Interacting With Your Cluster

Anthony E. Nocentino
aen@centinosystems.com

Centino systems

Course Overview

- **Module 0** - Introduction
- **Module 1** - Container Fundamentals and Introducing Kubernetes
- **Module 2** - Kubernetes Architecture and API Objects
- **Lunch @ 12:00-12:45**
- **Module 3** - Interacting With Your Cluster
- **Module 4** - Deploying Applications in Kubernetes
- **Module 5** - Building and Deploying Container-based Applications in Kubernetes

Agenda

- **Interacting With Your Cluster**
  - Using `kubectl` to Interact with Your Cluster
  - Application Deployment Basics in Kubernetes

Using `kubectl`

- Primary CLI tool
- Control your Kubernetes Cluster
  - Operations - what you want to do
  - Resources - what you want to do it to
  - Output - if there's output, its format

Operations - what do you want to do?

`apply/create` - create resource(s)

`run` - start a pod from an image

`explain` - documentation of resources

`delete` - delete resource(s)

`get` - list resources

`describe` - detailed resource information

`exec` - execute a command on a container

`logs` - view logs on a container

https://kubernetes.io/docs/reference/kubectl/overview/#operations

Resources - what do you want to do it to?

- **`nodes (no)`**
- **`pods (po)`**
- **`services (svc)`**
- ..and many more

https://kubernetes.io/docs/reference/kubectl/overview/#resource-types

Output

- Specify kubectl's output format
  - **`wide`** - output additional info to stdout
  - **`yaml`** - YAML formatted API object
  - **`json`** - JSON formatted API object

https://kubernetes.io/docs/reference/kubectl/overview/#output-options

kubectl

kubectl	[command]	[type]	[name]	[flags]
kubectl	get	pods	pod1	--output=yaml
kubectl	create	deployment	nginx	--image=nginx

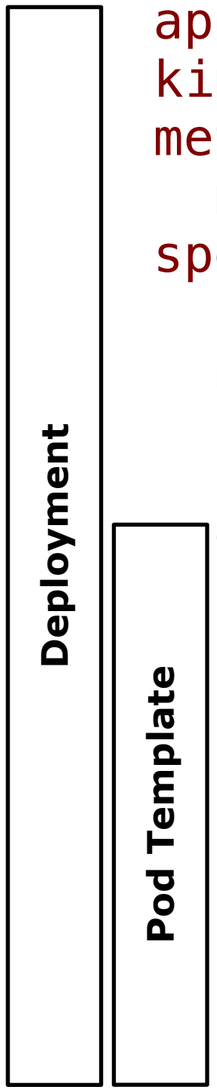
https://kubernetes.io/docs/reference/kubectl/kubectl/

https://kubernetes.io/docs/reference/kubectl/cheatsheet/

Application Deployment in Kubernetes

- Imperative
  - **`kubectl create deployment hello-world \`**
    `--image= psk8s.azurecr.io/hello-app:1.0`
  - **`kubectl run hello-world \`**
    `--image= psk8s.azurecr.io/hello-app:1.0`
- Declarative
  - Define our desired state in code
  - Manifest
  - YAML or JSON
  - **`kubectl apply -f deployment.yaml`**

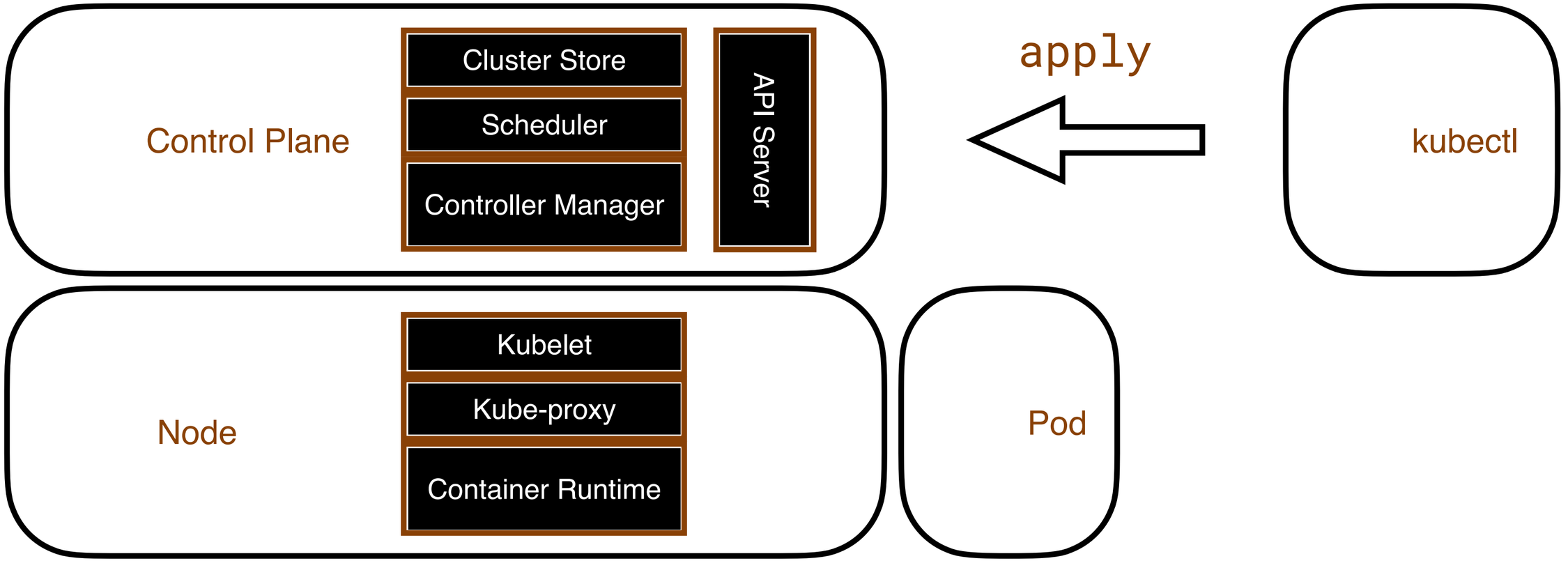
Defining a Deployment




```yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: hello-world
spec:
  replicas: 2
  selector:
    matchLabels:
      app: hello-world
  template:
    metadata:
      labels:
        app: hello-world
    spec:
      containers:
      - image: psk8s.azurecr.io/hello-app:1.0
        name: hello-world
        ports:
        - containerPort: 80
```

```
kubectl apply -f deployment.yaml
```

Application Deployment Process



Hands on lab

- Using **`kubectl`**
  - Nodes
  - Pods
  - API Resources
- Declaratively and Imperatively Deploying resources in your Cluster

Review

- **Interacting With Your Cluster**
  - Using `kubectl` to Interact with Your Cluster
  - Application Deployment Basics in Kubernetes