Inside Kubernetes
An Architectural Deep Dive

Anthony E. Nocentino
aen@centinosystems.com

Centino systems

Anthony E. Nocentino

- **Consultant and Trainer**
- **Founder and President of Centino Systems**
  - Specialize in system architecture and performance
  - Masters Computer Science
  - Microsoft MVP - Data Platform - 2017 - 2018
  - Linux Foundation Certified Engineer
  - Friend of Redgate - 2015-2019
- **email**: aen@centinosystems.com
- **Twitter**: @nocentino
- **Blog**: www.centinosystems.com/blog
- **Pluralsight Author**: www.pluralsight.com

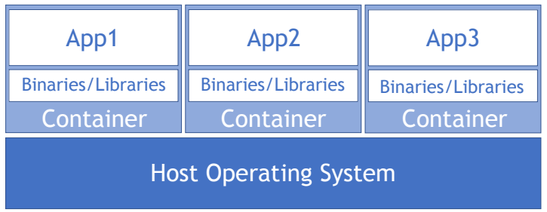
Centino systems

Agenda

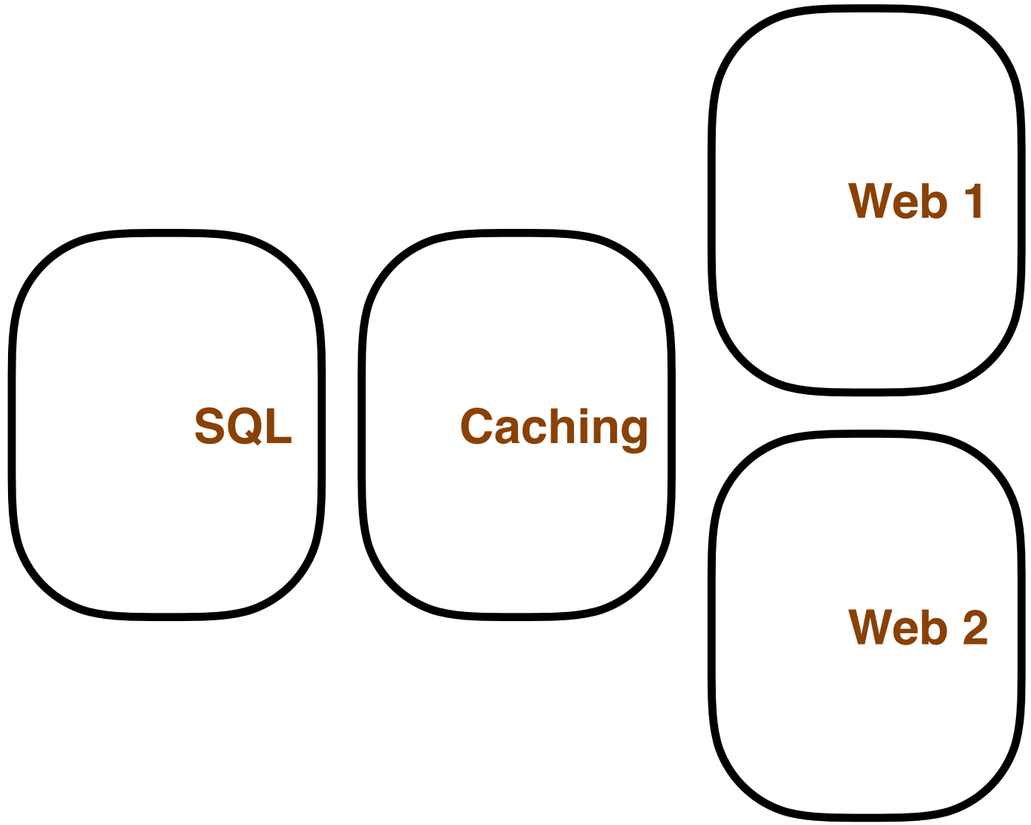
- Container Based Application Deployment
- What is Kubernetes
- Benefits of Using Kubernetes
- Kubernetes API Objects
- Exploring Kubernetes Architecture
- Deploying Applications
- Production Ready Clusters

Container Based Application Deployment

- Single-tier applications - anything written by IBM
- Multi-tier applications - Service oriented, Client/Server…
- Micro-services - smaller, more easily changed units



Modern Application Deployment



- Where do I run the application?
- How do I scale the application?
- How do I consistently deploy?
- How do I provide consistent services in a loosely connected system?

