ReplicaSet in Kubernetes

ReplicaSet for App Scaling

ReplicaSet in K8s

- What is ReplicaSet
- ReplicaSet vs ReplicationController
- ReplicaSet vs Bare Pods
- HandsOn Demonstration

Replica-Set

- **ReplicaSet** is enhanced version of ReplicationController.
- Like ReplicationController, ReplicaSet's purpose is to maintain a stable set of replica Pods running at any given time.
- The main difference between a **ReplicaSet** and a ReplicationController right now is the selector support.
- Label Selectors is used to identify a set of objects in Kubernetes.

Replica Set

- **ReplicaSets** allow us to use “set-based” label selector.

- In, NotIn, Exists operators are used to Match K8s Object Labels.

Replica-Set vs Controller

```
....
spec:
  replicas: 3
  selector:
    app: alipne-box
  template:
    metadata:
....
```

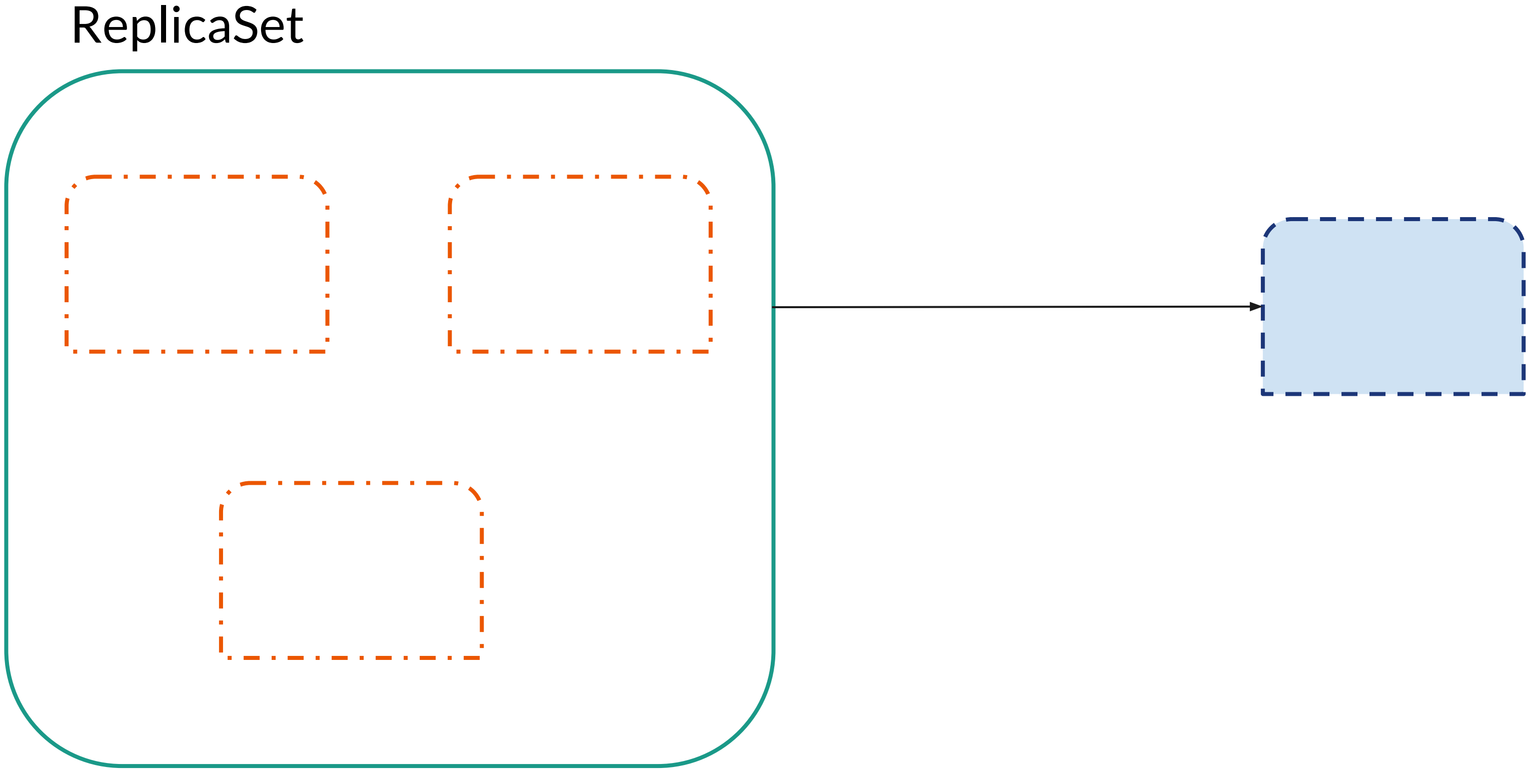
```
...
spec:
   replicas: 3
   selector:
     matchExpressions:
      - {key: app, operator: In, values: [example, example, rs]}
      - {key: teir, operator: NotIn, values: [production]}
  template:
     metadata:
...
```

Bare Pods & ReplicaSet

- While created Bare Pods, bare Pods do not have labels which match the selector of one of your ReplicaSets.

- **ReplicaSet** is not limited to owning Pods specified by its template-- it can acquire other Pods which have matching Labels.

Bare Pods & ReplicaSet



Hands On Demonstration

Thank You…

Don’t be the Same! Be Better!!!