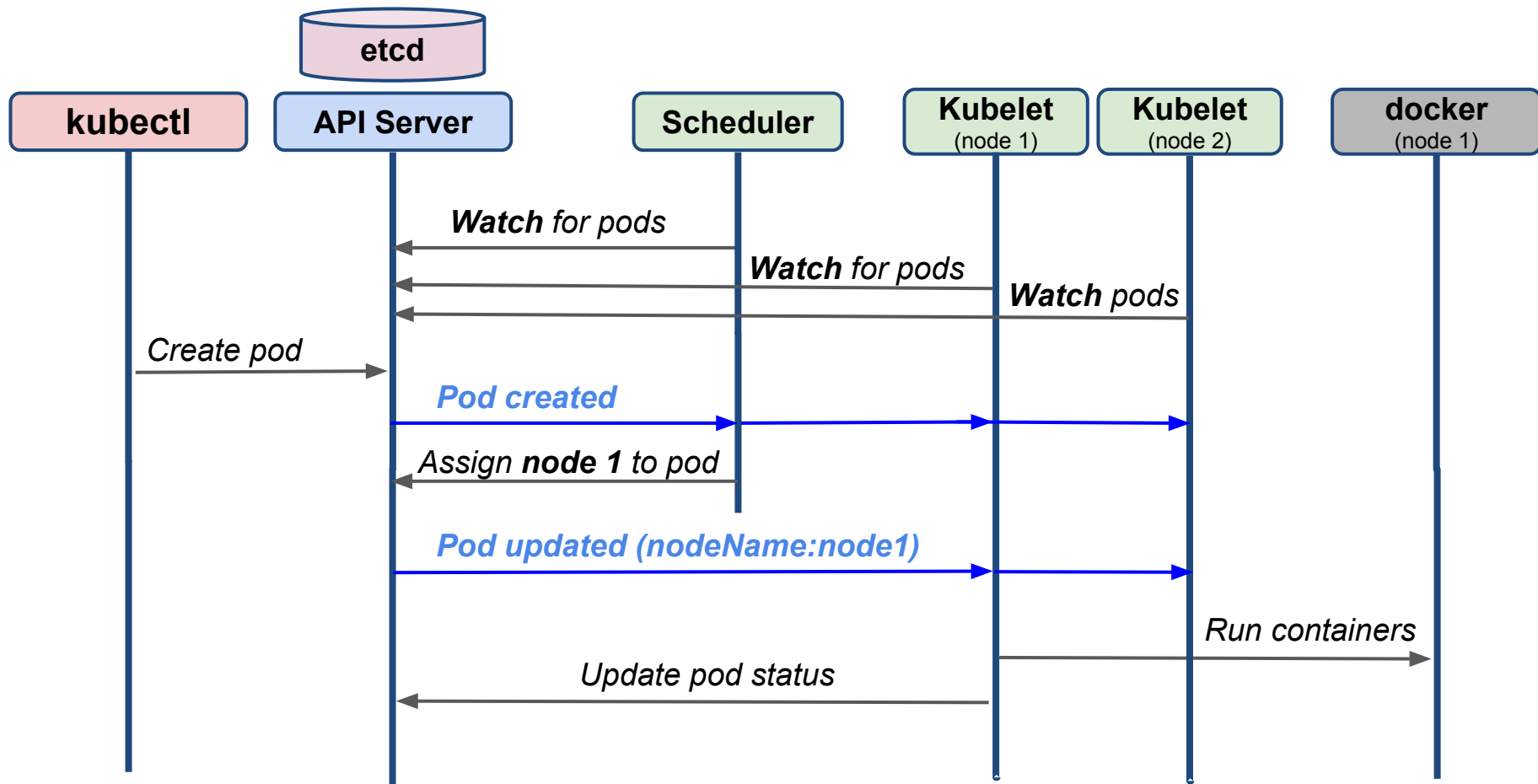


# POD: creation workflow

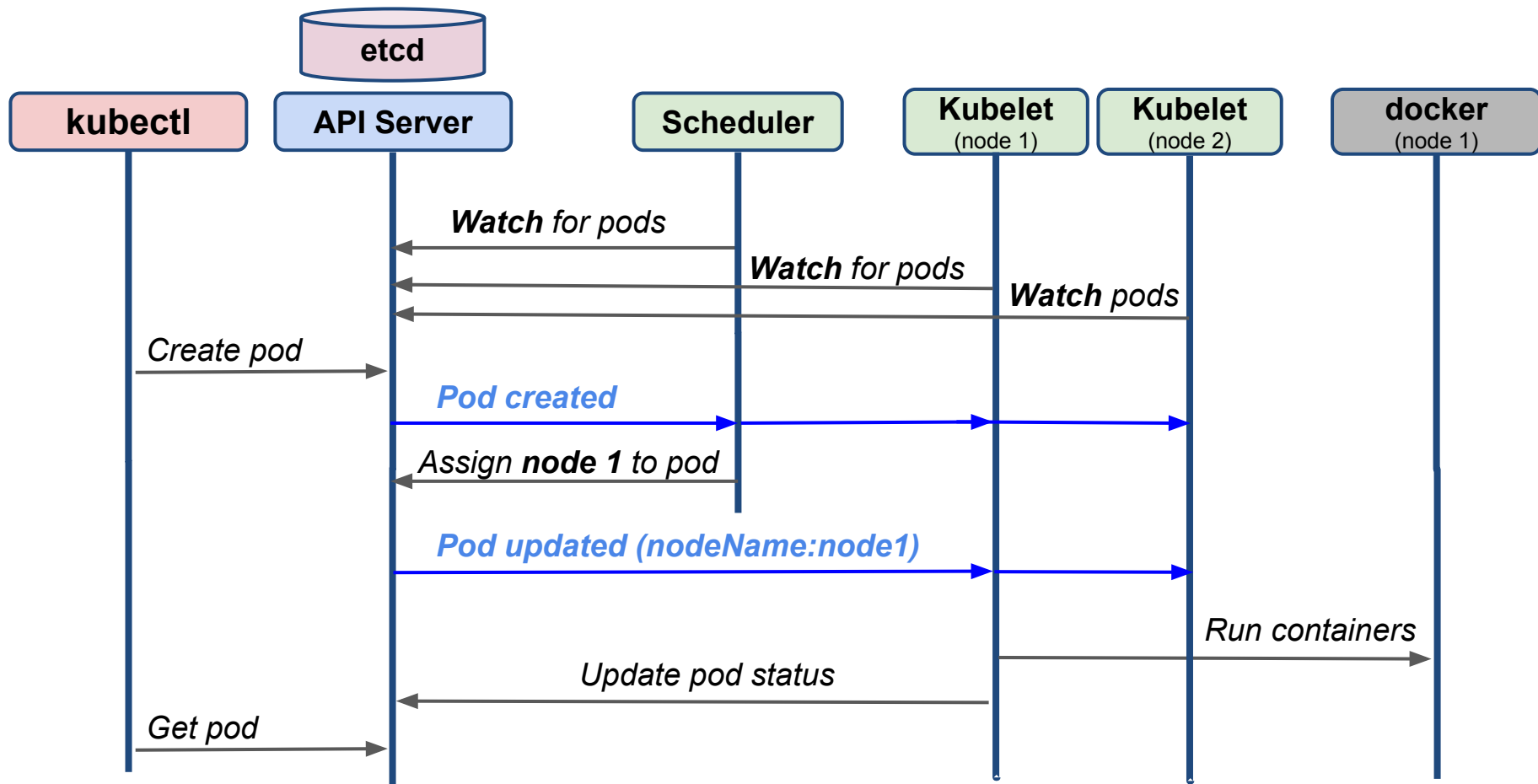




# POD: creation workflow



# POD: creation workflow



# REPLICASET

```
apiVersion: apps/v1
kind: ReplicaSet
metadata:
  name: votingapp
spec:
  replicas: 3
  selector:
    matchLabels:
      app: votingapp
  template:
    metadata:
      labels:
        app: votingapp
    spec:
      containers:
        - image: paulopez/votingapp:0.1
          name: votingapp
```

# REPLICASET

```
apiVersion: apps/v1
kind: ReplicaSet
metadata:
  name: votingapp
spec:
  replicas: 3
  selector:
    matchLabels:
      app: votingapp
  template:
```

```
    metadata:
      labels:
        app: votingapp
    spec:
      containers:
        - image: paulopez/votingapp:0.1
          name: votingapp
```

**POD**

# REPLICASET

```
apiVersion: apps/v1
kind: ReplicaSet
metadata:
  name: votingapp
```

```
spec:
```

```
  replicas: 3
```

```
  selector:
```

```
    matchLabels:
```

```
      app: votingapp
```

```
  template:
```

```
    metadata:
```

```
      labels:
```

```
        app: votingapp
```

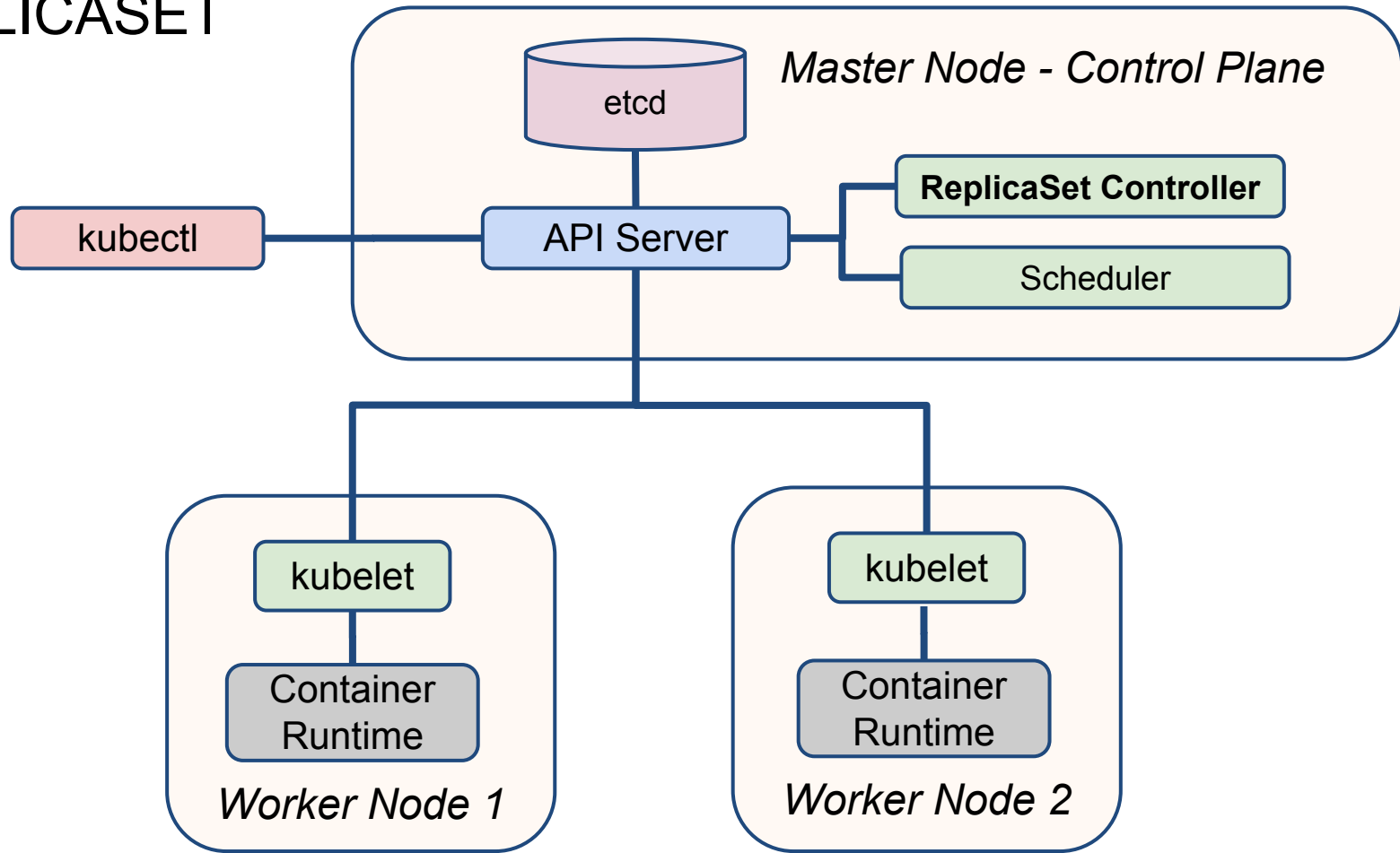
```
    spec:
```

```
      containers:
```

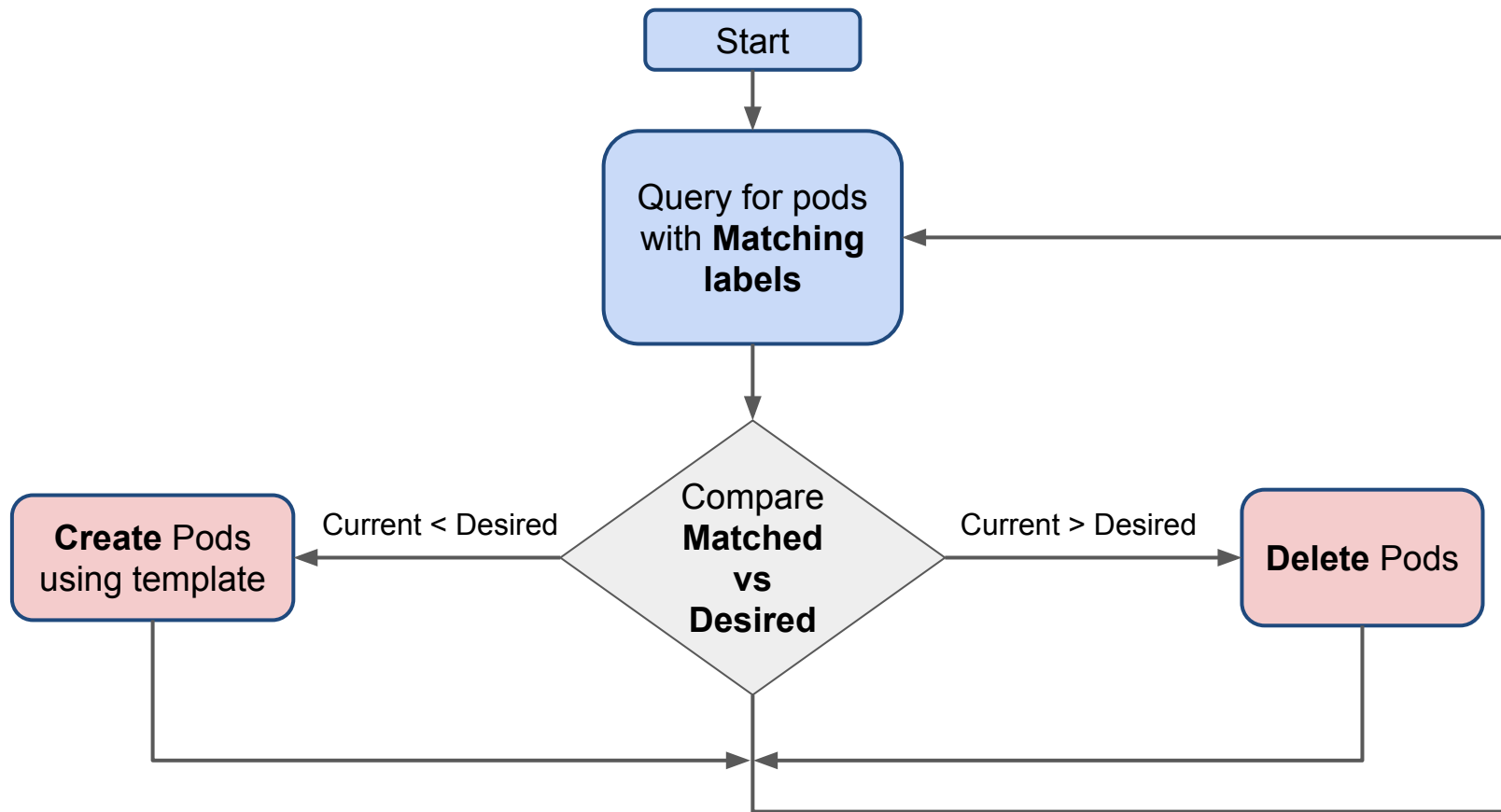
- image: paulopez/votingapp:0.1
- name: votingapp