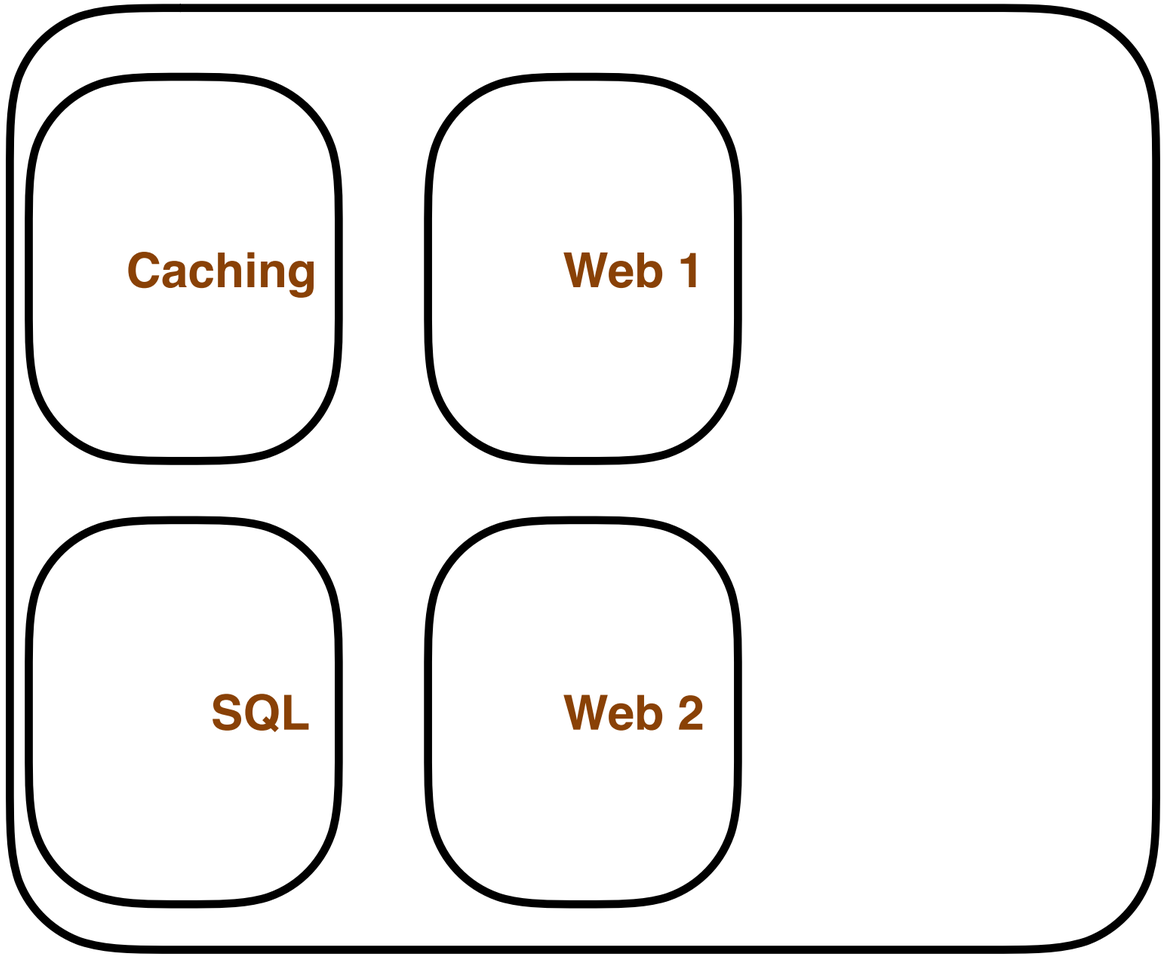
What is Kubernetes?

- Container Orchestrator
- Infrastructure Abstraction
- Desired State

Kubernetes Benefits

- Managing state, starting things and keeping them up
- Speed and consistency of deployment
- Ability to absorb change quickly
- Ability to recovery quickly
- Hide complexity in Cluster
- Persistent application access endpoints

Kubernetes Cluster



Kubernetes API

- **API Objects** - Represent resources in your system
- **API Server** - Main communication hub
  - Pods
  - Controllers
  - Services
  - Storage
  - …and more

Pods

- One or more containers
- It’s your application or service
- The most basic unit of work
- Unit of scheduling
- Ephemeral - no Pod is ever “redeployed”

Controllers

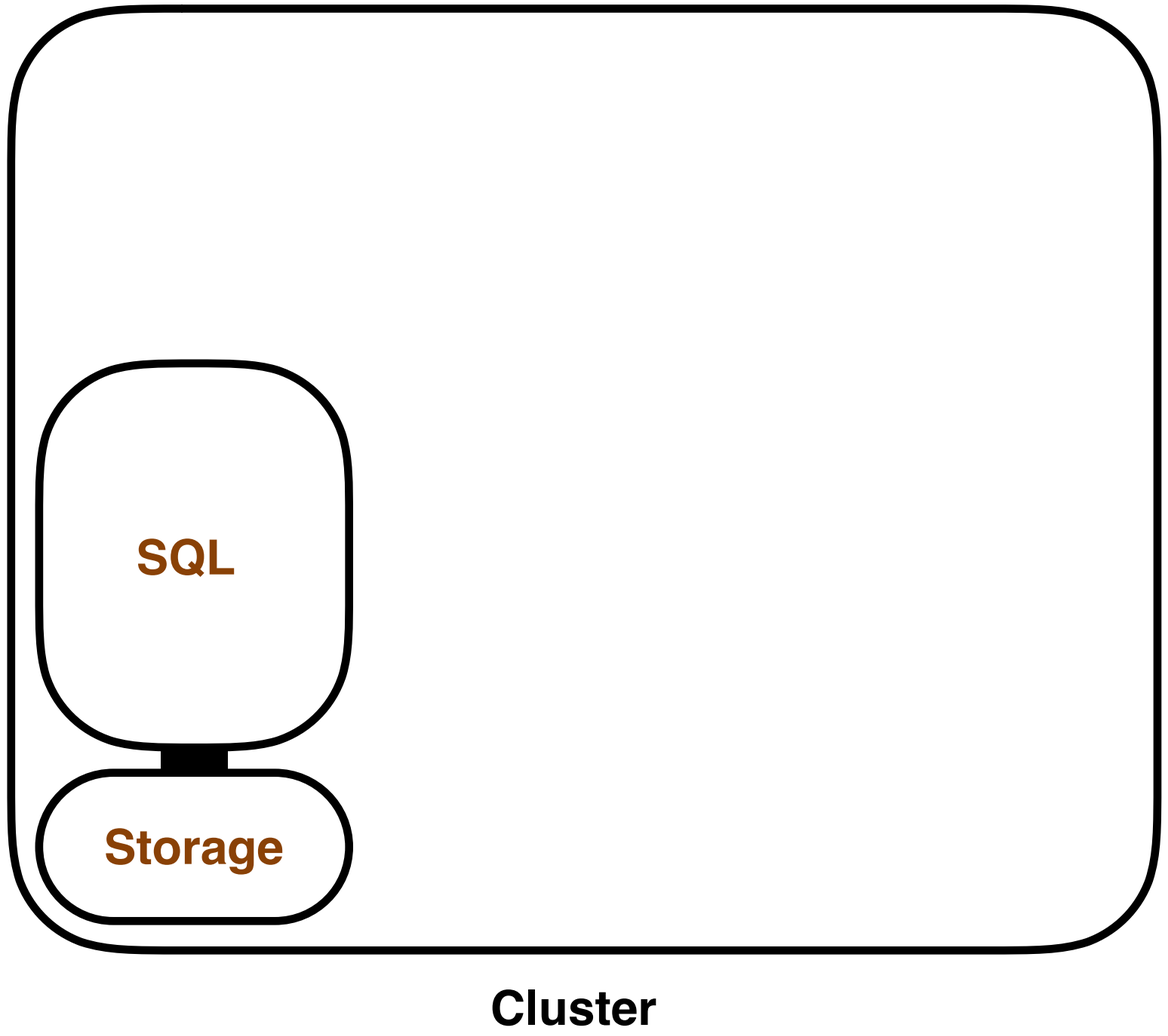
- Create and manage Pods for you
- Define your desired state
- Respond to Pod State and Health
- `ReplicaSet`
- `Deployment`