**POD**

# REPLICASET



# REPLICASET CONTROLLER: Reconciliation Loop



# SERVICE

```
apiVersion: v1
kind: Service
metadata:
  name: votingapp
spec:
  ports:
  - port: 8080
    targetPort: 5000
  selector:
    app: votingapp
  type: ClusterIP
```



# SERVICE

```
apiVersion: v1
kind: Service
metadata:
  name: votingapp
spec:
  ports:
    - port: 8080
      targetPort: 5000
  selector:
    app: votingapp
  type: ClusterIP
```

```
apiVersion: apps/v1
kind: ReplicaSet
metadata:
```

```
  name: votingapp
```

```
spec:
```

```
  replicas: 3
```

```
  selector:
```

```
    matchLabels:
```

```
      app: votingapp
```

```
  template:
```

```
    metadata:
```

```
      labels:
```

**POD**

```
        app: votingapp
```

```
    spec:
```

```
      containers:
```

```
        - image: paulopez/votingapp:0.1
          name: votingapp
```

# SERVICE

**Cluster IP**  
10.96.138.12:8080

**VotingApp Service**

app=votingapp

**eth0:10.244.1.2**

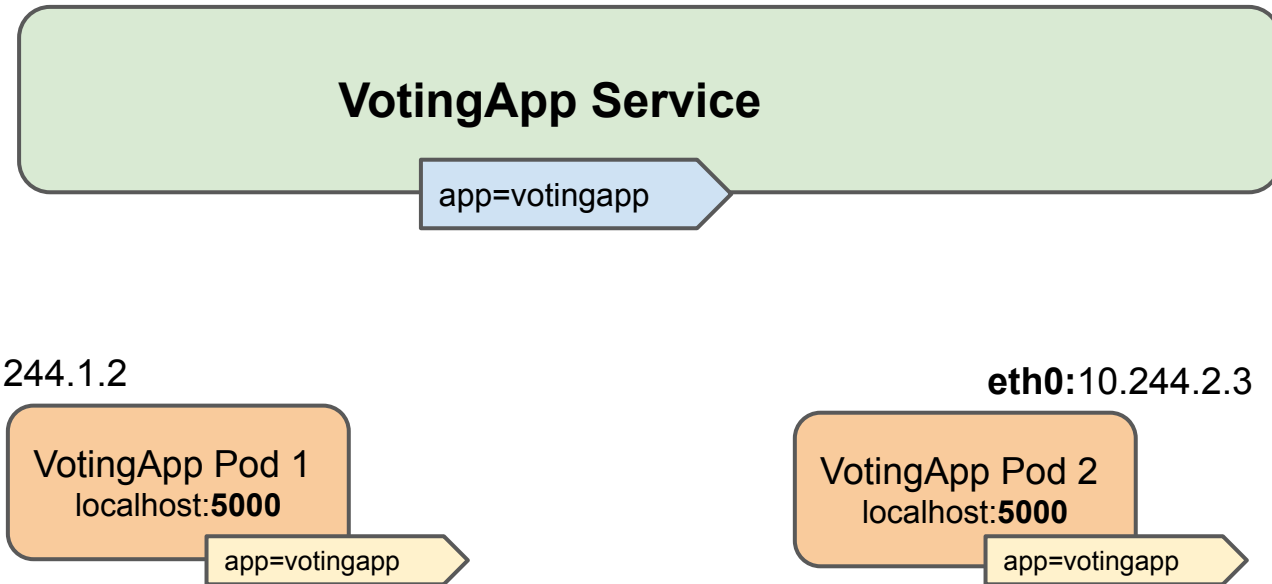
VotingApp Pod 1  
localhost:5000

app=votingapp

**eth0:10.244.2.3**

VotingApp Pod 2  
localhost:5000

app=votingapp



# SERVICE

**Cluster IP**  
10.96.138.12:8080

**VotingApp Service**

app=votingapp

**eth0:10.244.1.2**

VotingApp Pod 1  
localhost:5000

app=votingapp

**eth0:10.244.2.3**

VotingApp Pod 2  
localhost:5000

app=votingapp

# SERVICE

**Cluster IP**  
10.96.138.12:8080

**VotingApp Service**

app=votingapp

**eth0:10.244.1.2**

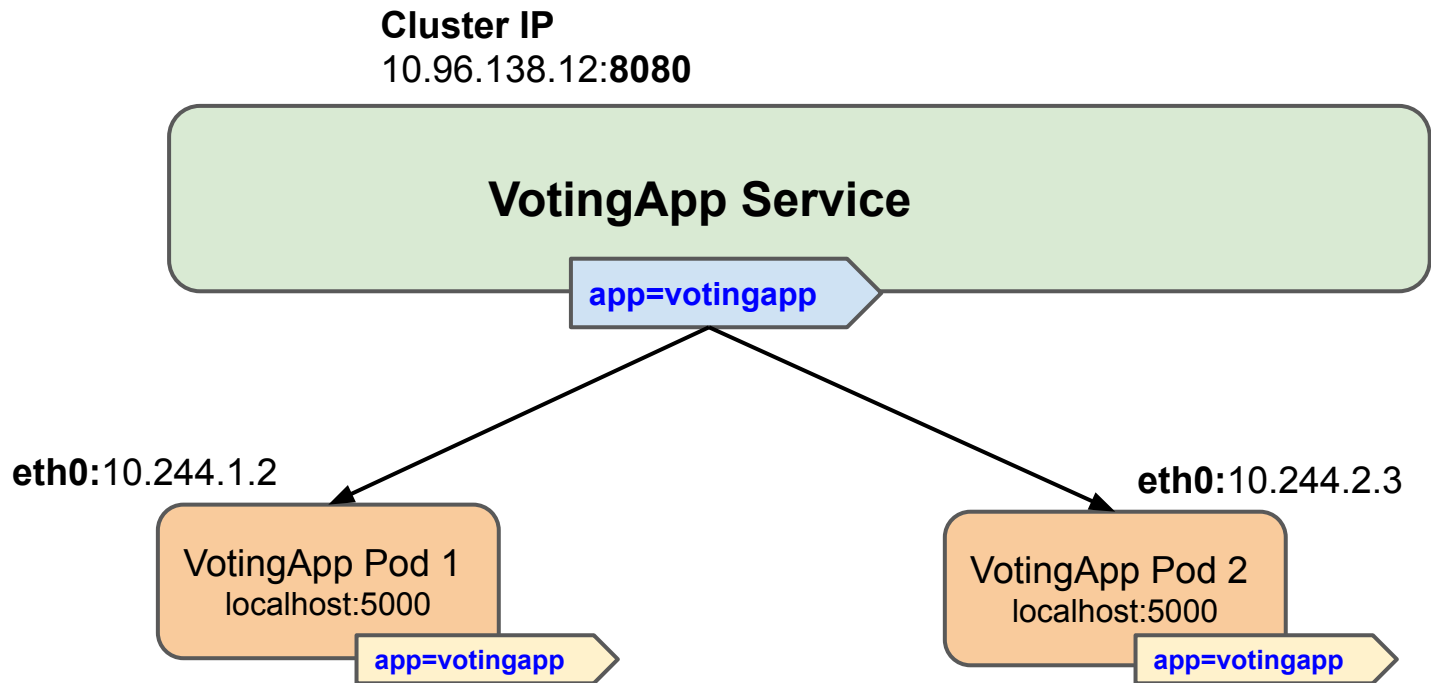
VotingApp Pod 1  
localhost:5000

app=votingapp

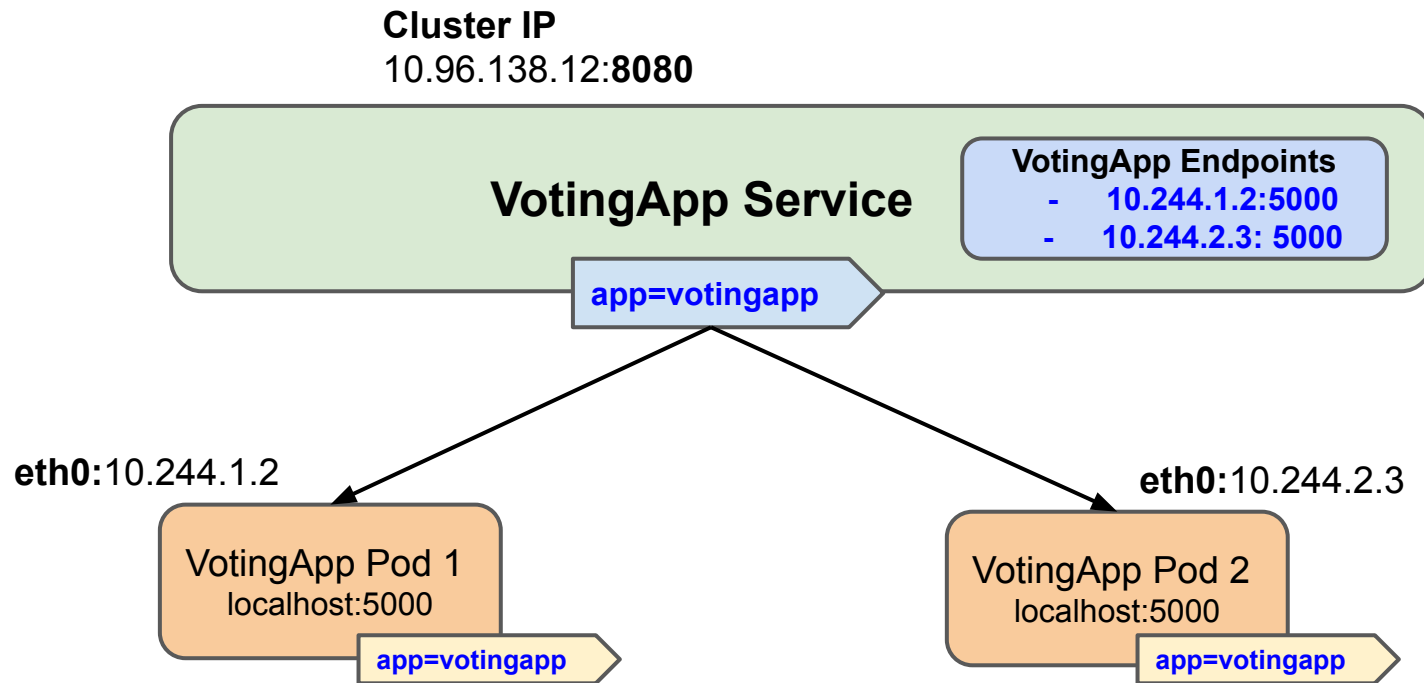
**eth0:10.244.2.3**

VotingApp Pod 2  
localhost:5000

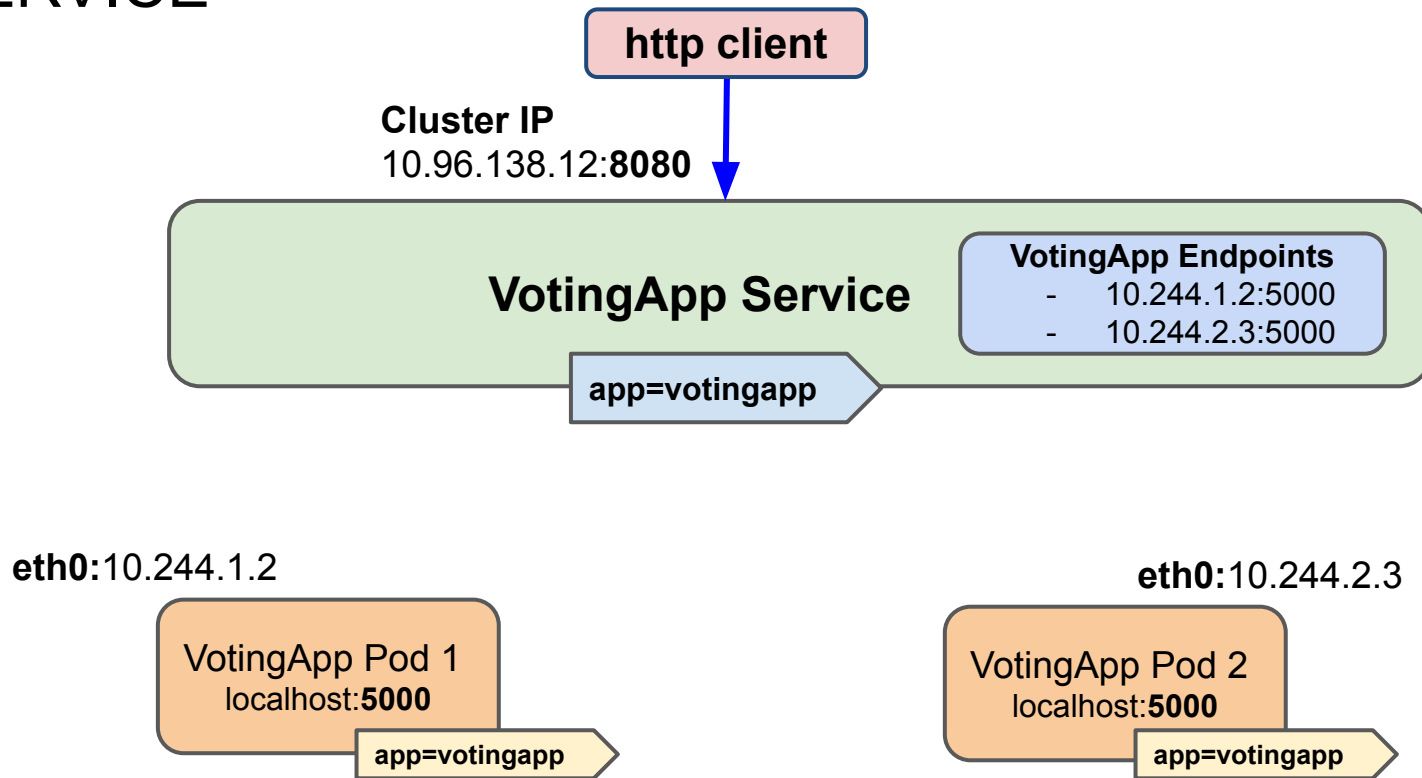
app=votingapp



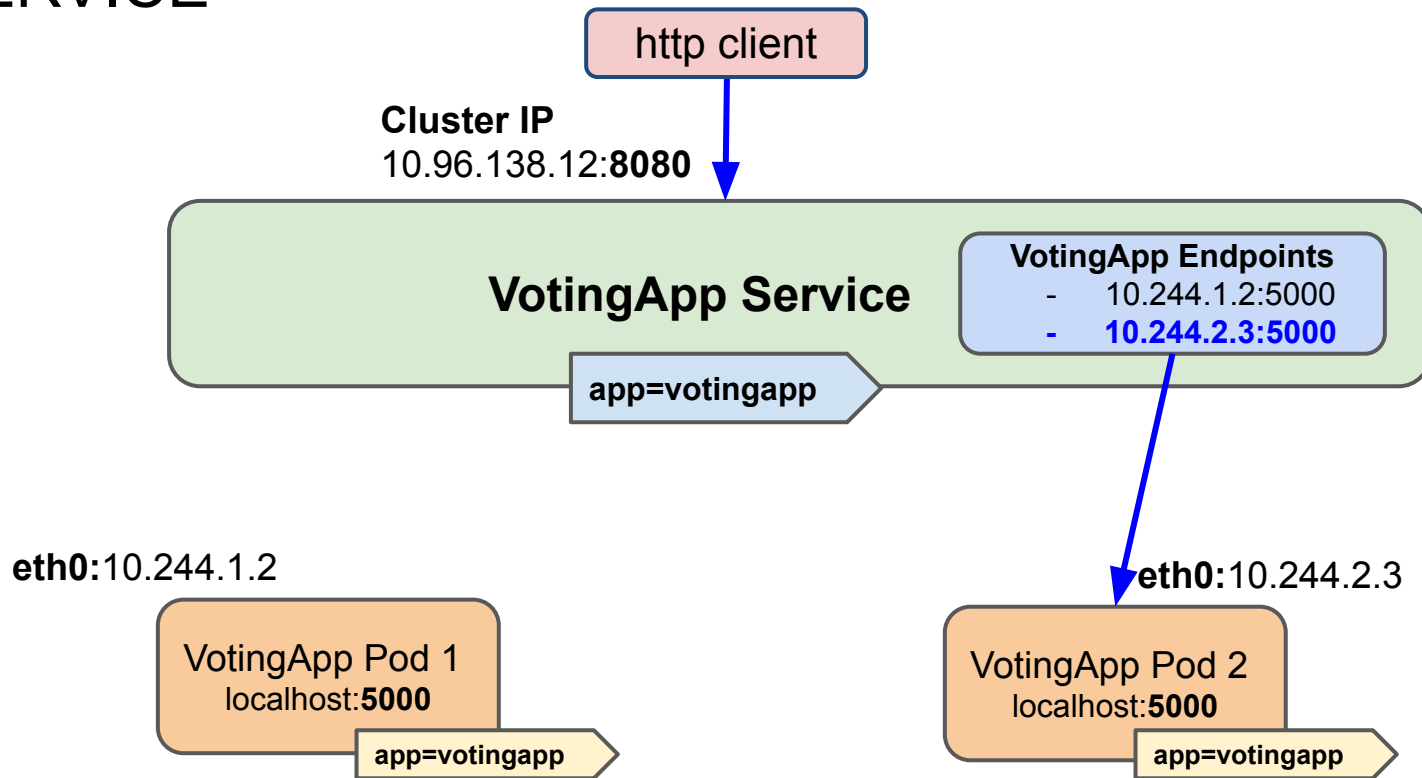
# SERVICE



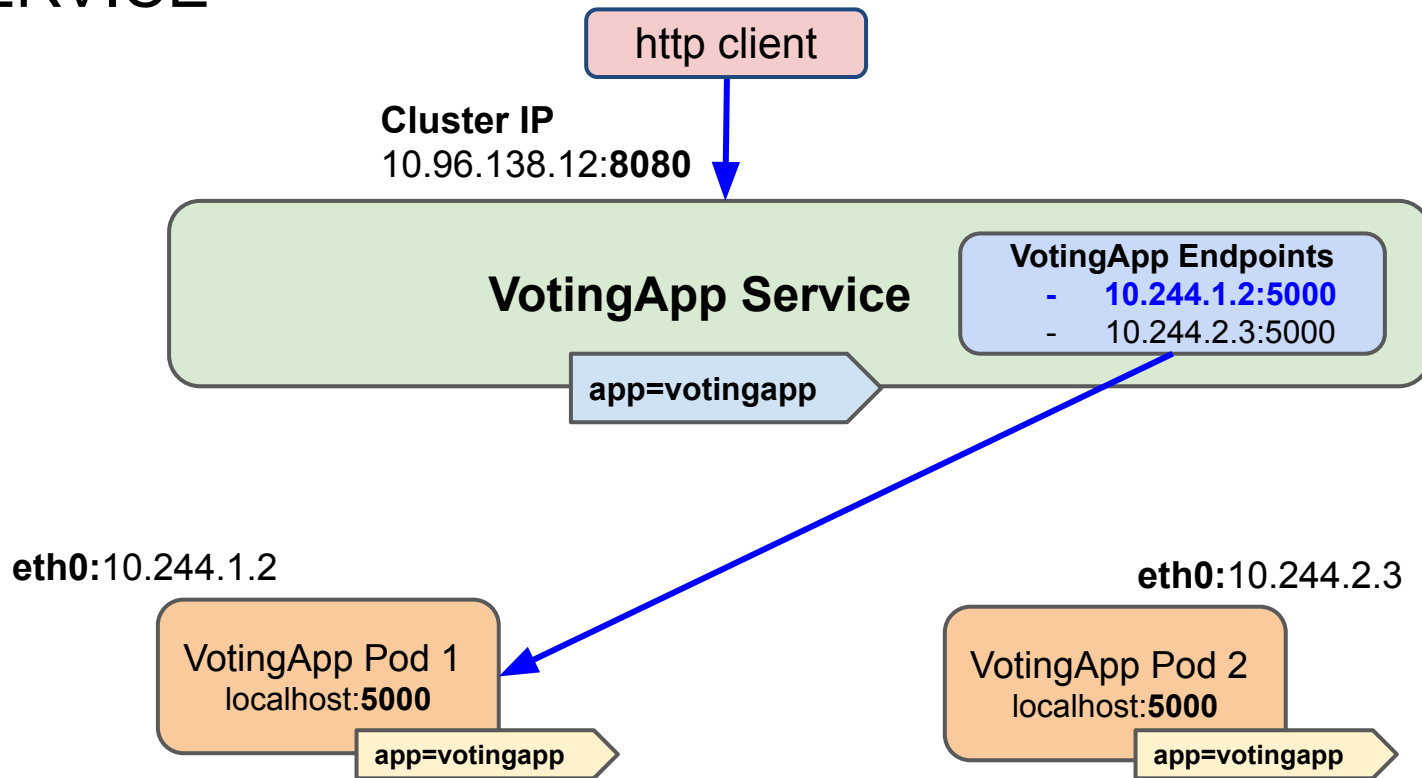
# SERVICE



# SERVICE

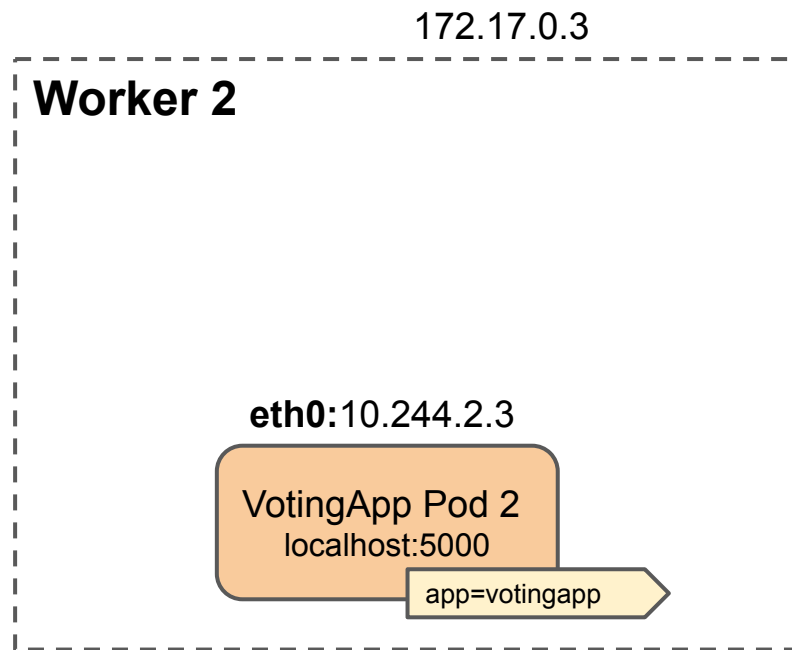
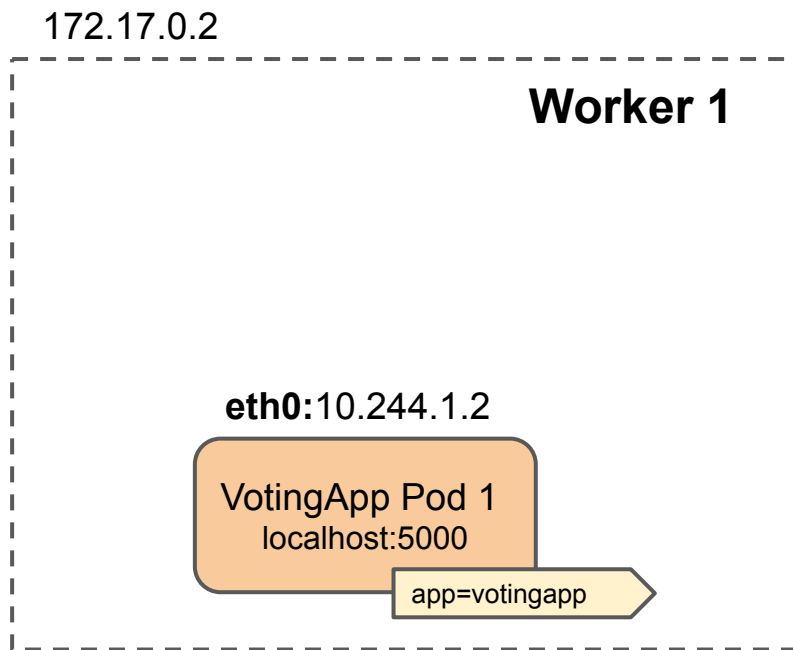