Services

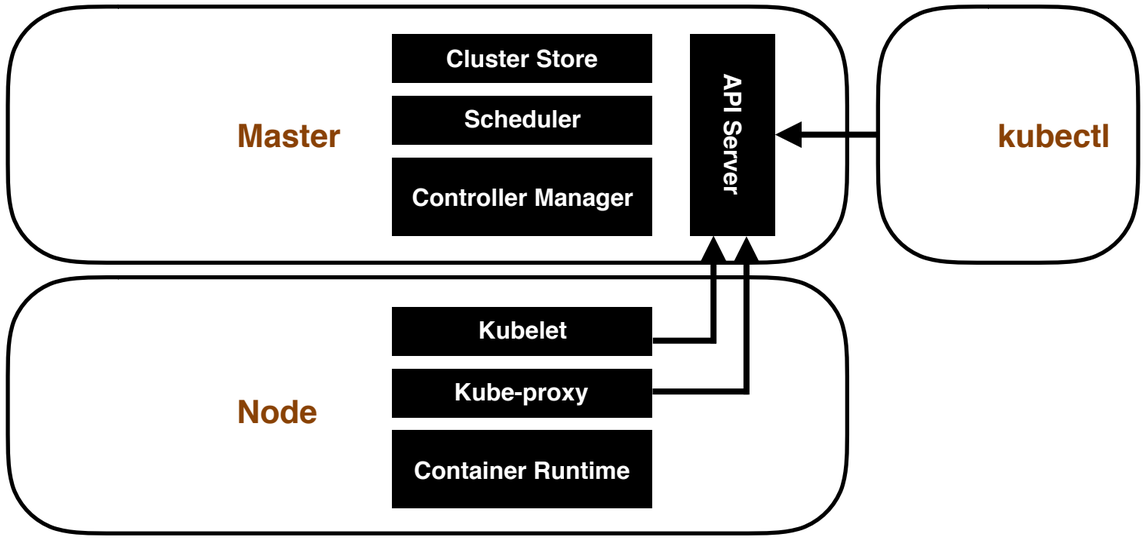
- Adds persistency to our ephemeral world
- Networking abstraction for Pod access
- IP and DNS name for the service
- Load balancing
- Redeployed Pods automatically updated
- Scaled by adding/removing Pods

Storage

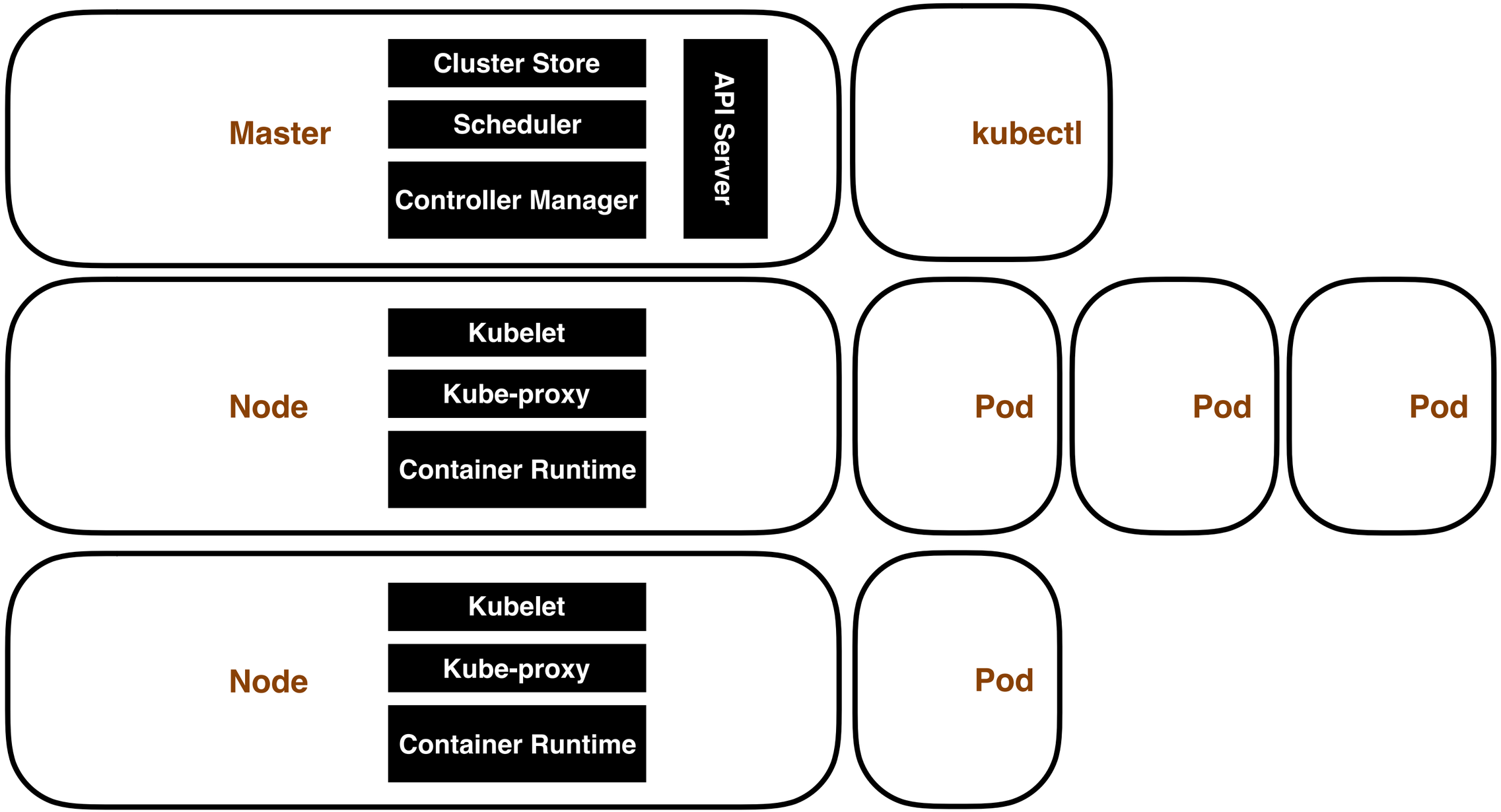
- Persistent Volumes
  - Pod independent storage
  - Administrator defined storage
- Persistent Volume Claims
  - The Pod “claims” the PV
  - Decouples the Pod and the storage



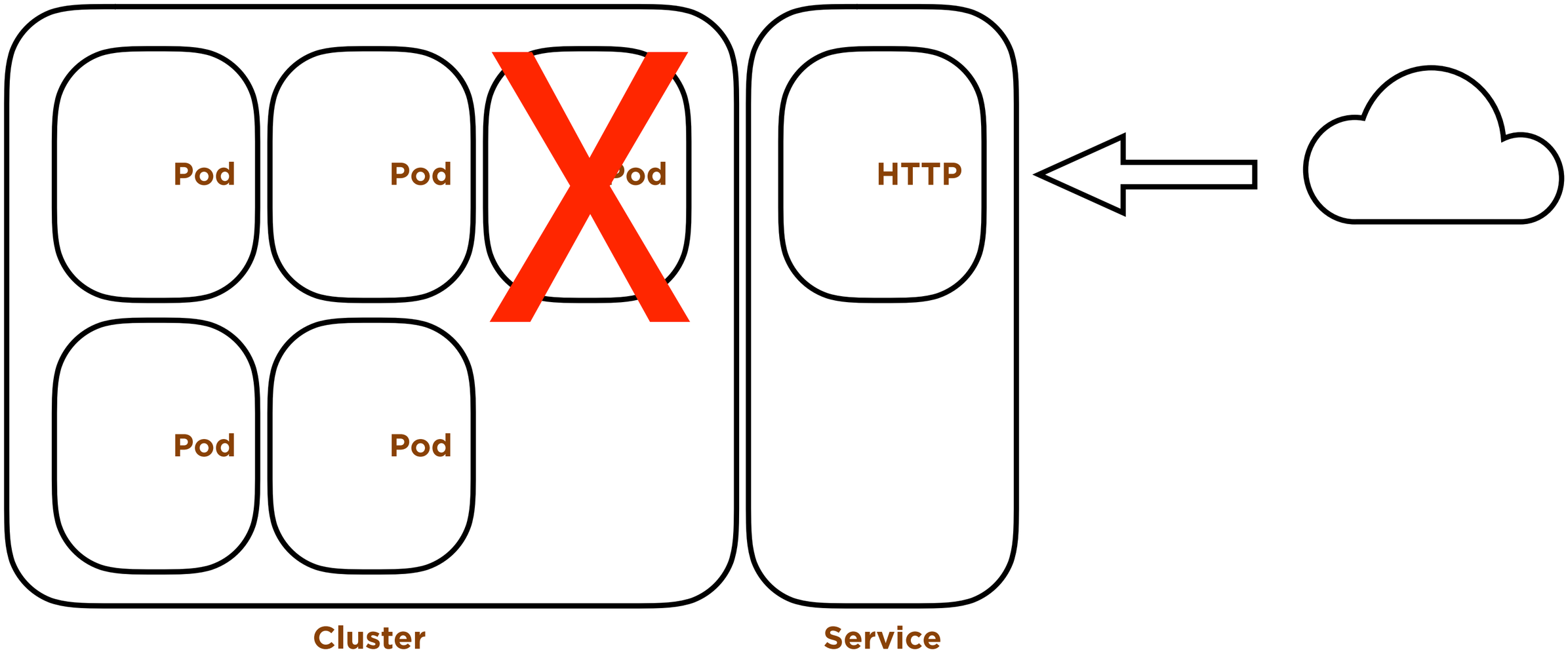
Exploring Kubernetes Architecture



Controller Operation of Pods