# SERVICE

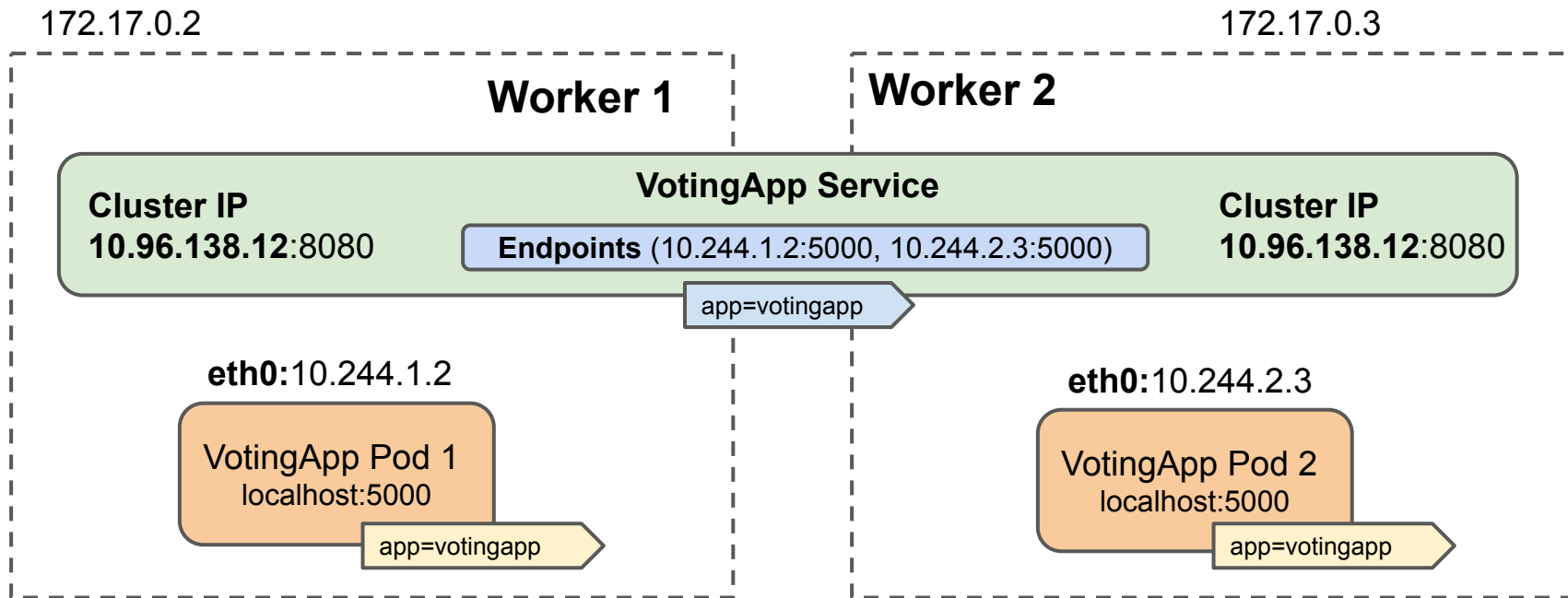




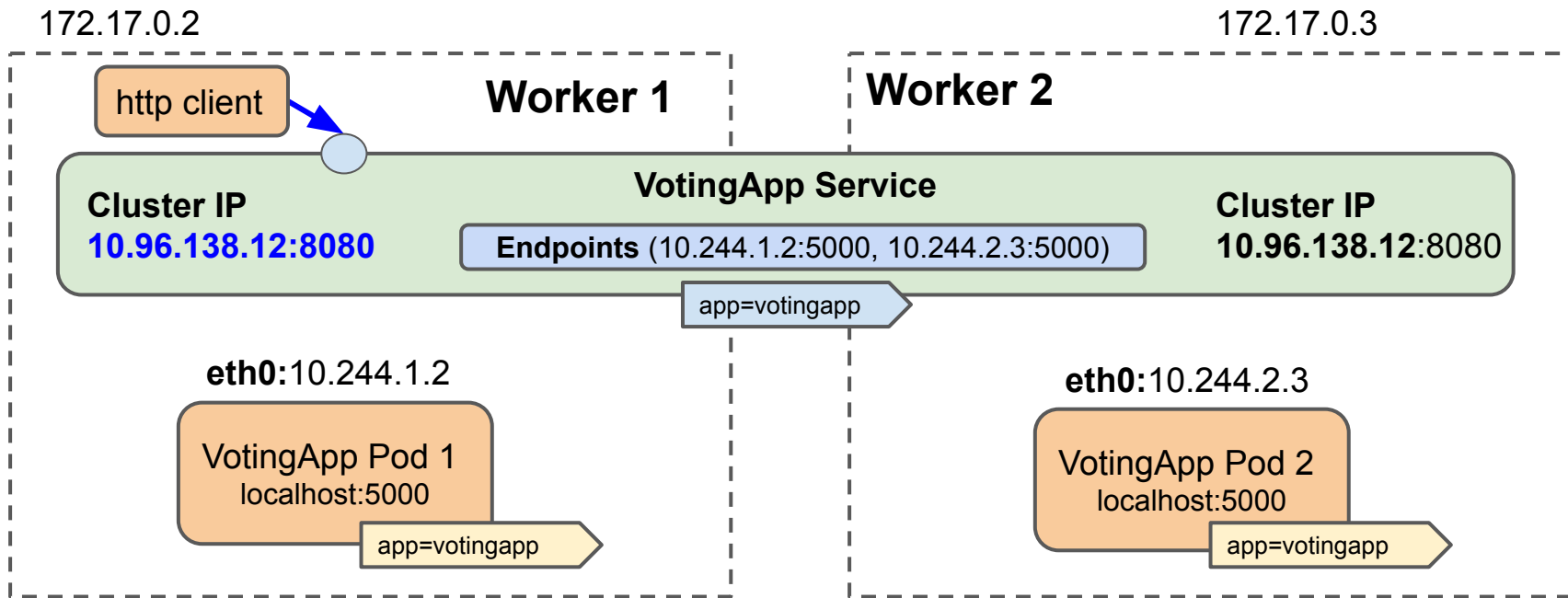
# SERVICE: CLUSTER IP



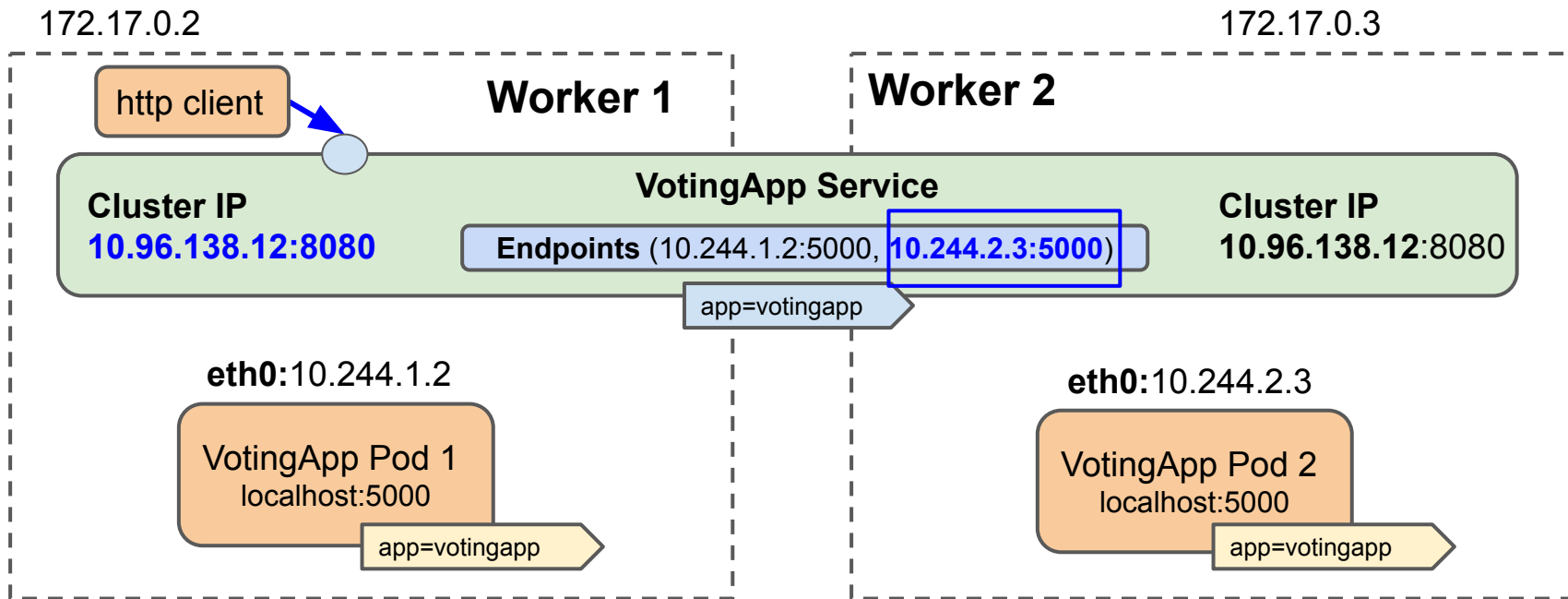
# SERVICE: CLUSTER IP



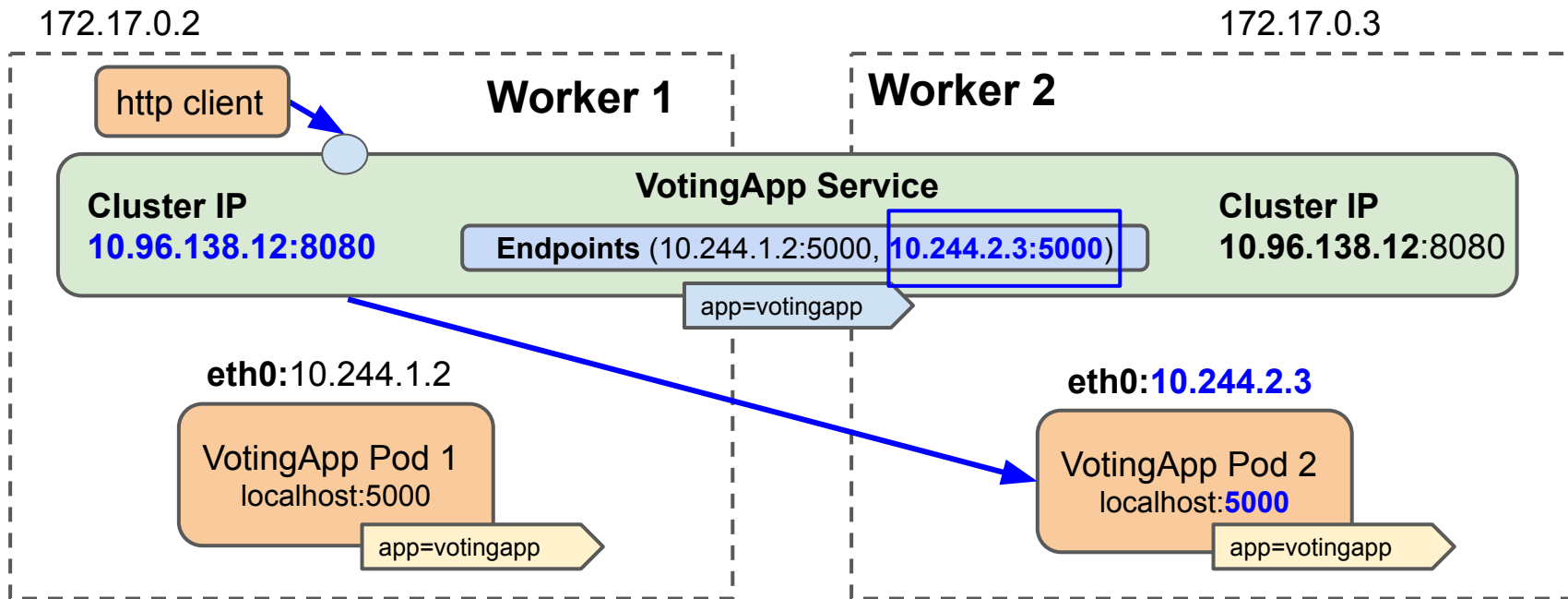
# SERVICE: CLUSTER IP



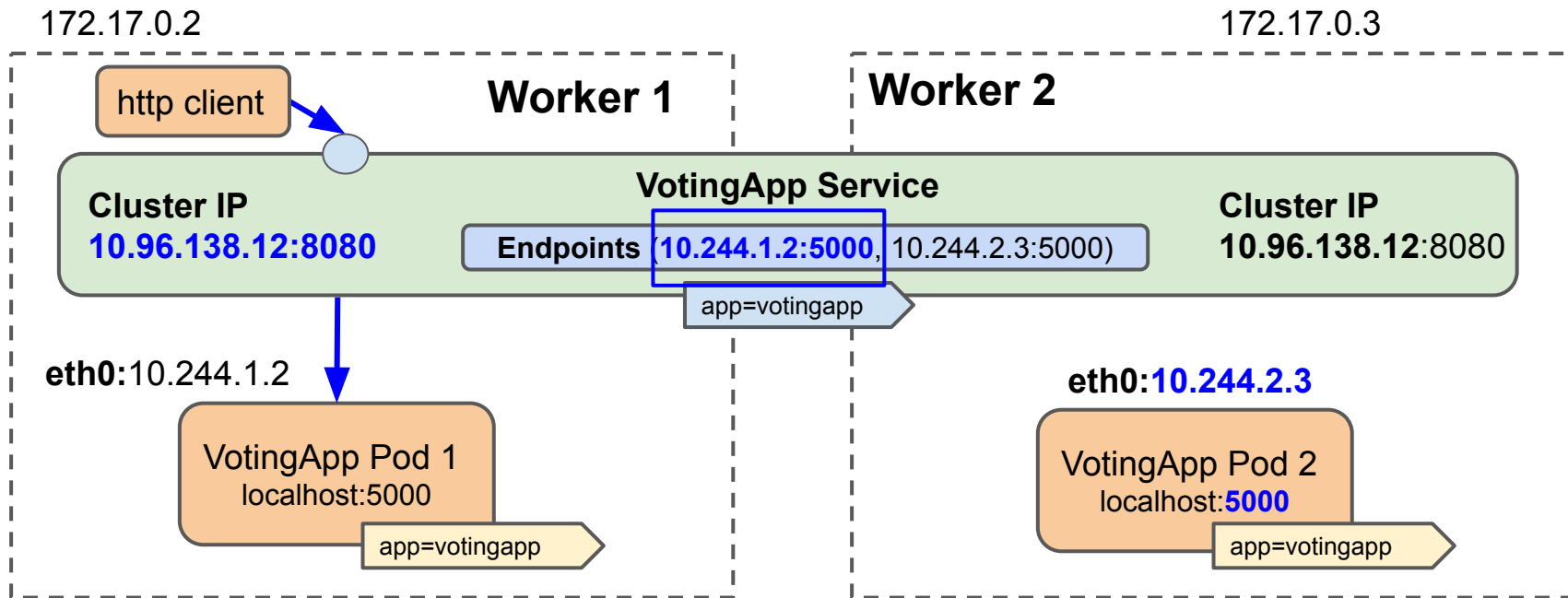
# SERVICE: CLUSTER IP



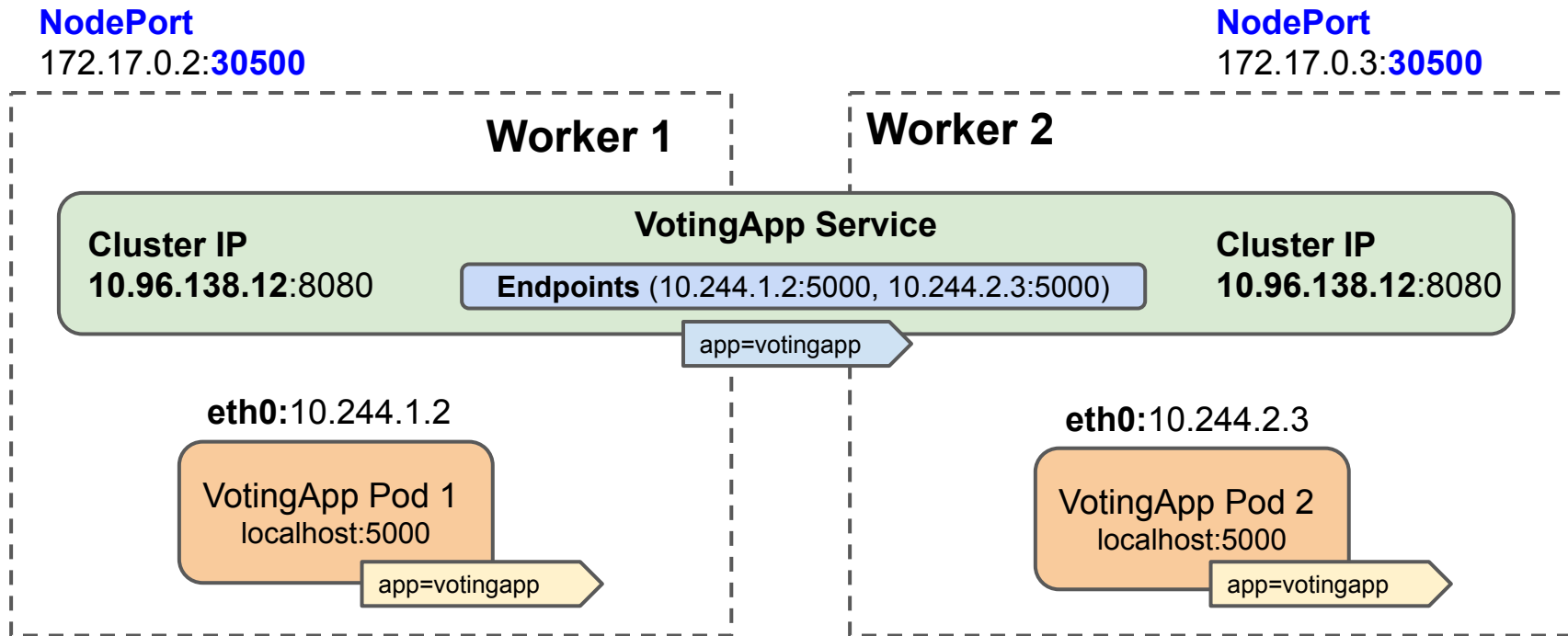
# SERVICE: CLUSTER IP



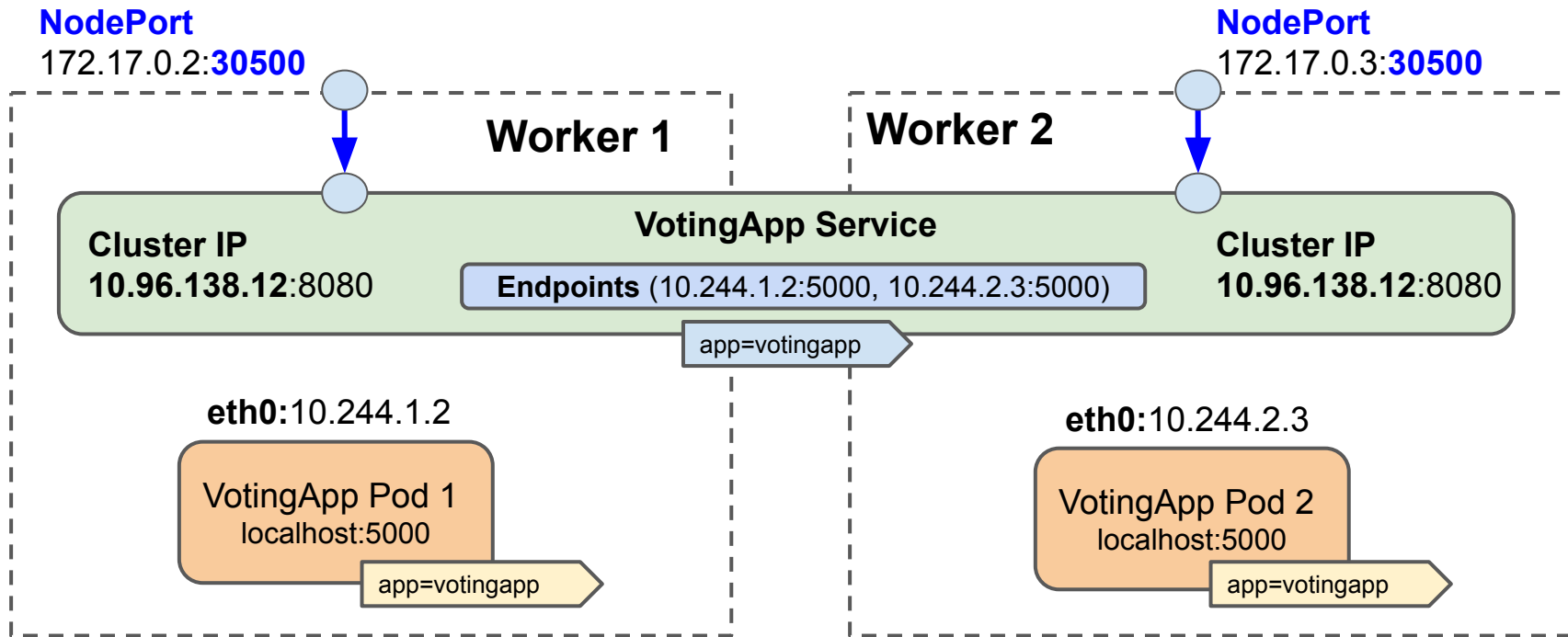
# SERVICE: CLUSTER IP



# SERVICE: NODEPORT

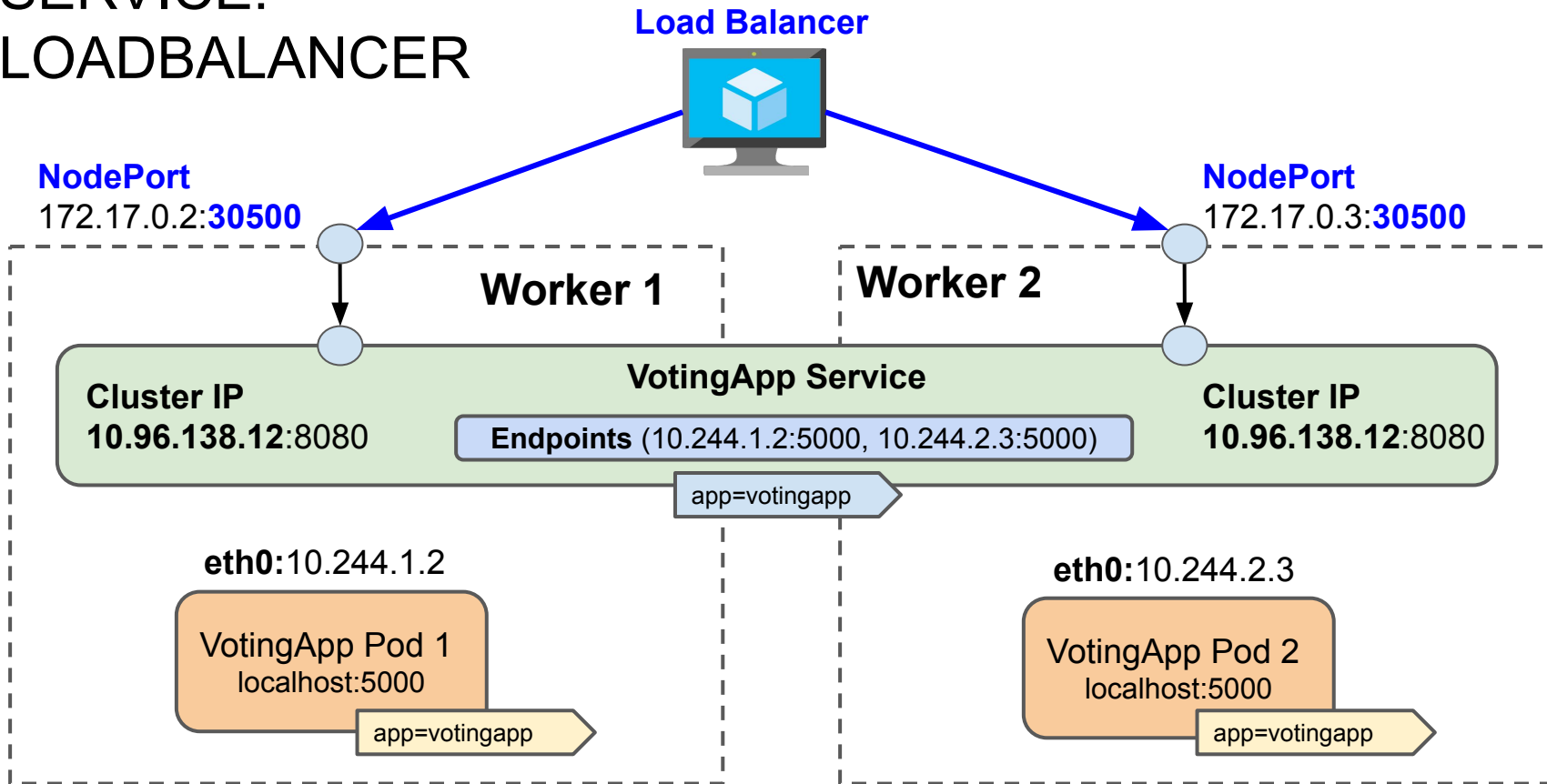


# SERVICE: NODEPORT

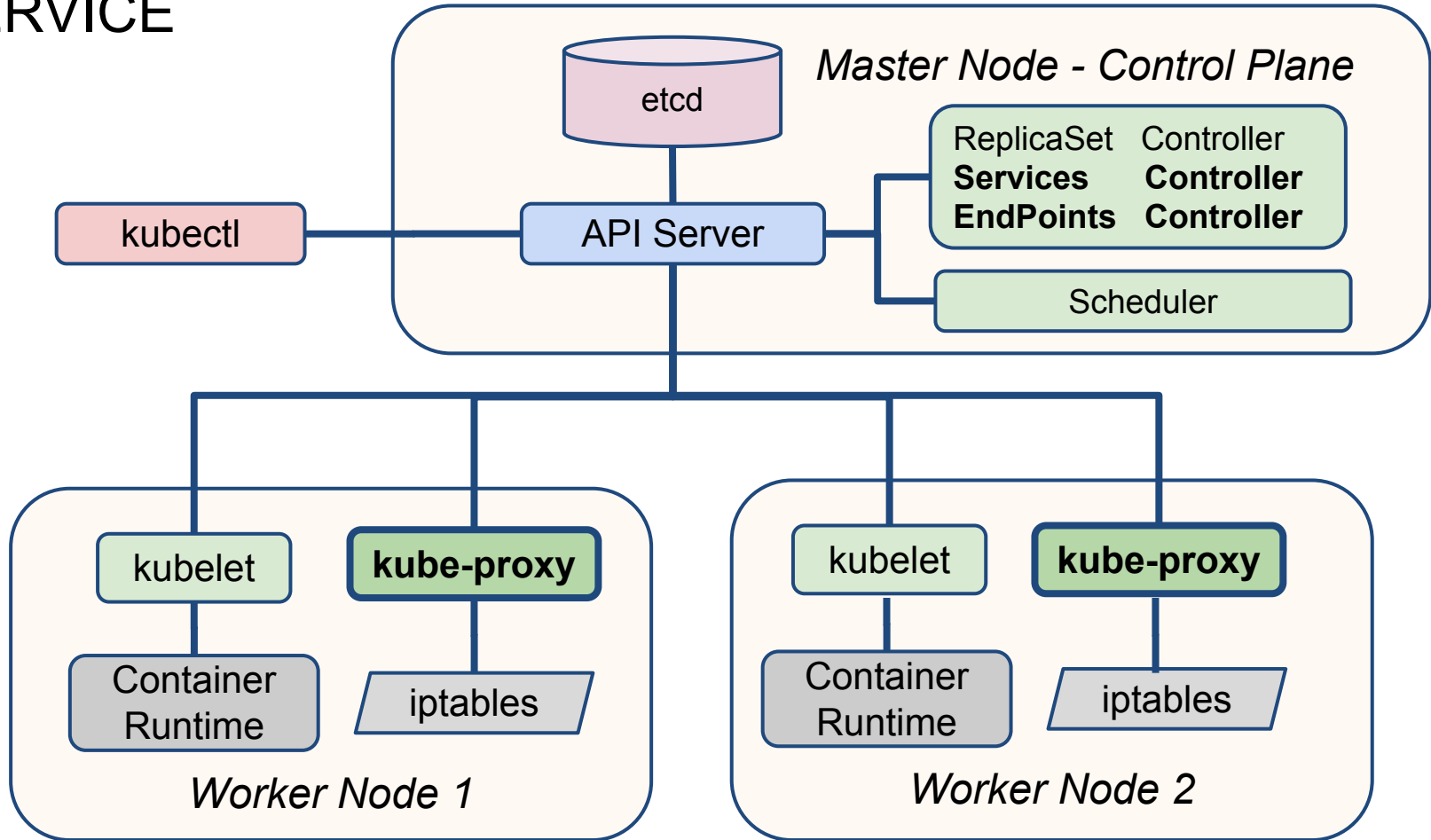




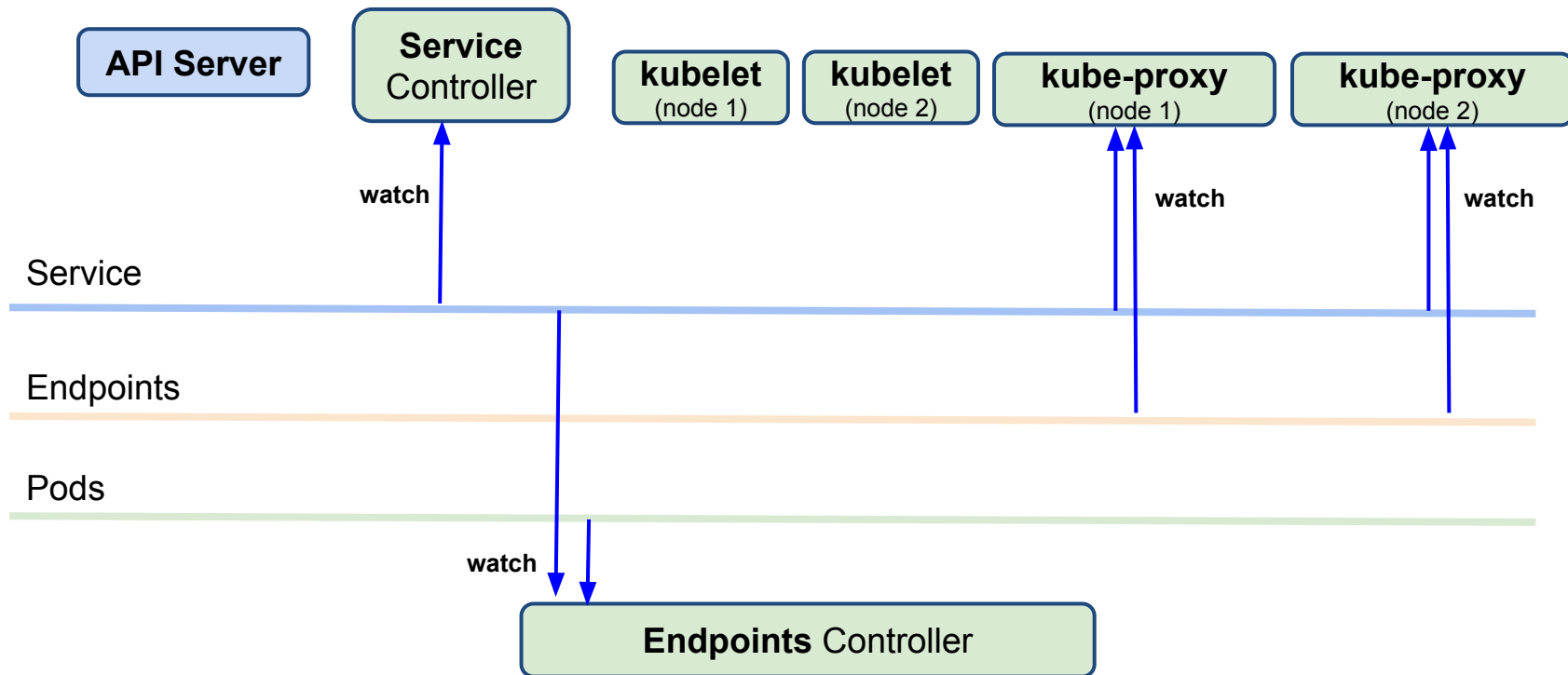
# SERVICE: LOADBALANCER



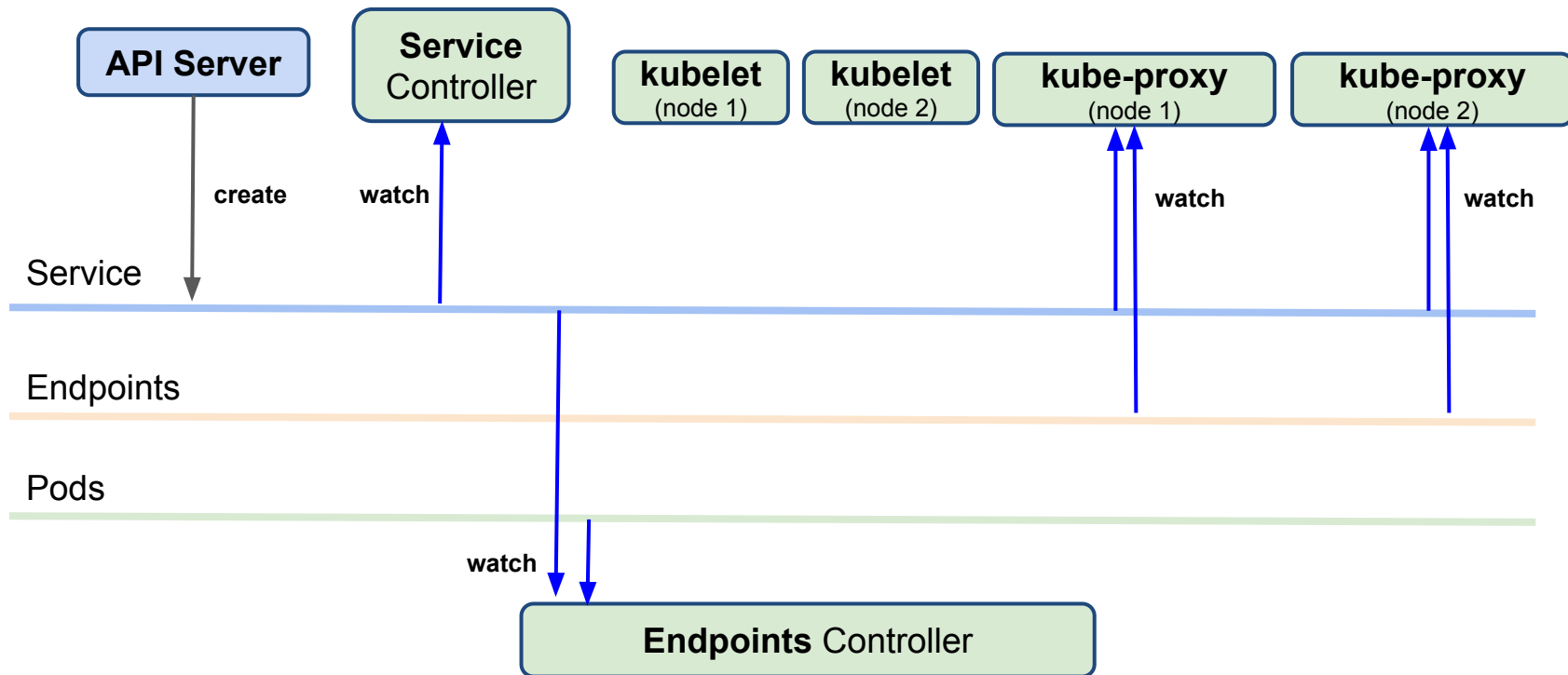
# SERVICE



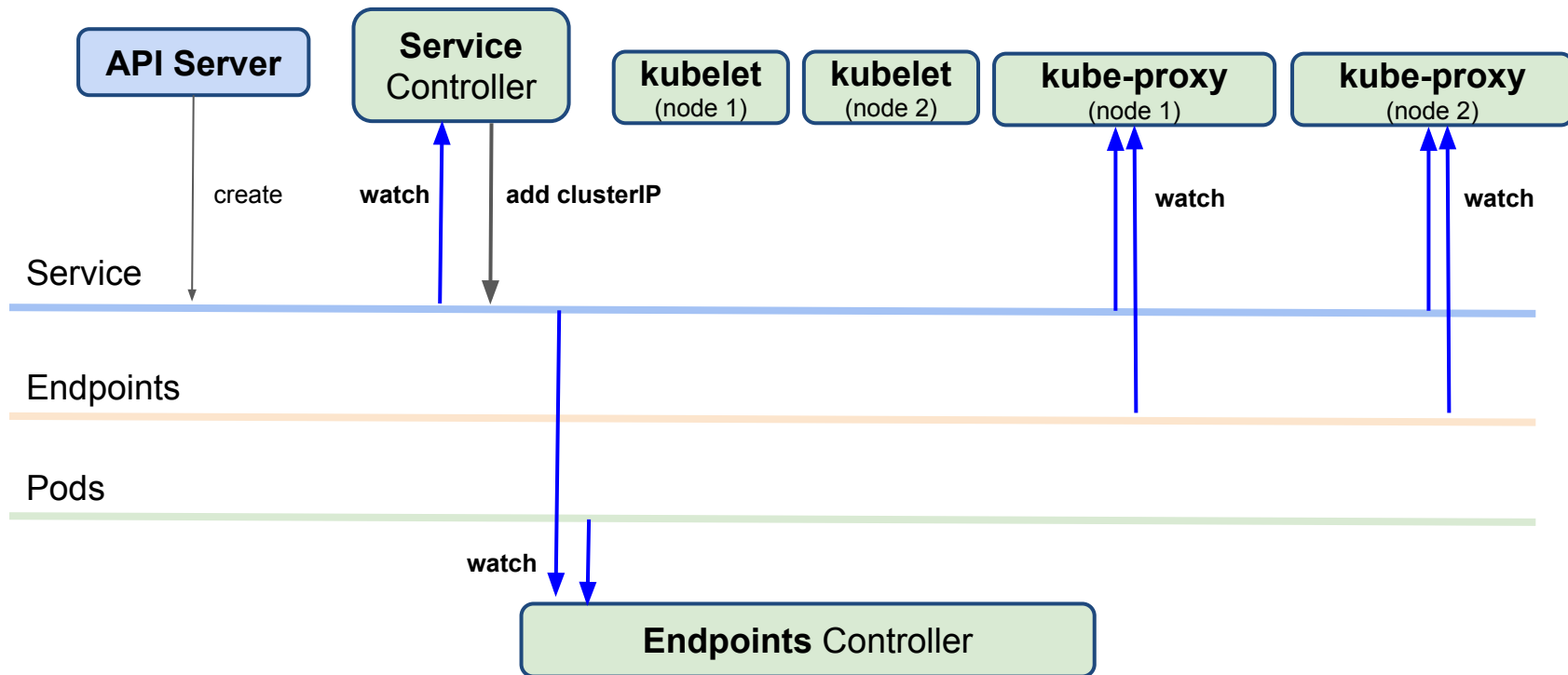