Services



Deploying Applications

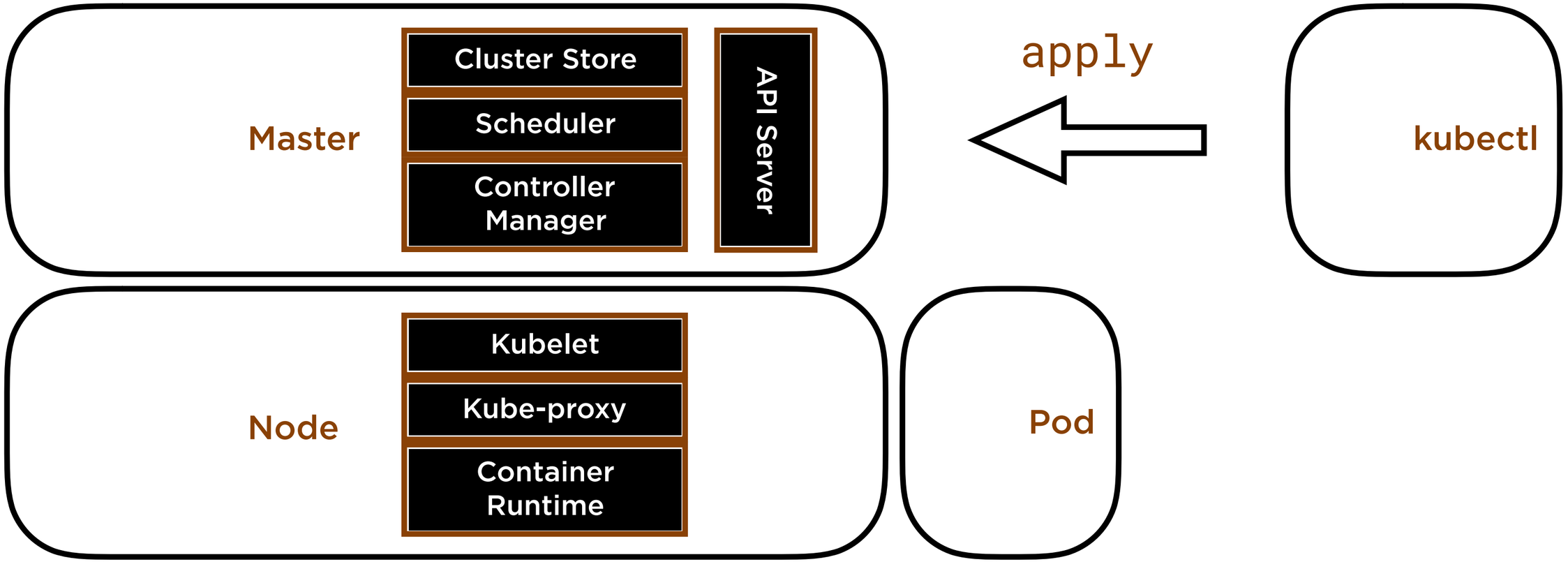
- Imperative
- Declarative
- YAML and JSON