# SERVICE: creation workflow



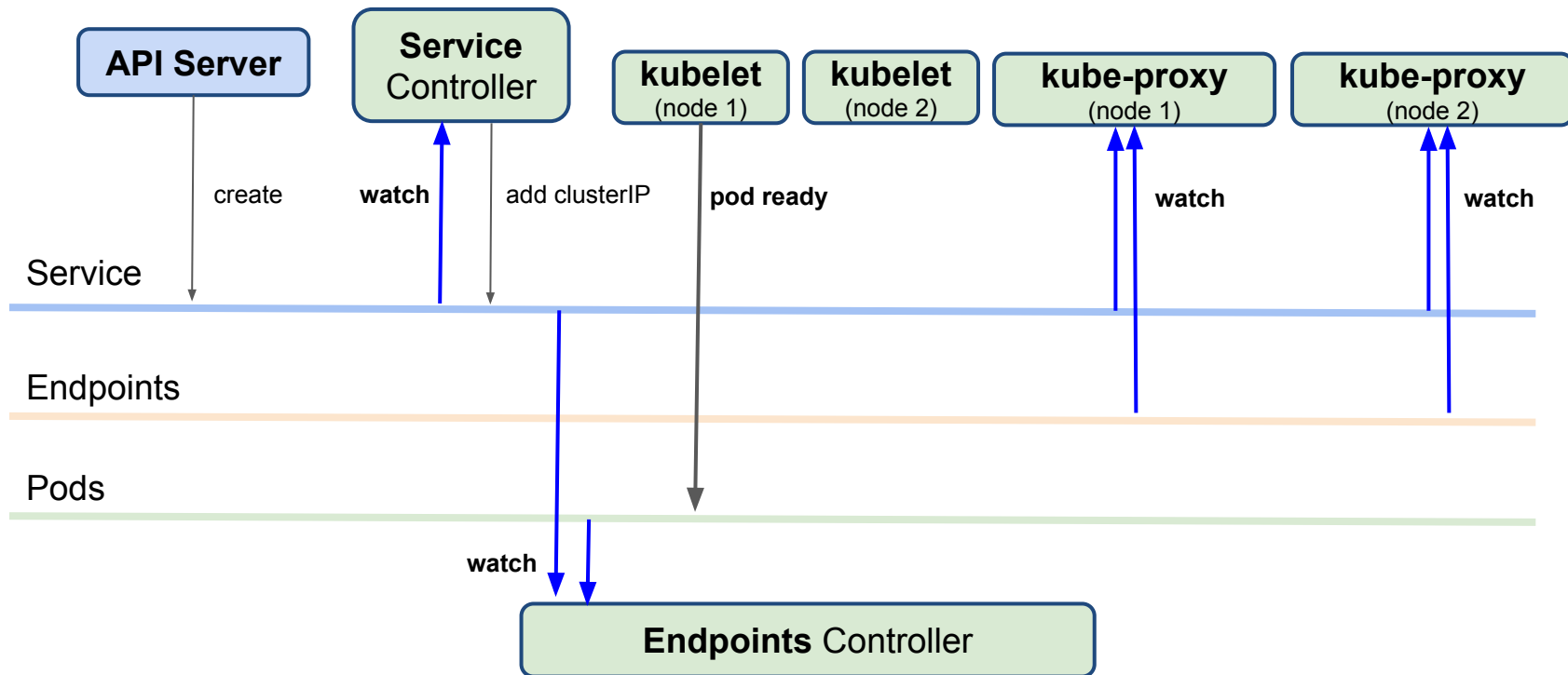
# SERVICE: creation workflow



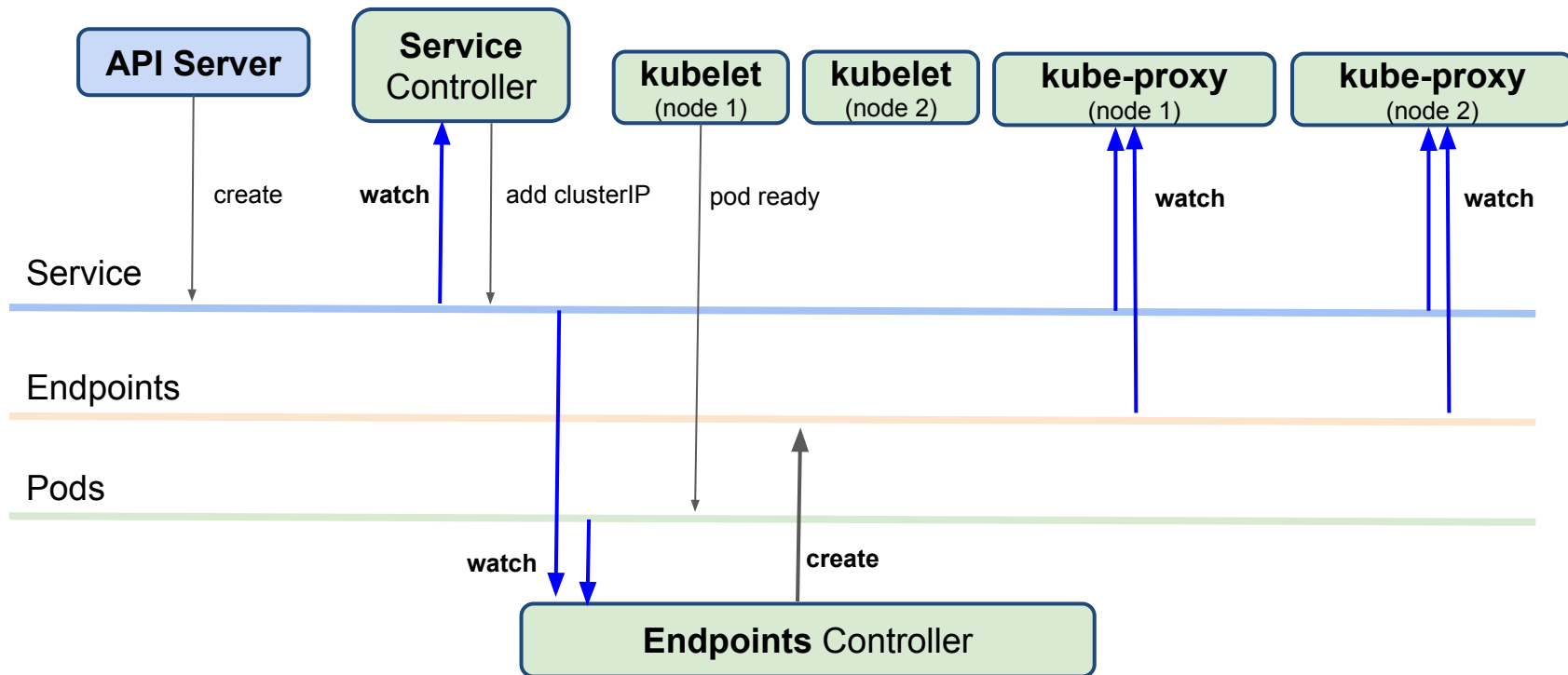
# SERVICE: creation workflow



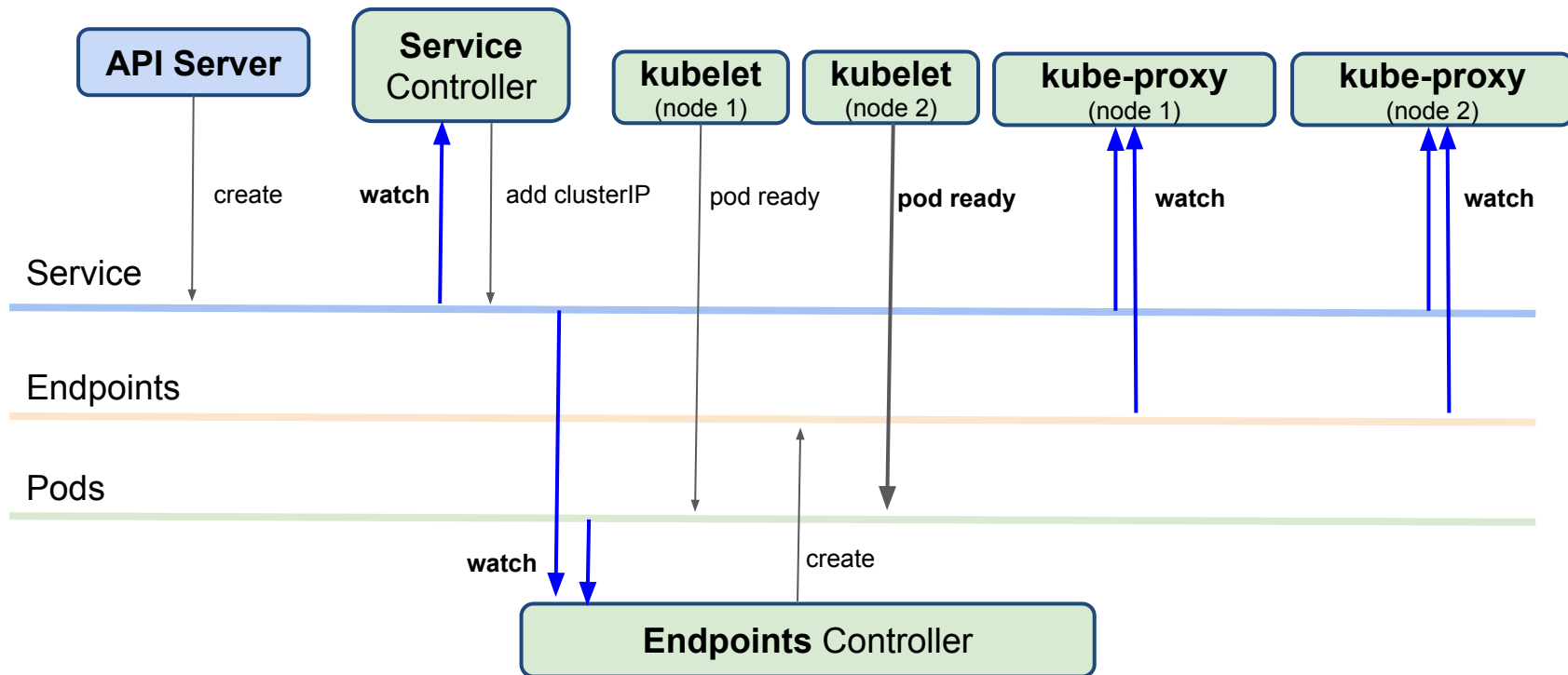
# SERVICE: creation workflow



# SERVICE: creation workflow

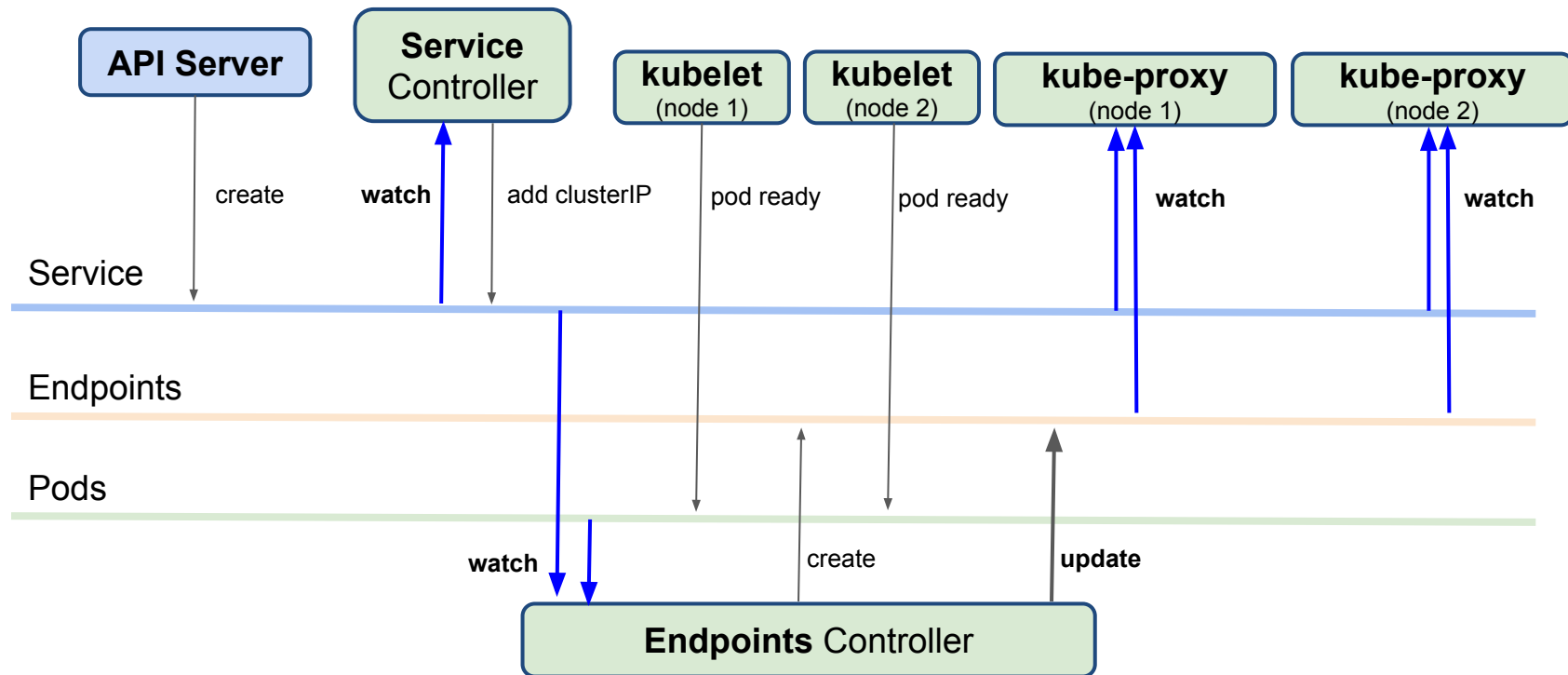


# SERVICE: creation workflow

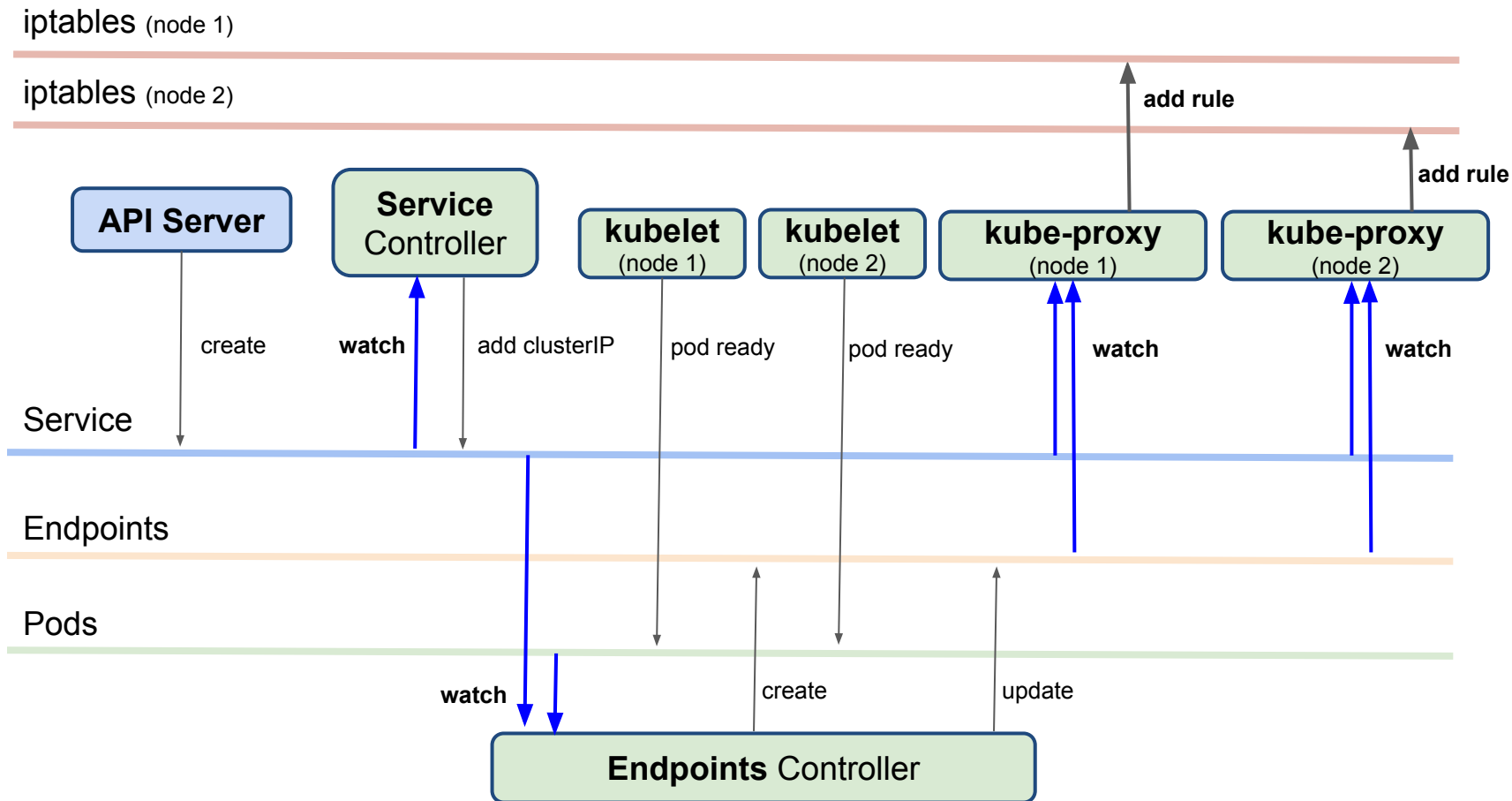




# SERVICE: creation workflow



# SERVICE: creation workflow



# DEPLOYMENT

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: votingapp
spec:
  strategy:
    type: RollingUpdate
  replicas: 3
  selector:
    matchLabels:
      app: votingapp
  template:
    metadata:
      labels:
        app: votingapp
    spec:
      containers:
        - image: paulopez/votingapp:0.1
          name: votingapp
```

# DEPLOYMENT

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: votingapp
spec:
  strategy:
    type: RollingUpdate
```

```
replicas: 3
selector:
  matchLabels:
    app: votingapp
template:
  metadata:
    labels:
      app: votingapp
  spec:
    containers:
      - image: paulopez/votingapp:0.1
        name: votingapp
```

**ReplicaSet**

# DEPLOYMENT

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: votingapp
spec:
  strategy:
    type: RollingUpdate
```

```
replicas: 3
```

```
selector:
```

```
  matchLabels:
```

```
    app: votingapp
```

```
  template:
```

```
    metadata:
```

```
      labels:
```

```
        app: votingapp
```

```
    spec:
```

```
      containers:
```

- **image: paulopez/votingapp:0.1**  
 name: votingapp

**ReplicaSet**

**POD**

# DEPLOYMENT

```
apiVersion: v1
kind: Service
metadata:
  name: votingapp
spec:
  ports:
    - port: 8080
      targetPort: 5000
  selector:
    app: votingapp
  type: ClusterIP
```

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: votingapp
spec:
  strategy:
    type: RollingUpdate
```

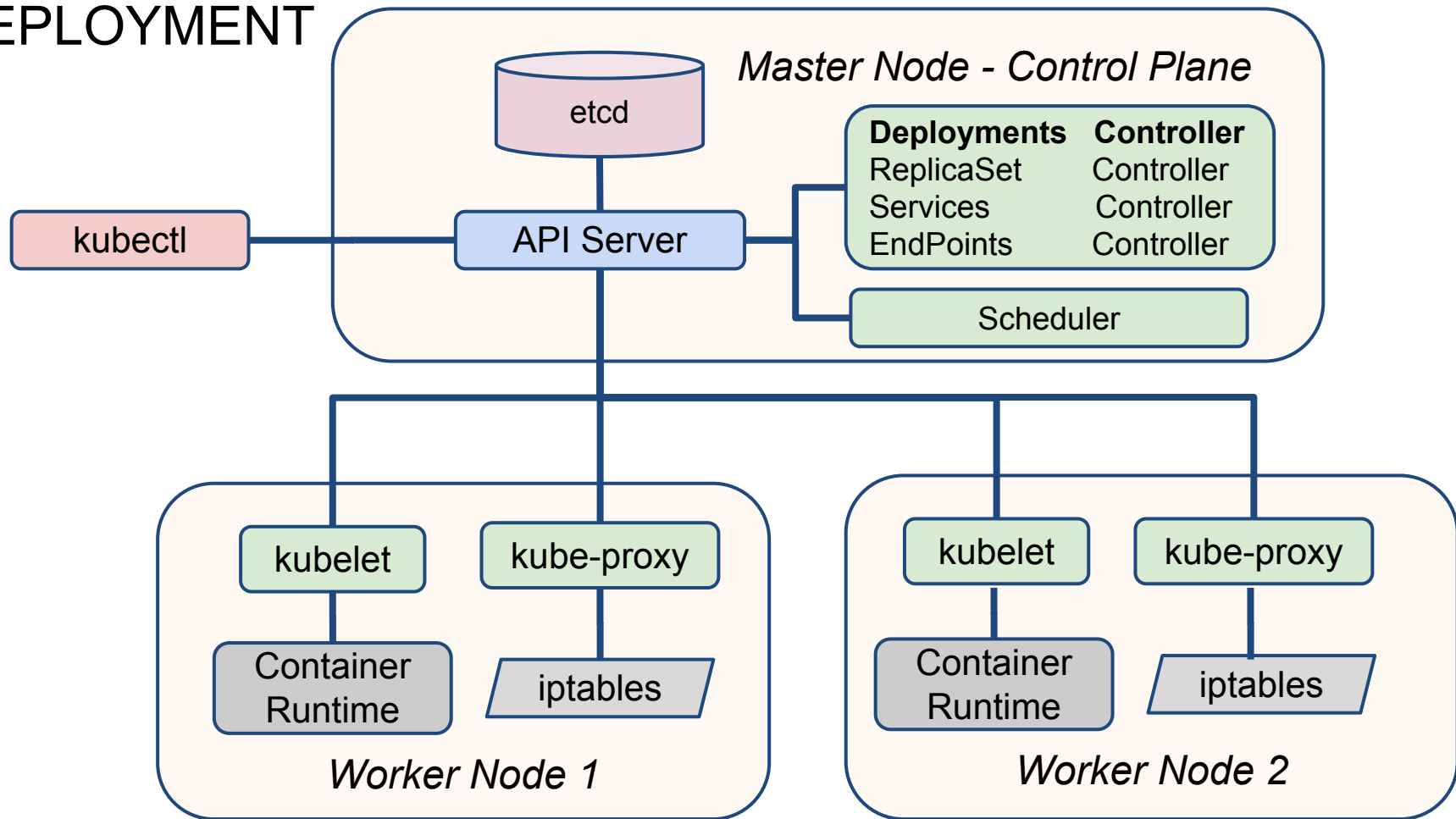
```
replicas: 3
selector:
  matchLabels:
    app: votingapp
template:
  metadata:
    labels:
      app: votingapp
  spec:
    containers:
      - image: paulopez/votingapp:0.1
        name: votingapp
```

**ReplicaSet**

**POD**



# DEPLOYMENT



# ROLLING UPDATE

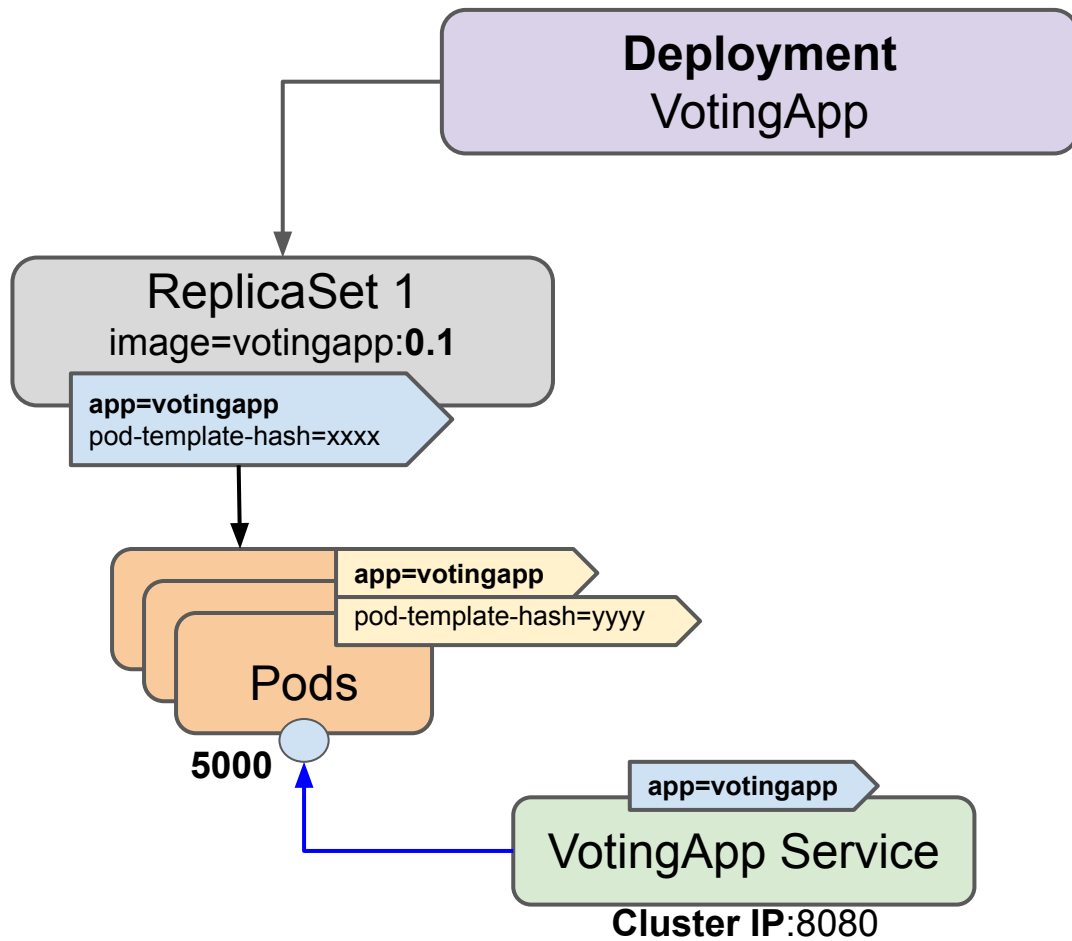
```
kubectl set image deployment/votingapp \
votingapp=votingapp:0.2-beta
```