Declarative Deployment - Manifests

```
apiVersion: v1
kind: Pod
metadata:
  name: nginx-pod
spec:
  containers:
  - name: nginx
    image: nginx
    ports:
    - containerPort: 80
```

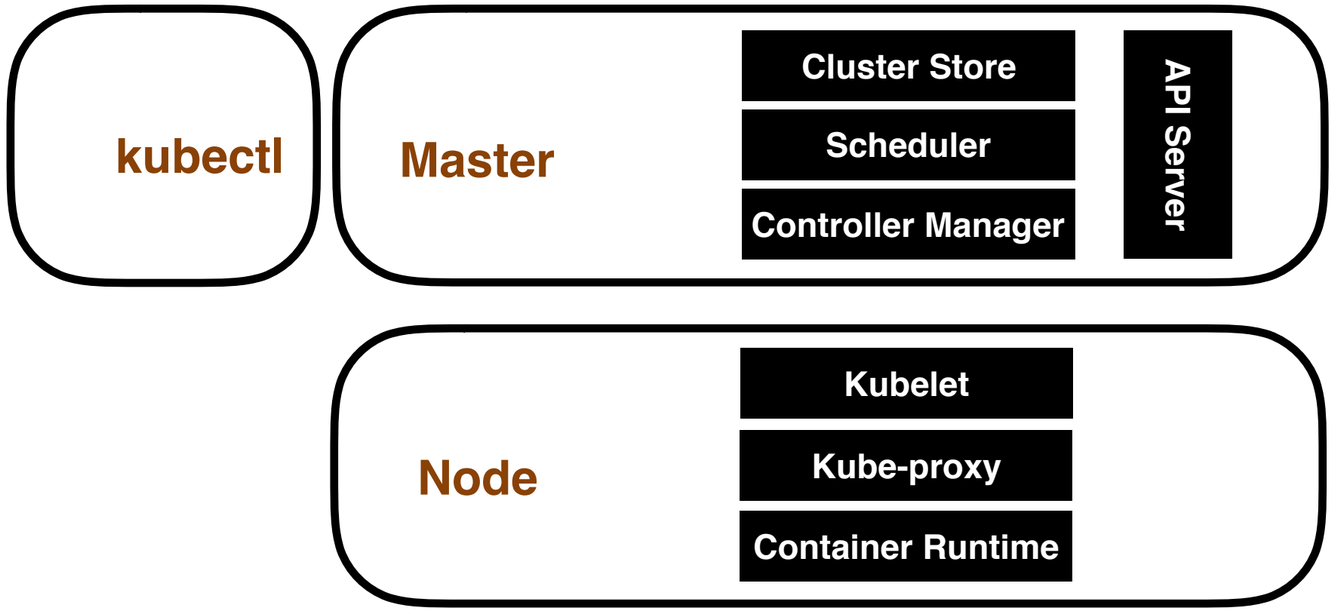
```
kubectl apply -f nginx.yaml
```

Application Deployment Process



Demo!

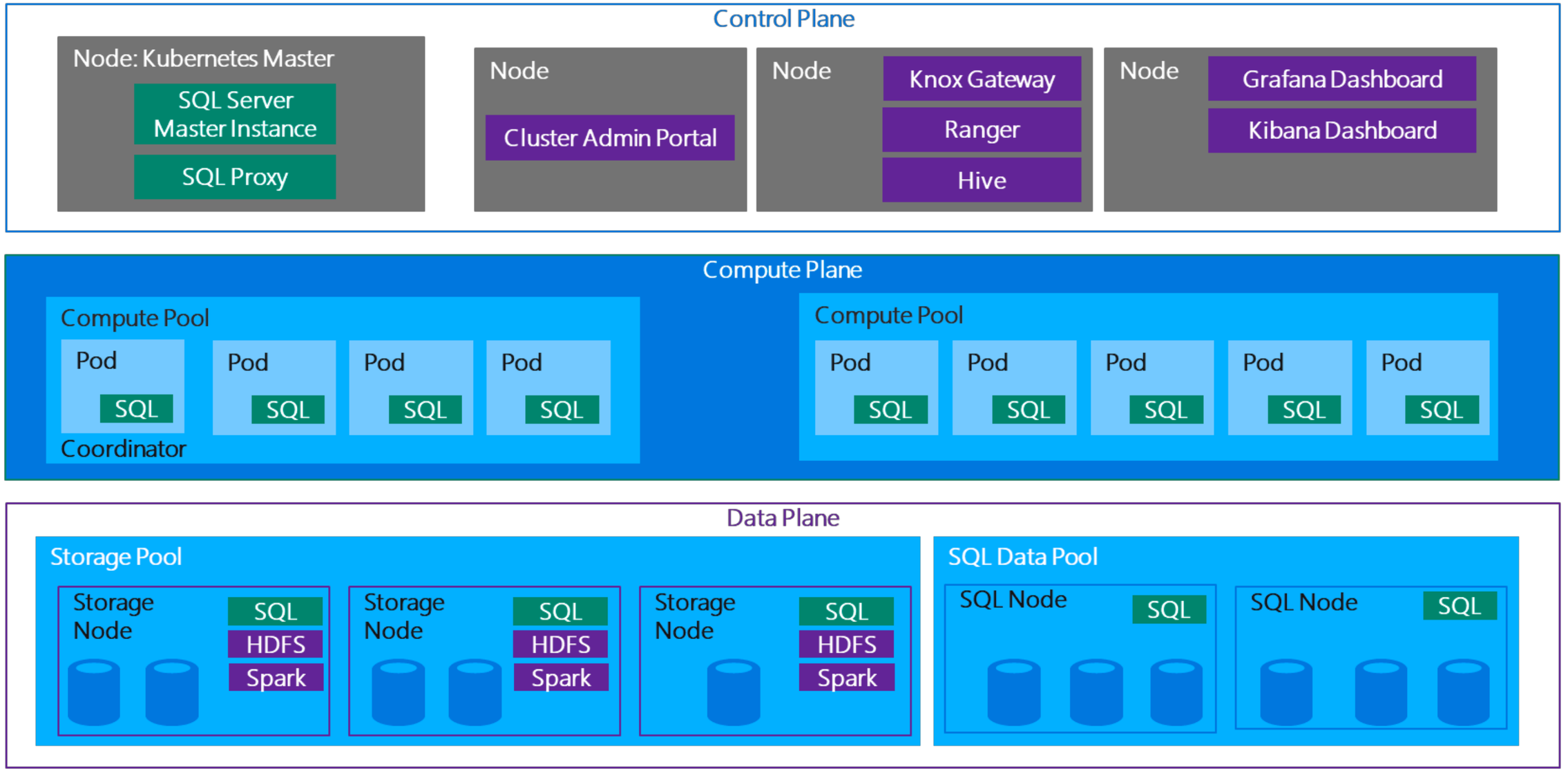
- Deploying a web application
- Accessing Services within a Cluster
- Deploying SQL Server in a ReplicaSet with Persistent Storage



Building Production Ready Clusters

- Scalability - number of Nodes
- Inter-cluster communication patterns (Network connectivity)
- High Availability
  - API Server - Load Balanced
  - etcd - Multiple Replicas
- Disaster Recovery
  - etcd Backups
- Persistent Volumes

Kubernetes Cluster



From: https://docs.microsoft.com/en-us/sql/big-data-cluster/big-data-cluster-overview?view=sqlallproducts-allversions

Review

- Container Based Application Deployment
- What is Kubernetes
- Benefits of Using Kubernetes
- Kubernetes API Objects
- Exploring Kubernetes Architecture
- Deploying Applications
- Production Ready Clusters

More Resources

- **Docker for Windows/Mac**
- **Minikube**
- **Managed Service Providers**
  - Azure Kubernetes Service **(AKS)**
    - https://docs.microsoft.com/en-us/azure/aks/kubernetes-walkthrough
  - **Elastic Container Service for Kubernetes (EKS)**
    - https://aws.amazon.com/getting-started/projects/deploy-kubernetes-app-amazon-eks/
  - **Google Kubernetes Engine (GKE)**
    - https://cloud.google.com/kubernetes-engine/docs/how-to/
- **Pluralsight! - Kubernetes Installation and Configuration Fundamentals**
  - https://app.pluralsight.com/profile/author/anthony-nocentino

Need more data or help?

http://www.centinosystems.com/blog/talks/

Links to resources
Demos
Presentation
Pluralsight

aen@centinosystems.com
@nocentino
www.centinosystems.com

Solving tough business challenges with technical innovation

Centino systems

Questions?

Thank You!