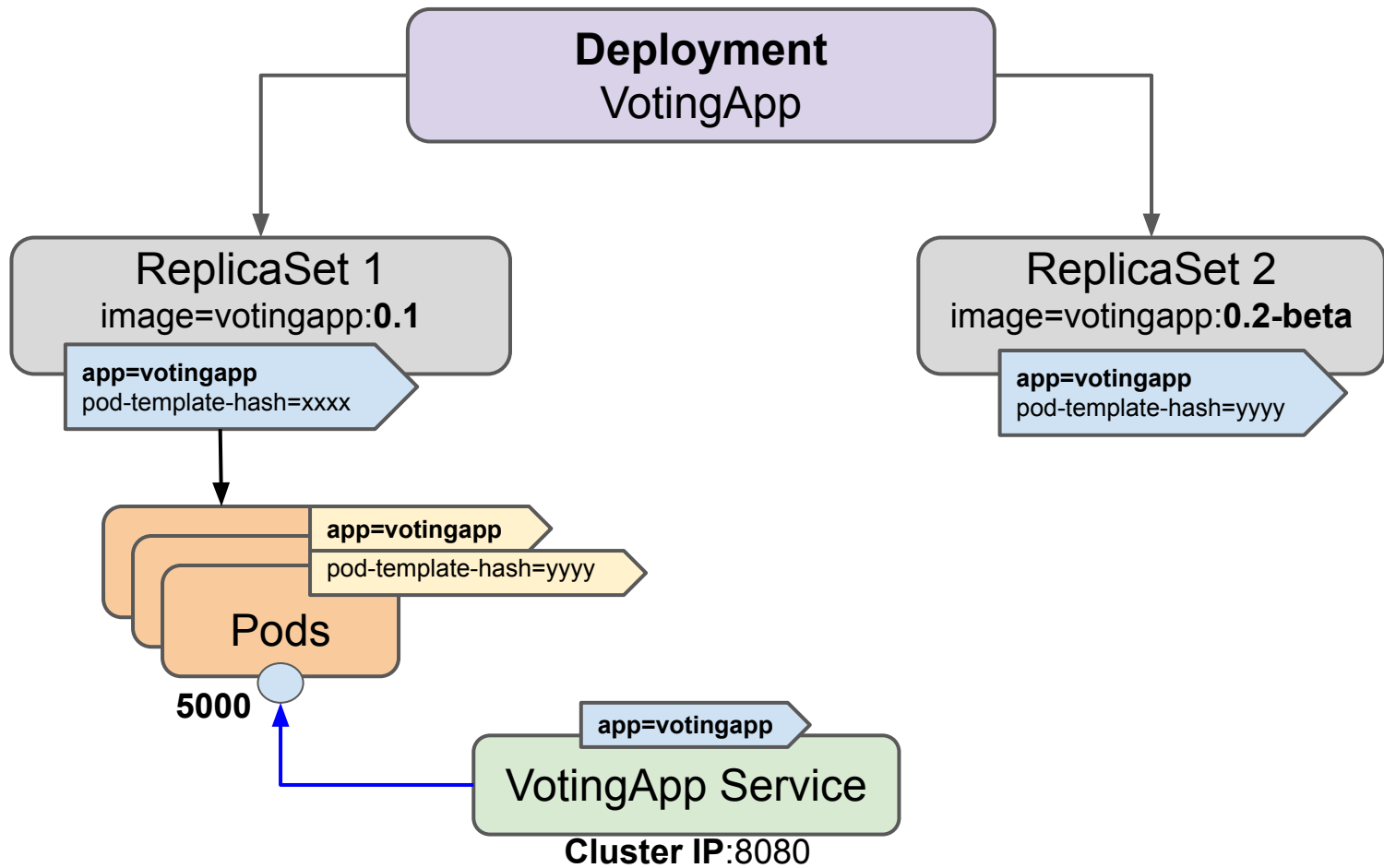
# ROLLING UPDATE

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: votingapp
spec:
  strategy:
    type: RollingUpdate
  replicas: 3
  selector:
    matchLabels:
      app: votingapp
  template:
    metadata:
      labels:
        app: votingapp
    spec:
      containers:
        - image: paulopez/votingapp:0.2-beta
          name: votingapp
```

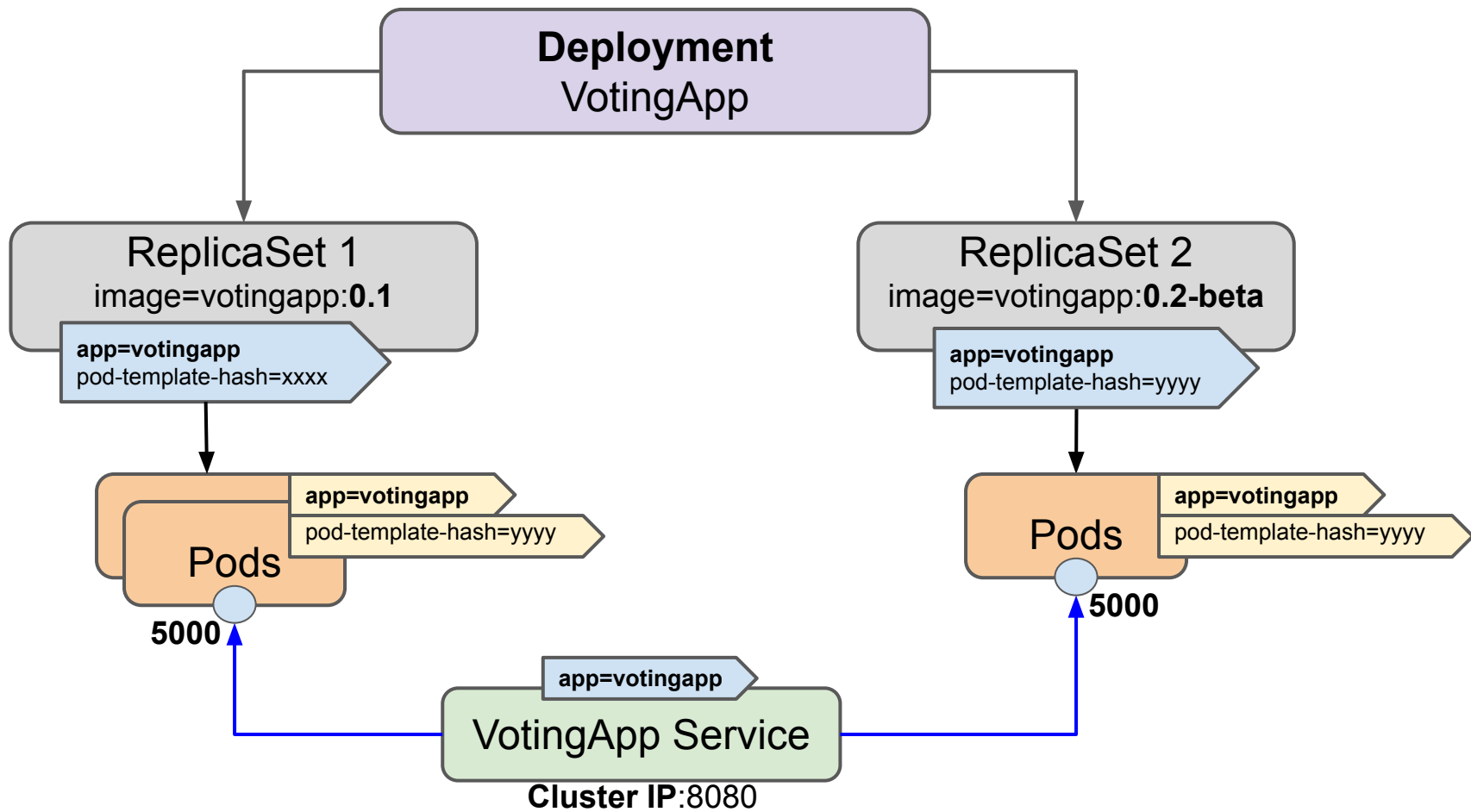
# ROLLING UPDATE



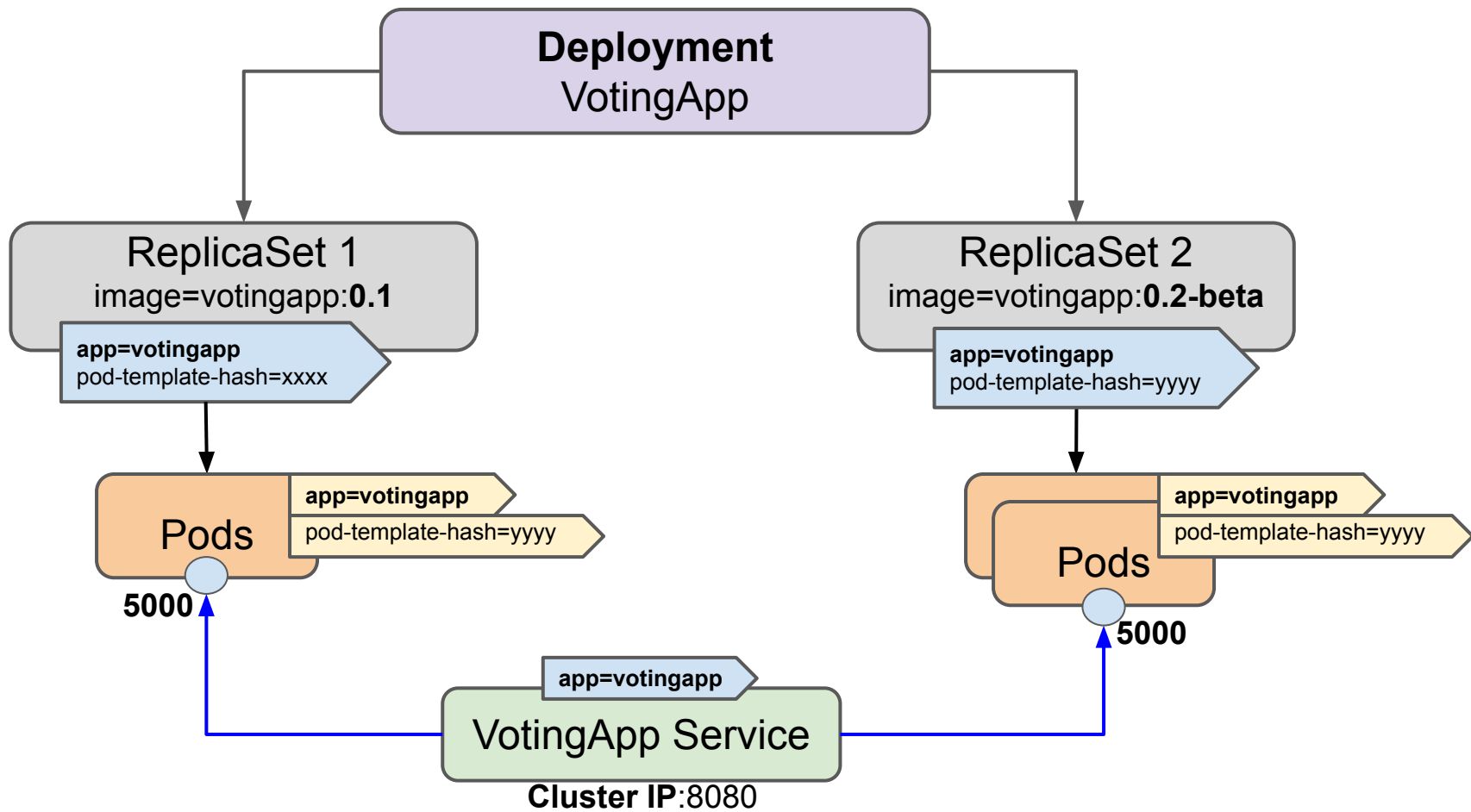
# ROLLING UPDATE



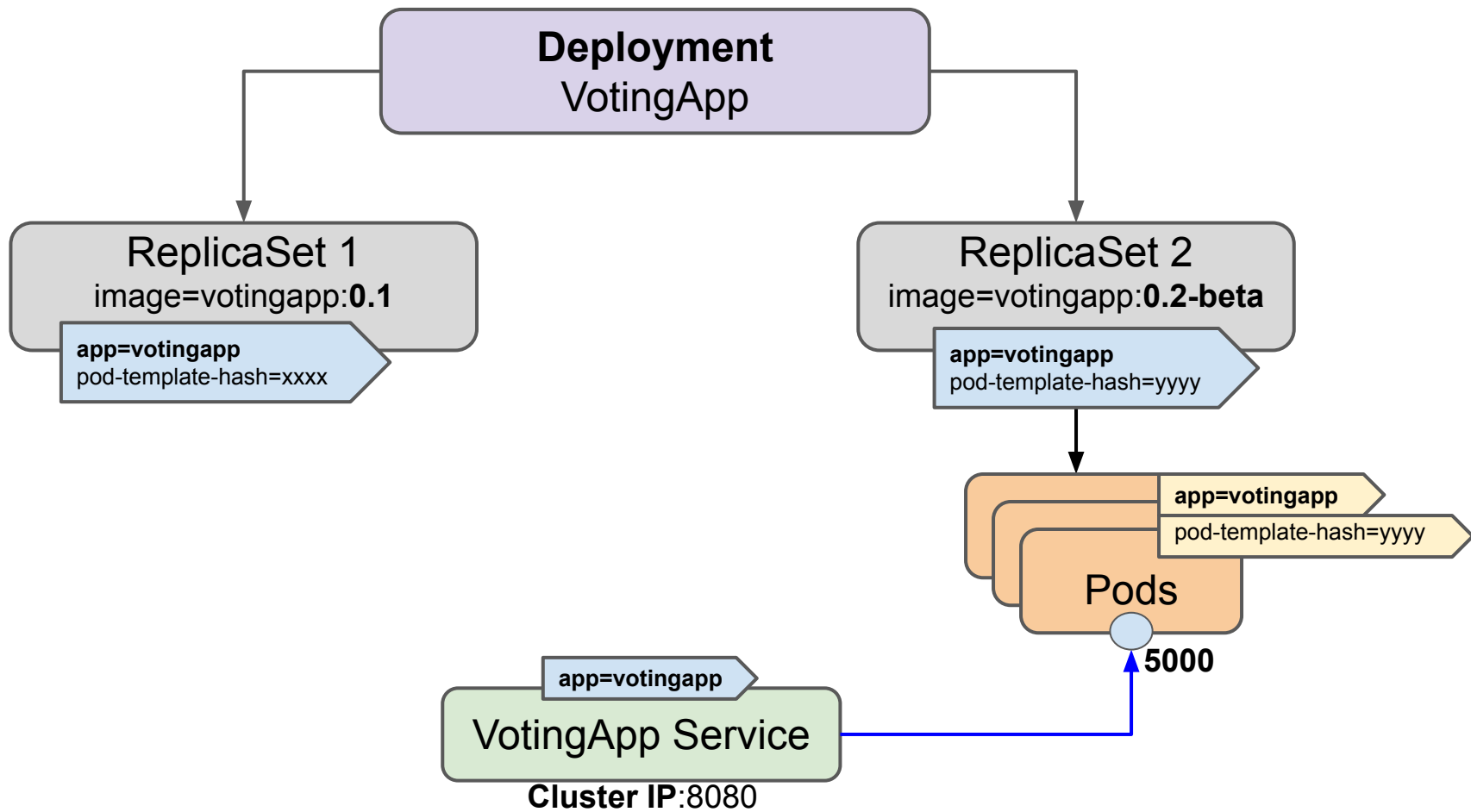
# ROLLING UPDATE



# ROLLING UPDATE



# ROLLING UPDATE



THANKS!



@memenetes